Wellbore Heat Transfer Model for Wax Deposition in Permafrost Region

By

Xiaoting Cui

Submitted to the graduate degree program in Chemical and Petroleum Engineering and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science.

Committee: _____

Chairperson   Jenn-Tai Liang

_____

G. Paul Willhite

_____

Jyun Syung Tsau

Date Defended: _____

The Thesis Committee for Xiaoting Cui

certifies that this is the approved version of the following thesis:

Wellbore Heat Transfer Model for Wax Deposition
in Permafrost Region

Committee: _____

Chairperson   Jenn-Tai Liang

_____

G. Paul Willhite

_____

Jyun Syung Tsau

Date approved: _____

2

Abstract

Producing waxy oil in arctic area may cause wax deposited on the well wall. Since wax deposition is strongly thermal related, accurate heat transfer model is necessary in predicting and preventing wax depostion. A mathematical model was derived based on energy balances for heat exchange between the producing fluids and production string as well as the formation/permafrost. To simplify the calculation, oil and gas were assumed well mixed as one single-phase in the tubing.

Furthermore, Singh's model for wax deposition was coupled with the heat transfer model. Wax concentration and effective diameter were updated with time in the temperature calculation. Pressure distribution was calculated over time to check whether the reservoir energy was sufficient to produce the oil during the production process. Besides, a user friendly GUI was developed by VB and MATLAB to run the simulation.

The effects of permafrost, thermal insulation, well geometry and wax deposition on the heat transfer calculation were studied. Simulation results illustrated insulating the wellbore and evacuating the production casing annulus effectively reduced the wellbore heat loss. The model can be used in the temperature prediction of an injection well or production well in permafrost region or non-permafrost region.

Acknowledgement

First and foremost, I would like to express my sincere gratitude to Dr. Jenn-Tai Liang for his guidance throughout my study at university of Kansas, especially the understanding and support in the toughness time of my life.

I am much obliged to Prof. G. Paul Wilhite for his enlightening guidance from the initial to the final enabled me to develop an understanding of the subject and his great efforts to explain things clearly and simply.

Deep appreciation extends to Dr. Jyun-Syung Tsau for serving on my committee and providing sound advice and encouragement besides the research.

Special thanks go to my colleague Yaqin Wu for providing the code of wax deposition model. Thanks also go to all my friends here who made my life in KU more joyful.

Lastly, and most importantly, thank my husband (Qunfeng He) for his companionship getting me through all the difficult times with his beautiful heart. Also, I owe a debt of gratitude to my parents for their continuous support and unconditional love through my life. I dedicate this thesis to them.

Table of Contents

List of Figures

1. Introduction

In the wellbore, oil temperature drops after oil leaves the reservoir due to heat loss to the surroundings. In some cases, prediction of the temperature profile is crucial to flow assurance, optimization of the oil production strategy and minimization of the cost. The first model for temperature prediction was developed by Ramey for an injection well in 1962 [1]. Based on Ramey's work, many other models were developed involving extended energy balance and coupled multiphase fluid flow equations with energy balance equations.

Temperature prediction of waxy oil is more complicated. In some reservoirs crude oil contains waxy constituents which will precipitate from the oil when the temperature of the oil decreases to a value known as the wax appearance temperature (WAT). When this happens in a tubing string or flow line, a layer of wax builds up on the wall of the conduit and the process continues as long as the temperature of the flowing oil is at or less than WAT. The wax deposit reduces the heat loss from the oil to the surrounding because of its low thermal conductivity. However, reduction of the effective diameter occurs as wax deposition increases the thickness of the wax layer. As a consequence, the production rate decreases under a fixed pressure drop. When the production rate is too low, the conduit is shut down and the wax has to be removed by scraping from the walls or by injecting hot oil or other solvent to dissolve the deposit. In the worst case, the conduit may become plugged with wax and flow ceases abruptly.

The deposition of wax in production tubing is a potential problem when the production well passes through permafrost in the Alaska North Slope (ANS) area. In this case, heat is lost from the production well into the permafrost and may cause melting in the immediate vicinity of the well. Melting renders a nearly constant low temperature in the region around the production well, which contributes to further wax deposition.

It is possible to predict the wax deposition by coupling wax deposition with heat transfer model like the multiphase flow simulator OLGA [2] and the flow model developed by University of

Tulsa [3]. However, models contain proprietary coding and are not available to the general public. There is also not enough published information to assess how the wellbore structure is integrated with wax deposition and heat transfer calculation.

The purpose of the study is to develop a mathematical model to analyze the thermal performance of the well with typical wellbore structure, especially in extremely cold environment. In addition, the wax deposition during the production of waxy crude oil is studied. The model development includes three parts: developing a heat transfer model, coupling Singh's wax deposition model, and designing a graphic user interface to permit easy access to the model. Each part is introduced in sequential chapters.

Example calculations using the model are presented in Chapter 6. These examples demonstrate the effects of thermal insulation, production rate, well geometry, and wax deposition on the production. The results show the model can be applied to evaluate various production scenarios of an oil well in permafrost region.

2. Literature Review

Flow assurance refers to ensuring successful and economical flow of hydrocarbon stream from reservoir to the point of sale [4]. One of the challenges to flow assurance is pipeline blockage due to hydrates or wax. The calculations of fluid temperature, pressure and wax deposition are critical to ensure the flow assurance. This chapter surveys models of temperature prediction regarding their limitations and applications. Singh's model is introduced as the basis to predict wax deposition in a producing wellbore.

Section 2.1 summarizes the studies on fluid temperature prediction in the wellbore. Especially the effect of permafrost on the wellbore heat transfer is discussed. Section 2.2 reviews various models of wax deposition and details on Singh's model are given as an example. Methods for wax control are briefly introduced as a necessary part of the overall technical solution.

2.1 Current Models for Temperature Prediction

2.1.1 Wellbore Heat Transfer Calculation

Temperature prediction in production string is of significance to wellbore design and operation. The first heat transfer model was proposed by Ramey in 1962 [1], which gives an analytical solution of radial heat transfer across the well for incompressible liquid or ideal gas flow. The model assumes the radius of formation to be infinitely large. Although the model was developed for an injection well, it can be easily adapted to production fluid by changing the sign of mass flow.

$$T(z,t) = az + b - aA + (T_0 + aA - b)e^{-z/A}$$ Eq. 2-1

$$A = \frac{Wc_f \left[k_e + r_{to}Uf(t) \right]}{2\pi r_{to}Uk_e}$$ Eq. 2-2

Where a is geothermal gradient, ˚F/ft; b is surface geothermal temperature, ˚F; T_0 is temperature of injected fluid at the wellhead, ˚F; z is depth below surface, ft; and t is injection time, days.

In Ramey's model, only conduction and convection are considered in the overall heat transfer coefficient calculation. Based on Ramey's model, Willhite presented an approach to evaluate the heat transfer coefficient defined in Ramey's solution for numerous configurations [5]. The heat resistance calculations involve convection in the tubing, radiation and convection in the casing annulus and conduction in the tubing wall, casing wall, cement and formation. Methods proposed in the article have been widely cited and adopted.

Some models are derived based upon Ramey's assumptions such as extending the energy balances while considering the effect of pressure drop and coupling multiphase fluid flow equations with energy balance equations [6-9]; introducing Joule-Thompson expansion and kinetic energy effect for the well with gas injection [10-12]; adding a heat source term for the production well with downhole heaters [13].

However, above models were derived for vertical wells. As technology evolves, wells have been drilled purposely to deviate from the vertical angle to reach an objective location with a controlled angle. Hence models for vertical wells have to be corrected by incorporating the well inclination angle to extend the applicability.

2.1.2 Temperature Calculation in Formation/Permafrost

Assume that heat flows from the wellbore to the formation by conduction, several analytical methods were proposed to calculate formation temperature. Carslaw and Jaeger derived the

12

solution for the transient heat flow in an infinite cylindrical system with initial uniform temperature and constant boundary temperature [14]. Based on Carslaw and Jaeger's work, Ramey developed an analytical solution of constant heat-flux line source for a long time [1]. Hasan and Kabir proposed a similar solution to predict the transient formation temperature around a finite wellbore [7]. Both show good prediction accuracy compared with numerical solutions.

Permafrost is normally defined on the basis of temperature, as the soil or rock that remains below 0 ℃ and maintains a frozen layer for more than years [15]. In permafrost regions, it is necessary to consider the possibility in oil production that heat loss to the formation may cause melting of the permafrost. Permafrost melting results in soil consolidation, subsidence, and even damage to the wellbore.

Moving boundary refers to the interface separating thawed and frozen permafrost. The prediction of moving boundary location is unique to the permafrost temperature calculation. Sengul developed a heat transfer model which assumes the permafrost is thawed at one specific temperature [16]. Instead, some researchers hold that the permafrost melting and freezing take place within a temperature interval. Couch called the interval melting zone where the heat capacity changes continuously with the release of latent heat during the whole melting/freezing process in the calculation of enthalpy change [17]. Merriam used a ramp function of specific heat within the range [18]. To simplify the calculation, the permafrost thawing temperature is set to be 32 ℉ in our study.

Note that models presented in Section 2.1 do not combine the heat transfer calculation in the permafrost with that in the wellbore. It is necessary for our study to develop a simplified model to describe the heat transfer mechanisms in both wellbore and formation/permafrost. Heat transfer study in such scenario will help to evaluate the production and preventive measures against wax deposition.

13

2.2 Prediction of Wax Deposit in a Producing Oil Well

2.2.1 Wax Deposition

Wax forms and deposits on the oil flow lines when oil temperature falls below WAT. The thickness of wax deposition depends on temperature, oil composition, fluid velocity, and pressure [19]. As illustrated in Figure 2.1, wax precipitation and deposition reduce the area open to flow, causing significant loss in production and additional operating costs.
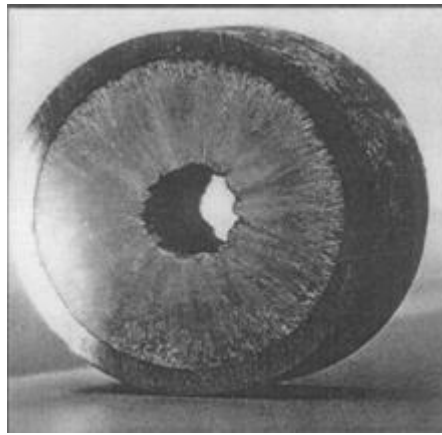
Fig. 2-1 Cross-sectional view of plugged pipeline [20]

2.2.2 Prediction of Wax Deposition

A proper wax deposition model is of much significance in flow assurance evaluation. There are two major factors leading to the wax deposition, molecular diffusion and shear dispersion. Wax deposition is molecular diffusion dominant when the flow is in the laminar regime. When the flow is in the turbulent regime, the stress effect on wax deposition should not be neglected. These two kinds of wax deposition models have been proposed depending on the flow regime.

Singh developed a wax deposition model based on studies in a short flow loop under laminar, single-phase flow condition [21-23]. The major assumptions of the model are: 1) wax deposition

is dominated by molecular diffusion; 2) the whole process is in quasi-steady state; 3) heat is transferred only in the radial direction; 4) uniform gel film; 5) constant fluid and wall temperature; 6) thermal conductivity of the gel is a function of its wax content. Aging and growth rates are derived from the mass balance. A numerical method is used to calculate the changes of wax thickness and the wax fraction in the gel over time. At each time step, the effective radius of the flow loop is updated. Singh's model was adopted to integrate wax deposition model with heat transfer model for our numerical study. Thus, our model is only valid for laminar flow.

To summarize, the key elements of Singh's model are given as below,

Aging rate:

$$\frac{d\left(\pi(R^2 - r_i^2)\rho_{gel}F_w L\right)}{dt} = 2\pi r_i L k_l \left(C_{wb} - C_{ws}(T_i)\right) \qquad \text{Eq. 2-3}$$

Growth rate:

$$-2\pi r_i F_w \rho_{gel} \frac{dr_i}{dt} = 2\pi r_i k_l \left[C_{wb} - C_{ws}(T_i)\right] - 2\pi r_i \left(-D_e \frac{dC_{ws}}{dr}\bigg|_i\right) \qquad \text{Eq. 2-4}$$

The wax content of the oil is calculated by Eq. 2-5.

$$C_{wb} = C_{wbo} - \int_0^L \frac{\pi(R^2 - r_i^2)F_w\rho_{gel}}{V_R} dL \qquad \text{Eq. 2-5}$$

Where R and r_i are radius and effective radius of the flow loop respectively, m; ρ_{gel} is wax-oil gel density, kg/m^3; F_w is weight fraction of solid wax in the gel; L is the length of flow loop, m; C_{wb}, C_{wbo} are bulk and initial bulk concentration of wax, kg/m^3; C_{ws} is solubility of wax in the oil solvent, kg/m^3; T_i is interfacial temperature, ˚C; k_l is mass transfer coefficient, m/s; De is effective diffusivity of wax inside the gel, m^2/s; V_R is total volume of the closed system, m^3.

Note that experiments are necessary to measure the solubility of wax and effective diffusivity of wax molecules into the gel.

Effects of shear rate and stress on wax deposition in turbulent flow have been studied [24-26]. Lee modified Singh's model for turbulent flow by introducing the turbulent axial velocity and thermal and mass transfer eddy diffusivities [27]. Results presented by Keating prove neglecting the turbulent shear in wax deposition leads to over-predicting the wax deposition rate [28]. Investigations by Hsu show shear stress is a function of temperature, wax concentration in solution, and flow velocity [29]. Singh also presented a way to estimate deposition rate by multiplying wax deposition tendency, calculated from the laboratory flow loop, and heat transfer rate [30]. However, quantification of shear stress effect on wax deposition has been estimated mainly using lab data.

Several commercial software packages are used to estimate wax deposition. OLGA [2] is a transient multiphase simulator, dealing with steady wax deposit process. Wax depositions is calculated by three models, with different correlations for molecular diffusion coefficient [31]. Besides molecular diffusion and shear dispersion, the effect of shear stripping reduction on wax deposition is considered. Labes-Carrier compared the wax deposition calculated by OLGA with data from North sea fields, and the results reflected OLGA overestimated wax deposition for multiphase flow while having reasonable agreement for single-phase flow [32].

University of Tulsa developed a multiphase fluid flow model without considering the effect of shear stripping reduction on wax deposition. Flow patterns and mixture properties are taken into flow behavior calculation. Same correlations are applied as OLGA to the calculation of molecular diffusion coefficient [31]. Based on Tulsa model, Couto gave a simplified model for oil/water flow, in which the mixture solubility and physical properties are expressed as the function of water content [3].

Many methods have been proposed to prevent or remove wax deposition including: 1) changing operation conditions [33]; 2) injecting hot oil, solvent or inhibitor [34-36]; 3) adding heat source [13]; 4) using vacuum insulated tubing (VIT); and 5) pigging the pipelines.

Published models do not provide sufficient details about how to integrate the heat transfer model with the wax deposition model. At this time, there are no example calculations of permafrost/formation temperature distribution from these models incorporating ANS conditions.

3. Modeling Study on Wellbore Heat Transfer Problem

This chapter explains the wellbore heat transfer model development, simplification and related solving procedures. The first section is devoted to the wellbore characterization with emphasis on typical wellbore structures in the permafrost and non-permafrost regions. The second part of this chapter gives the assumptions and premises. The last section develops the model.

Wellbore heat transfer model provides a numerical description of forced convection in the tubing, natural convection and radiation in the casing annulus, and conduction in the formation/permafrost. The permafrost thermal behavior under production conditions and the effect of gas lift on heat transfer are also analyzed. To simplify the numerical computation of the model, we approximate the behavior in the flow string as single-phase flow under a series of steady-state conditions.

3.1 Wellbore Introduction

We study a well located in ANS region. The permafrost around the wellbore is about 1500ft in thickness. A major part of pipeline is inclined. Three gas lift vales are mounted in the wellbore. The wellbore profile is depicted in Figure 3-1. Although the original focus of the study was on the well in permafrost region, the wellbore in the non-permafrost region is considered to extend the application. The discussion on the structures of these two kinds of wellbore follows in the rest of the section.
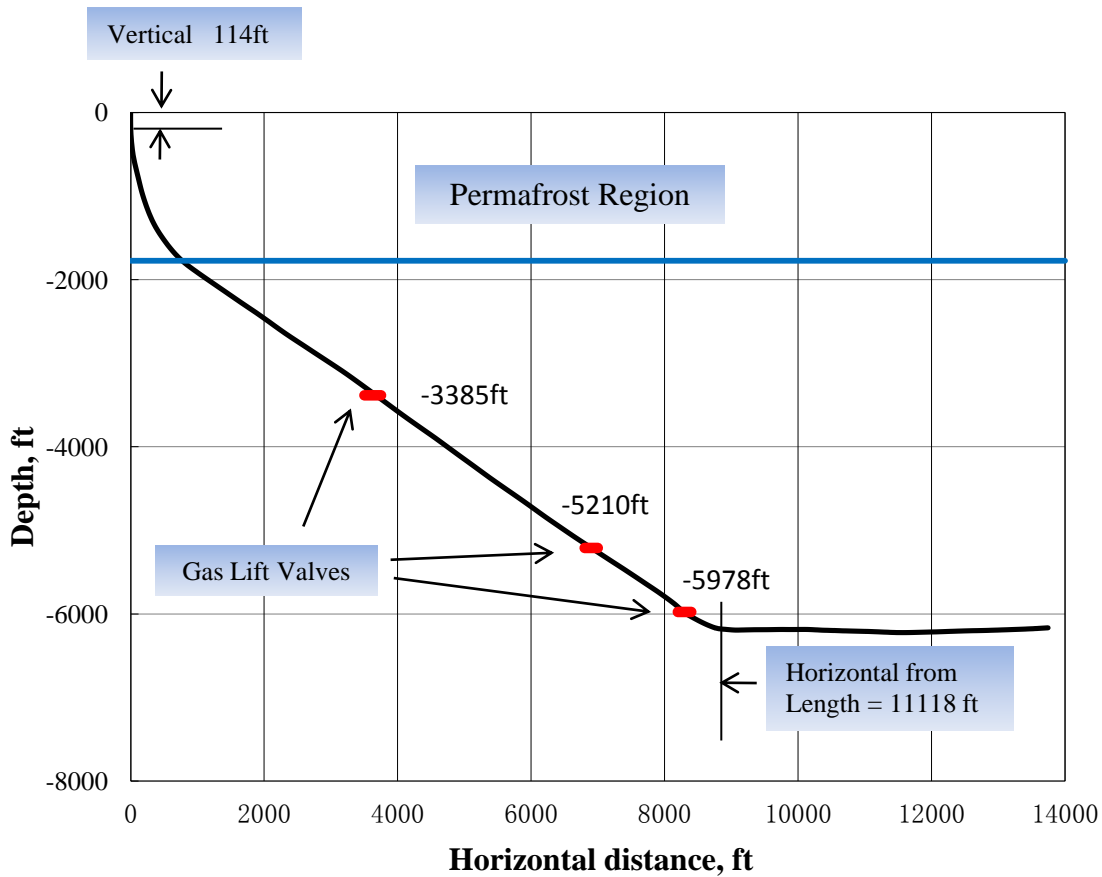
Fig. 3-1 Wellbore profile

Wellbore Schematics in Permafrost Region

The wellbore in the permafrost region has production casing, surface casing, conductor (casing), insulator, and casing around the tubing. The annulus between the tubing and the production casing is filled with gas. The surface casing, conductor, casing are cemented. Drilling mud is in the annulus between the production casing and drillhole. Part of or the whole conductor is protected by steel jacket protection insulator.

In the target well, the production casing is cemented from 11,282 ft to 9727 ft MD while mud is filled in the annulus between the production casing and the drill hole from 9727 ft to surface.
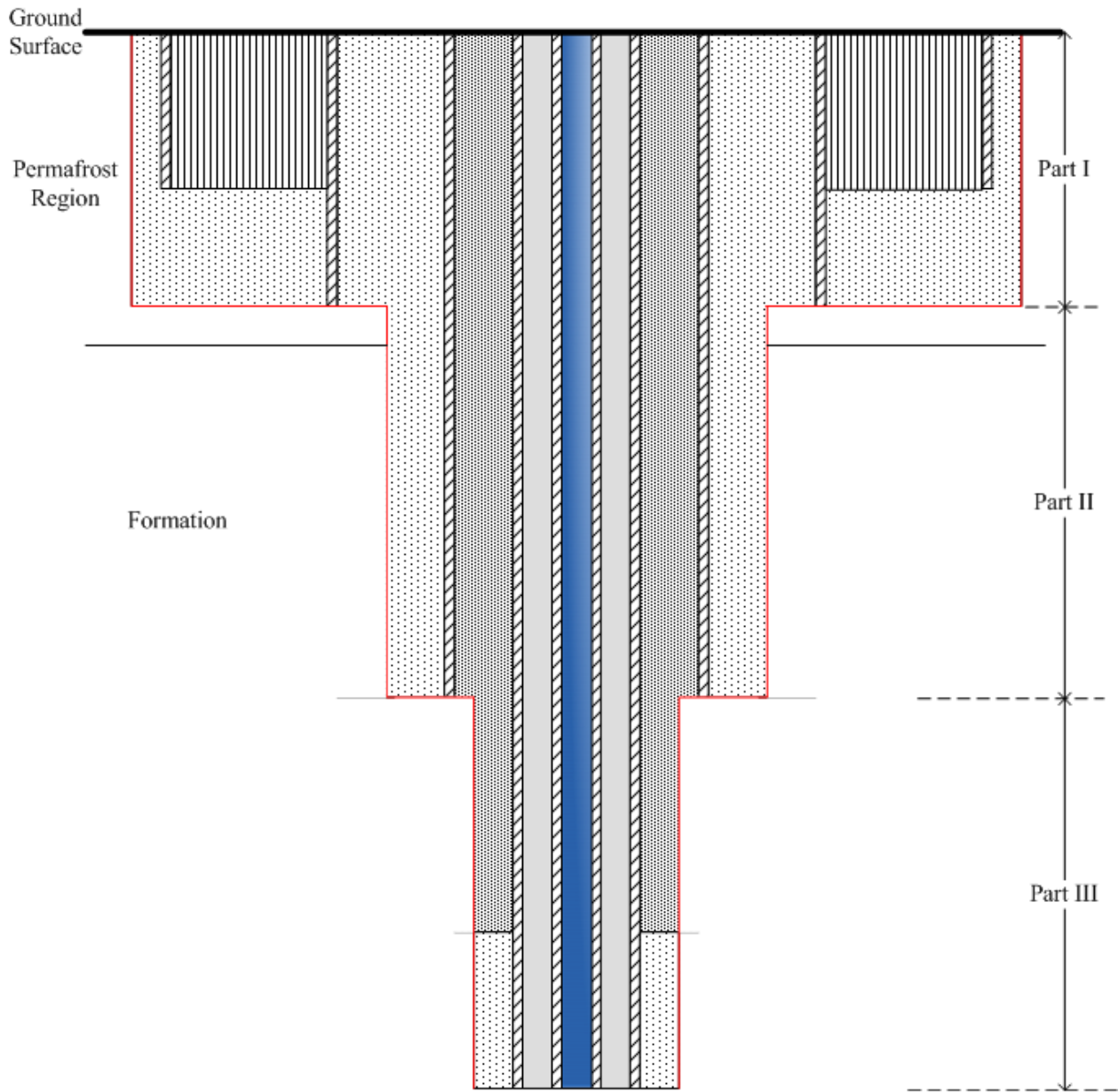
Fig. 3-2 Typical wellbore structure in the permafrost region

	Cement		Drilling mud		Insulation
	Oil		Gas		Drillhole

Steel; From the wellbore center to the outside are tubing, production casing, surface casing, conductor, and casing, respectively.
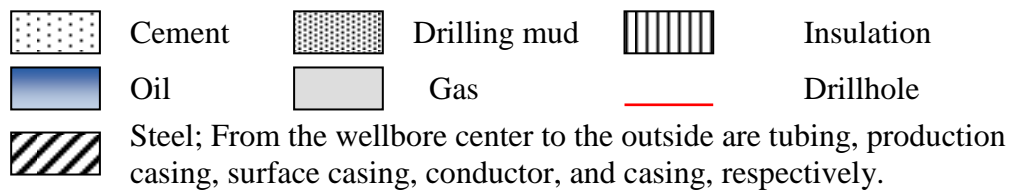
Wellbore Schematics in Non-Permafrost Region

In the non-permafrost region, the production casing and surface casing are around the tubing. The annulus between the tubing and the production casing is filled with gas. The surface casing is cemented. Drilling mud is between the production casing and the drill hole.
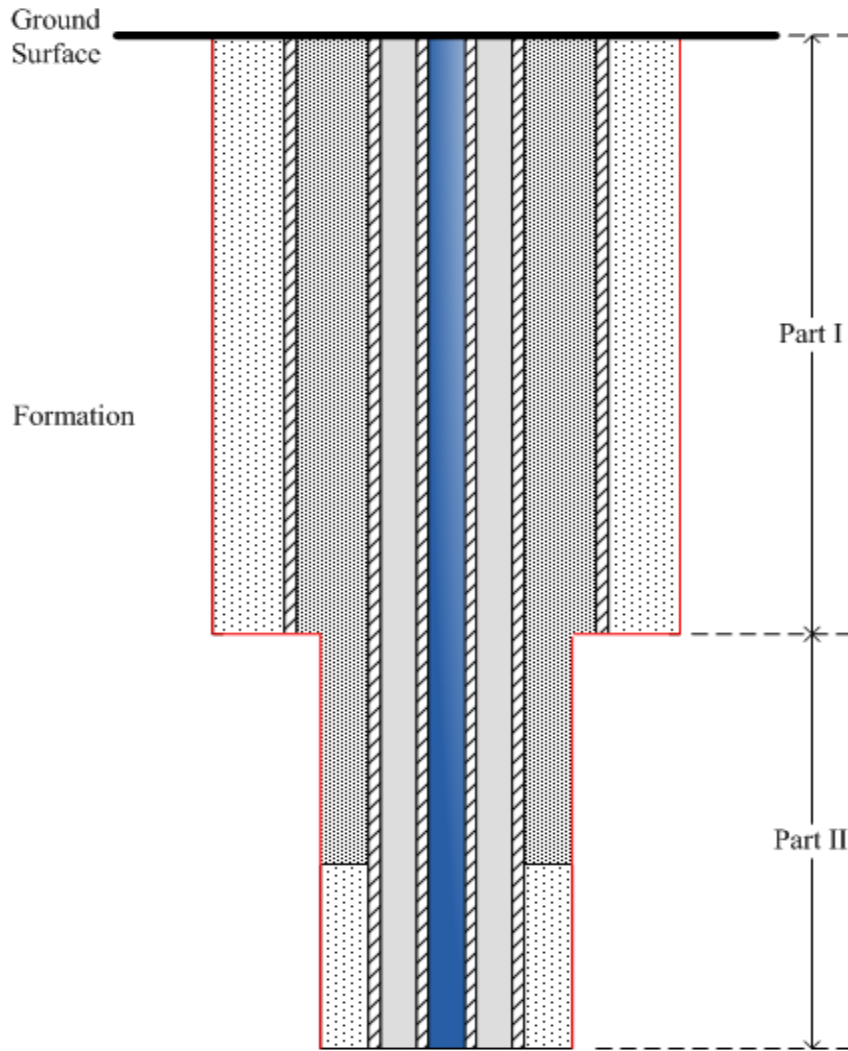
Fig. 3-3 Typical wellbore structure in non-permafrost region

Cement   Drilling mud   Drillhole

Oil   Gas

Steel; From the wellbore center to the outside are tubing, production casing, surface casing, conductor, and casing, respectively.
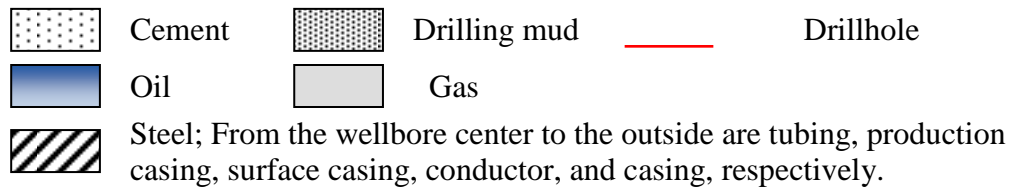
3.2 Model Assumptions

We aim at developing a new mathematical model to predict the temperature profile in a production or injection well, located in permafrost or non-permafrost region. The effects of wax deposit and gas-lift valve on the heat transfer and fluid flow will be evaluated as well. The next chapter will present how to couple the model with wax deposition model developed by Singh.

We categorize the conditions and assumptions of the model as below. Based on the above wellbore structures, following conditions are assumed for the wellbore.

1. Only oil and gas flow in the tubing;
2. Oil and gas are well mixed flow in the tubing, which means the fluid properties are uniform and there is no temperature difference between the oil and gas at cross-sectional area;
3. Heat transfer through the wellbore is in quasi-steady state;
4. Heat transfer from the wellbore to the permafrost/formation is only in radial direction;
5. Kinetic and potential energy are neglected in the heat transfer process.

The assumptions relevant to the formation/permafrost are

1. The formation/permafrost temperature is undisturbed and equal to the geothermal temperature at the location far away from the wellbore;
2. The formation and permafrost are incompressible, the thermal properties of which are invariant over time and depth;
3. In the permafrost, both thawed region and frozen region have the same density but thermal conductivities and heat capacities are different;
4. Heat loss in depth direction can be negligible, and heat is transferred by conduction in the radial direction;
5. No heat source or sink.

3.3 Model Development and Description

This section develops a model to predict pressure and temperature distributions of the in-tubing fluid flow based on the energy balance equations (Section 3.3.3 and Section 3.3.4). As necessary elements in the model computation, fluid properties such as density and viscosity are first reviewed in Section 3.3.1. Section 3.3.2 provides a comprehensive discussion on the boundary conditions that will be adopted to confine the numerical solving process.

3.3.1 Fluid Density and Viscosity

Given the conditions as in Section 3.2, oil and gas in the tubing can be treated as a well-mixed single phase. Hence the density and viscosity of the mixture is averaged and weighted by the mass percentage of each element (Eq. 3-1 and Eq. 3-2).

$$\rho_m = x_g \rho_g + x_o \rho_o \qquad \text{Eq. 3-1}$$

$$\mu_m = x_g \mu_g + x_o \mu_o \qquad \text{Eq. 3-2}$$

Where ρ_o, ρ_g, ρ_m are oil, gas and oil/gas mixture density, respectively, lbm/ft^3; μ_o, μ_g, μ_m are oil, gas and oil/gas mixture viscosity, lb/hr ft; x_o, x_g are mass fractions of gas and oil.

3.3.2 Boundary Conditions

To define the limits of computation, boundary conditions are imposed on the wellbore and around formation/permafrost in the model. The origin is set at the surface of the wellbore. The vertical direction is positive as depth increases.

In the following sub-sections, we first review the constraints on wellbore and those on the formation afterwards.

3.3.2.1 Constraints on the Wellbore

Production well:

Oil flows into the production tubing at reservoir temperature, and the oil temperature remains constant while the oil flows in the horizontal production string. If gas is injected into the casing annulus during the production, the gas temperature at the surface is the same as the air temperature (Eq. 3-3 and Eq. 3-4). Hence,

$$T_f(Bottom,t) = T_{reservoir} \qquad \text{Eq. 3-3}$$

$$T_g(Surface,t) = T_{air} \qquad \text{Eq. 3-4}$$

Where T_f, T_g are fluid and gas temperatures, respectively, °F.

If the well has gas-lift valves, the boundary conditions will have to be established at the location of gas-lift valve. First, energy balance hold at the gas-lift mandrel, as Figure 3-4 shows. Neglect the turbulence around the gas-lift valve caused by gas injection, the energy balance is described by Eq. 3-5.
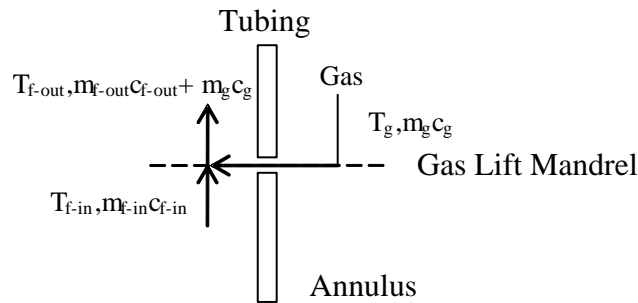
Fig. 3-4 Temperature distribution at gas-lift mandrel

$$\left(m_{m-out}c_{m-out} + m_g c_g\right)T_{f-out} = m_{m-in}c_{m-in} \cdot T_{f-in} + m_g c_g T_g \qquad \text{Eq. 3-5}$$

Where c_m, c_g are specific heat of mixture and gas, respectively, Btu/lb °F; m_m, m_g are mass flow rates of mixture and gas, respectively, lb/hr.

<u>Injection Well:</u>

In this case, at the wellhead, fluid is injected into the tubing at a stable temperature.

$$T_f(Surface, t) = T_{injection}$$ Eq. 3-6

3.3.2.2 The Formation/Permafrost

Model constraints on the region of the formation/permafrost include outer/inner and moving boundaries.

1) Outer Boundary

Figure 3-5 illustrates the cylindrical grid geometry for a single well. On the outer boundary ("r_e"), the formation/permafrost temperature "T_e" is constant and undisturbed by the well production. The geothermal temperature profile in the permafrost is provided by ConocoPhillips Corp., as plotted in Figure 3-6.

Let $T(r, t)$ represent the formation temperature at certain depth, at the outer boundary "r_e"

$$T(r_e, t) = T_e$$ Eq. 3-7
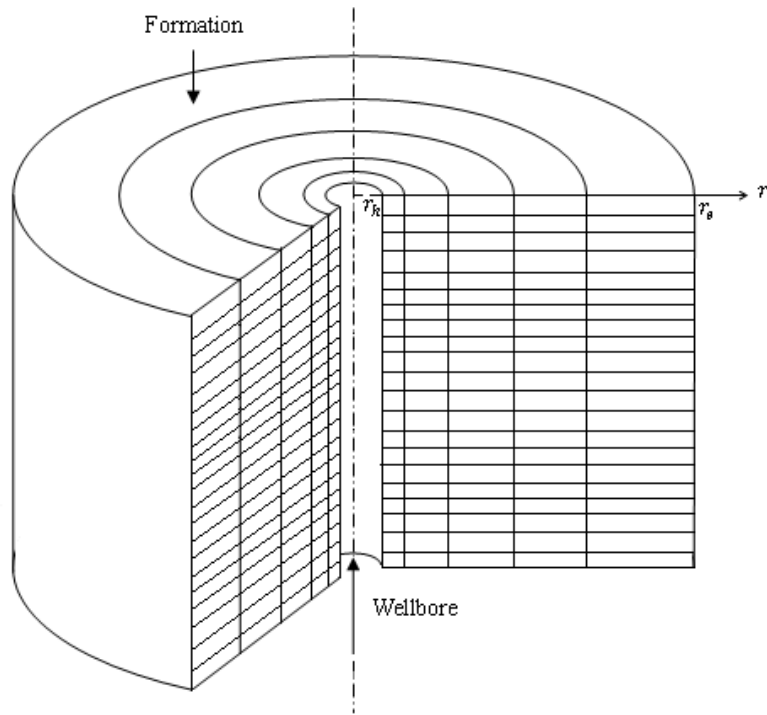
Fig. 3-5 Wellbore in the formation
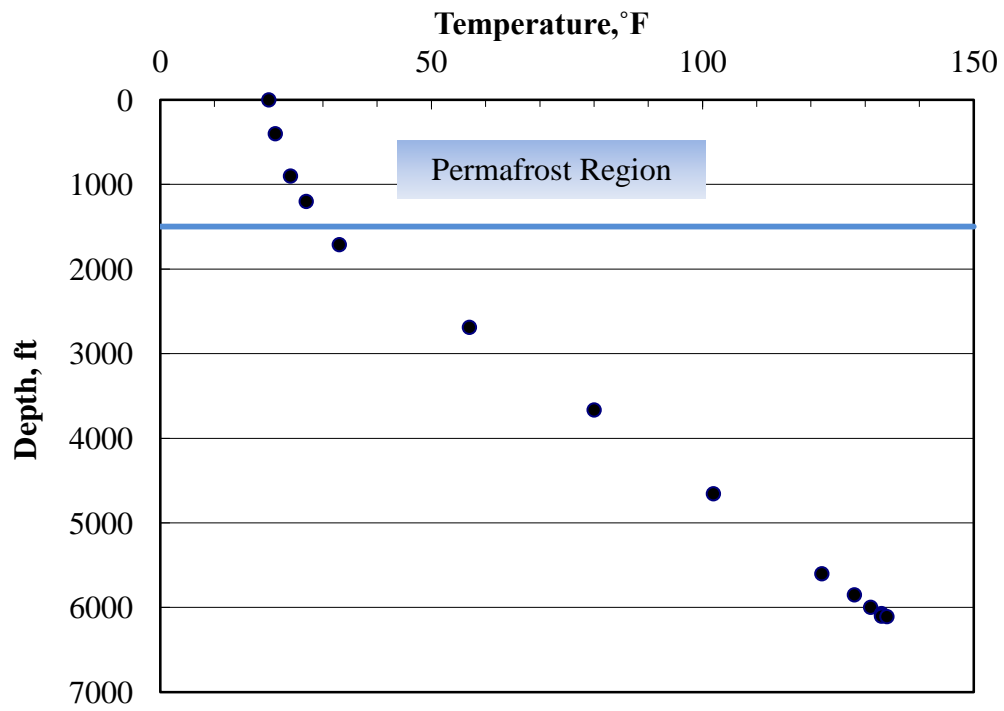
Fig. 3-6 Geothermal temperature "Te"

2) Inner Boundary

On the inner boundary "r_h", where the wellbore/formation interface, the energy balance establishes:

$$2\pi r_{pco} U_{pco}(T_{pco} - T_h) = -2\pi r_h k_e \left(\frac{\partial T}{\partial r}\right)_{r_h}$$

Eq. 3-8

Where U_{pco} is the overall heat transfer coefficient, Btu/hr ft^2 °F,; r_{pco}, r_h are radius of production casing at outside surface and wellbore, respectively, ft; T_{pco}, T_h are temperatures at the outside surface of production casing and wellbore/formation interface, respectively, °F; k_e is thermal conductivity of formation, Btu/hr ft °F.

U_{pco} Calculation

The calculation of U_{pco} depends on the structure difference according to the profile of the drillhole. As the structure shown in Figure 3-2, it includes three parts.

Part I: if the tubing is surrounded by production casing, surface casing, conductor (casing), insulator, and casing, we have
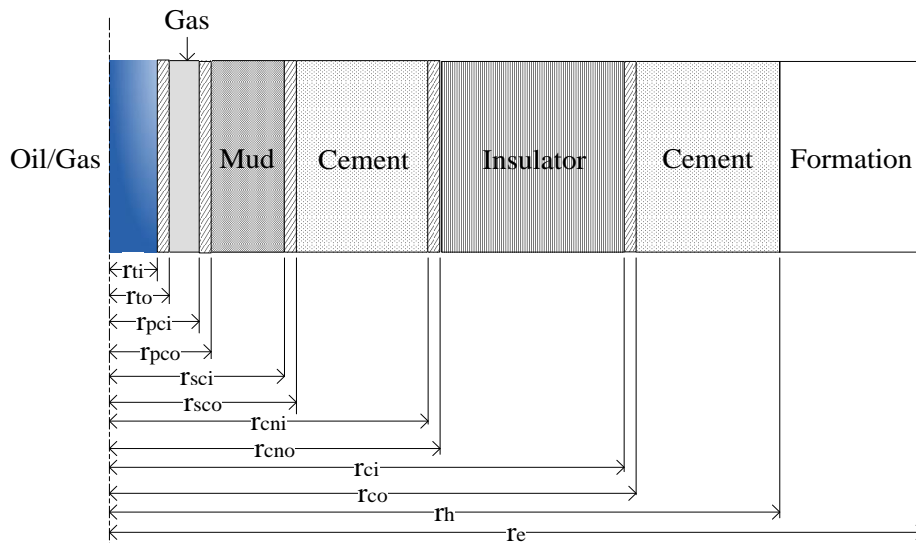
Fig. 3-7 Wellbore structure, part I

$$U_{pco} = \left[\frac{r_{pco}}{k_{mud}} \ln\left(\frac{r_{sci}}{r_{pco}}\right) + \frac{r_{pco}}{k_{cem}} \left[\ln\left(\frac{r_{cni}}{r_{sco}}\right) + \ln\left(\frac{r_h}{r_{co}}\right) \right] + \frac{r_{pco}}{k_{cas}} \left[\ln\left(\frac{r_{sco}}{r_{sci}}\right) + \ln\left(\frac{r_{cno}}{r_{cni}}\right) + \ln\left(\frac{r_{co}}{r_{ci}}\right) \right] + \frac{r_{pco}}{k_{ins}} \ln\left(\frac{r_{ci}}{r_{cno}}\right) \right]^{-1}$$

Eq. 3-9

Where $k_{cem}, k_{cas}, k_{ins}$ are thermal conductivities of cement, casing and insulation, respectively, Btu/hr ft °F; $r_{pci}, r_{sci}, r_{cni}, r_{ci}$ represent inside radius of production casing, surface casing, conductor and casing, respectively, ft; $r_{to}, r_{pco}, r_{sco}, r_{cno}, r_{co}$ are outside radius of tubing, production casing, surface casing, conductor and casing, respectively, ft.

Part II: If the production casing and surface casing are around the tubing, we have
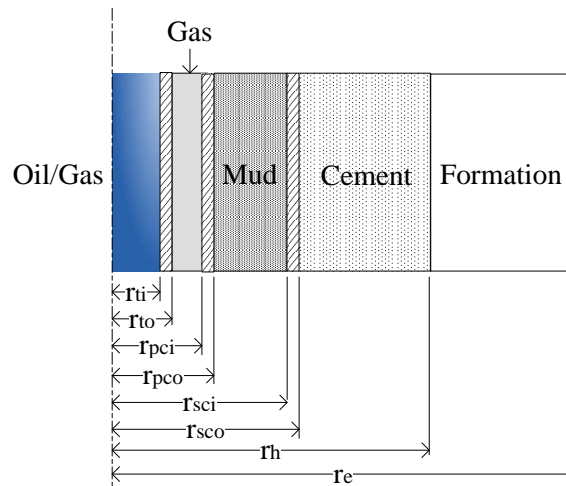
Fig. 3-8 Wellbore structure, part II

$$U_{pco} = \left[\frac{r_{pco}}{k_{mud}} \ln\left(\frac{r_{sci}}{r_{pco}}\right) + \frac{r_{pco}}{k_{cem}} \ln\left(\frac{r_h}{r_{sco}}\right) + \frac{r_{pco}}{k_{cas}} \ln\left(\frac{r_{sco}}{r_{sci}}\right) \right]^{-1}$$

Eq. 3-10

Part III: When there is only production casing around the tubing,
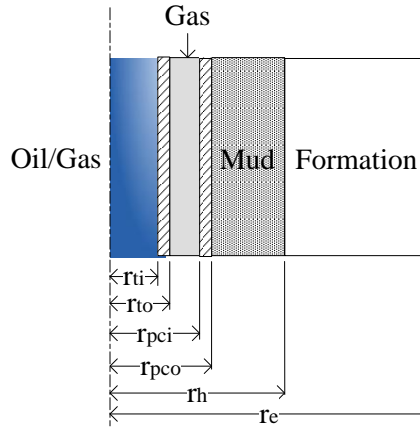
Fig. 3-9 Wellbore structure, part III

$$U_{pco} = \left[\frac{r_{pco} \ln(r_h/r_{pco})}{k_{mud}} \right]^{-1}$$

Eq. 3-11

Note that in non-permafrost region, such as the wellbore structure in Figure 3-3, the overall heat transfer coefficient calculation is similar to above procedures. For the part I in Figure 3-3, the tubing is surrounded by production casing and surface casing, Eq. 3-10 is applied to estimate U_{pco}. When only production casing is around the tubing (part II), Eq. 3-11 is adopted.

3) Moving Boundary

When the permafrost melts, the interface separates the thawed region and frozen region in the permafrost called the moving boundary, as illustrated in Figure 3-10. There are two conditions at the moving boundary "r_{MB}".

First, the heat conservation meets the Stefan boundary condition, defined in Eq. 3-12.

$$k_S \frac{\partial T_S}{\partial r} - k_L \frac{\partial T_L}{\partial r} = L_H \rho \frac{dr_{MB}}{dt}$$

Eq. 3-12

Where k_S and k_L are the thermal conductivity of frozen and thawed regions in permafrost, Btu/hr ft °F; T_S and T_L are temperature of frozen and thawed regions in permafrost, °F; L_H is the latent heat of fusion, Btu/lb; ρ_e is the formation density, lbm/ft³; r_{Df} is the location of the moving boundary from the wellbore center, ft.

29

Second, the temperature satisfies the continuity requirement:

$$T_L(r_{MB}, t) = T_S(r_{MB}, t) = T_w$$ Eq. 3-13

Where T_w is the melting temperature of permafrost, ˚F. In our study, the melting temperature of the permafrost is 32 °F.
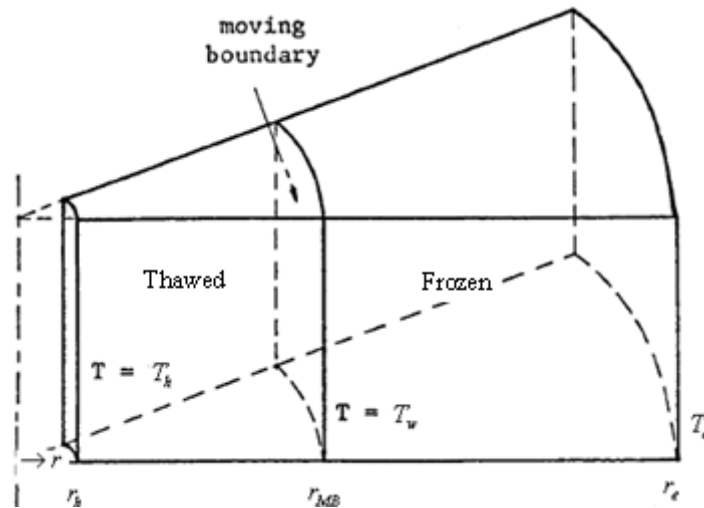
Fig. 3-10 Radial heat conduction with phase change [16]

3.3.3 Energy Balance Equations for Temperature Calculation

3.3.3.1 Model Development

The model aims at predicting the temperatures of tubing fluid, annulus gas and formation/permafrost. To develop a model applicable for both vertical and inclined wells, we calculate the temperature distributions along the well length instead of vertical depth.

When the production well is located in the permafrost region, the direction of mass flow in tubing is negative and the temperature gradient of the fluid is positive. The energy balance on the fluid flowing in the tubing can be expressed by Eq. 3-14.

$$-m_f c_f \frac{dT_f}{dL} = 2\pi r_{ti} h_{ti} (T_f - T_{ti})$$

<div align="right">Eq. 3-14</div>

Where c_f is specific heat of fluid, Btu/lb ˚F; r_{ti} is inside radius of tubing, ft; h_{ti} is heat transfer coefficient based on the inside tubing surface and the temperature difference between the flowing fluid and the surface, Btu/hr ft2 ˚F; T_{ti} is the temperature at inside surface of tubing, ˚F.

Consider the fact that radiation passes through the casing annulus without absorption, and the radiation between the tubing and the casing has no effect on the energy balance for gas. The direction of mass flow in the annulus and the temperature gradient of gas are both positive. So the energy balance on flowing gas is

$$m_g c_g \frac{dT_g}{dL} = 2\pi r_{to} h_{to} (T_{to} - T_g) - 2\pi r_{pci} h_{ci} (T_g - T_{pci})$$

<div align="right">Eq. 3-15</div>

Where h_{to} is heat transfer coefficient for convection based on the outside tubing surface and the temperature difference between the gas in annulus and the surface, Btu/hr ft2 ˚F; h_{ci} is heat transfer coefficient for convection based on the inside casing surface and the temperature difference between the gas in annulus and the surface, Btu/hr ft2 ˚F. T_{to} is the temperature at the outside surface of tubing, ˚F; T_{pci} is temperature at the inside surface of production casing, ˚F.

Neglect the heat transfer in depth direction, energy balance on the formation/permafrost is expressed in Eq. 3-16:

$$k_e \frac{1}{r} \left[\frac{\partial}{\partial r} \left(r \frac{\partial T}{\partial r} \right) \right] = \rho c_e \frac{\partial T}{\partial t}$$

<div align="right">Eq. 3-16</div>

Where k_e is thermal conductivity of formation/permafrost, Btu/hr ft ˚F; c_e is specific heat of formation/permafrost, Btu/lb ˚F; T is formation/permafrost temperature, ˚F.

In summary, Eq. 3-14 ~ Eq. 3-16 can be used to solve the temperature distributions in the wellbore and formation/permafrost. In section 3.3.3.2, the series of equations are simplified to ease the numerical simulation procedures.

3.3.3.2 Model Simplification

At the wellbore/formation interface, Eq. 3-8 can be rewritten into the following form:

$$2\pi r_{pco} U_{pco}(T_{pco} - T_h) = 2\pi r_h K_e \frac{T_h - T_1}{\Delta r}$$

Eq. 3-17

Here, T_1 is temperature of the grid in formation next to the cement/formation interface, °F; Δr is the distance between rh and r1. When the moving boundary stays within the first segment, as shown in Figure 3-11, T1 is replaced by the thawing temperature and $\Delta r = r_{MB} - r_h$. If the moving boundary location passes r_1, then$\Delta T = T_h - T_1$ and $\Delta r = r_1 - r_h$.
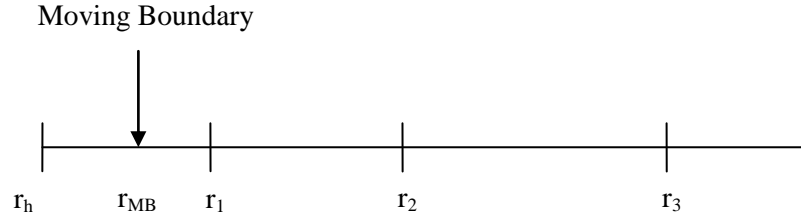
Fig. 3-11 Formation grids distribution

Let $\alpha_1 = \dfrac{r_h k_L}{(r_1 - r_h) r_{pco} U_{pco}}$, the temperature at the interface is:

$$T_h = \frac{T_{pco} + \alpha_1 T_1}{1 + \alpha_1}$$

Eq. 3-18

At the outside surface of the production casing, the energy balance is:

$$2\pi k_{cas} \frac{(T_{pci} - T_{pco})}{\ln\left(r_{pco} / r_{pci}\right)} = 2\pi r_{pco} U_{pco}\left(T_{pco} - T_h\right)$$

Eq. 3-19

Let $\alpha_2 = \dfrac{r_{pco} U_{pco} \ln\left(r_{pco} / r_{pci}\right)}{k_{cas}}$, the temperature at the outside production casing surface is:

$$T_{pco} = \frac{T_{pci} + \alpha_2 T_h}{1 + \alpha_2} \qquad\qquad \text{Eq. 3-20}$$

At the inside casing wall, the following relationship holds:

$$2\pi r_{pci} h_{ci}\left(T_g - T_{pci}\right) + 2\pi r_{to} h_r\left(T_{to} - T_{pci}\right) = 2\pi k_{cas} \frac{\left(T_{pci} - T_{pco}\right)}{\ln\left(r_{pco} / r_{pci}\right)} \qquad\qquad \text{Eq. 3-21}$$

With the expressions of T_h and T_{pco}, the energy balance can be rearranged to obtain the following relationship:

Let $\alpha_3 = \dfrac{r_{pci} U_{pci} \ln\left(r_{pco} / r_{pci}\right)}{k_{cas}}$; $\alpha_4 = \dfrac{r_{to} h_r \ln\left(r_{pco} / r_{pci}\right)}{k_{cas}}$; $\alpha_5 = \dfrac{\alpha_3\left(1 + \alpha_1 + \alpha_1\alpha_2\right)}{\alpha_1\alpha_2 + \left(\alpha_3 + \alpha_4\right)\left(1 + \alpha_1 + \alpha_1\alpha_2\right)}$;

$\alpha_6 = \dfrac{\alpha_4\left(1 + \alpha_1 + \alpha_1\alpha_2\right)}{\alpha_1\alpha_2 + \left(\alpha_3 + \alpha_4\right)\left(1 + \alpha_1 + \alpha_1\alpha_2\right)}$; $\alpha_7 = \dfrac{\alpha_1\alpha_2}{\alpha_1\alpha_2 + \left(\alpha_3 + \alpha_4\right)\left(1 + \alpha_1 + \alpha_1\alpha_2\right)}$

Then,

$$T_{pci} = \alpha_5 T_g + \alpha_6 T_{to} + \alpha_7 T_h \qquad\qquad \text{Eq. 3-22}$$

Also, the energy balance at the inside tubing wall is:

$$2\pi r_{ti} h_{ti}\left(T_f - T_{ti}\right) = 2\pi k_{tub} \frac{\left(T_{ti} - T_{to}\right)}{\ln\left(r_{to} / r_{ti}\right)} \qquad\qquad \text{Eq. 3-23}$$

Let $\alpha_8 = \dfrac{k_{tub}}{r_{ti} h_{ti} \ln\left(r_{to} / r_{ti}\right)}$, the temperature at the inner surface of tubing is:

$$T_{ti} = \frac{T_f + \alpha_8 T_{to}}{1 + \alpha_8} \qquad\qquad \text{Eq. 3-24}$$

In the casing annulus, the heat is lost from the tubing by convection to the gas in the casing annulus and by radiation from the outer tubing surface to the inner casing surface. The relationship is expressed in Eq. 3-25.

$$2\pi k_{tub} \frac{\left(T_{ti} - T_{to}\right)}{\ln\left(r_{to}/r_{ti}\right)} = 2\pi r_{to} h_{to}\left(T_{to} - T_g\right) + 2\pi r_{to} h_r\left(T_{to} - T_{pci}\right)$$

<div align="right">Eq. 3-25</div>

Let $\alpha_9 = \dfrac{r_{to} h_{to} \ln\left(r_{to}/r_{ti}\right)}{k_{tub}}$; $\alpha_{10} = \dfrac{r_{to} h_r \ln\left(r_{to}/r_{ti}\right)}{k_{tub}}$; $\alpha_{11} = 1 + \left[\alpha_9 + \alpha_{10}(1 - \alpha_6)\right](1 + \alpha_8)$

$\alpha_{12} = \dfrac{\alpha_{11} - 1}{\alpha_{11}}$; $\alpha_{13} = \dfrac{(\alpha_9 + \alpha_5 \alpha_{10})(1 + \alpha_8)}{\alpha_{11}}$

Combined with Eq. 3-22 and Eq. 3-24, the temperature at the outside surface of tubing can be obtained as:

$$T_{to} = \frac{1}{\alpha_{11}} T_f + \alpha_{12} T_g + \alpha_{13} T_h$$

<div align="right">Eq. 3-26</div>

We can combine Eq. 3-24 and Eq. 3-26 to get the new expression of T_{ti}, and then plug into Eq. 3-14. The relationship expressed by Eq. 3-14 is now described only using T_f, T_g and T_h:

$$\frac{dT_f}{dL} = AT_f + BT_g + CT_h$$

<div align="right">Eq. 3-27</div>

Where $\alpha_{14} = \dfrac{\alpha_7 \alpha_{10}(1 + \alpha_8)}{\alpha_{11}}$; $\alpha_{17} = -\dfrac{2\pi r_{ti} h_{ti}}{m_f c_f} \cdot \dfrac{\alpha_8}{1 + \alpha_8}$; $A = \alpha_{17}\alpha_{12}$; $B = -\alpha_{17}\alpha_{13}$; $C = -\alpha_{17}\alpha_{14}$

Similar algebraic manipulations and parameter elimination can be followed to Eq. 3-15, and we have Eq. 3-28.

$$\frac{dT_g}{dL} = DT_f + ET_g + FT_h$$

<div align="right">Eq. 3-28</div>

Here,

$\alpha_{18} = -\dfrac{2\pi r_{to} h_{to}}{m_g c_g}$; $\alpha_{19} = -\dfrac{2\pi r_{pci} h_{ci}}{m_g c_g}$; $\alpha_{15} = \alpha_6 \alpha_{14} + \alpha_7$; $\alpha_{16} = 1 - \alpha_5 - \alpha_6 \alpha_{13}$;

$$D = \frac{\alpha_{18} + \alpha_6\alpha_{19}}{\alpha_{11}} \; ; \quad E = \alpha_{18}(\alpha_{13} - 1) - \alpha_{16}\alpha_{19} \; ; \quad F = (\alpha_{18}\alpha_{14} + \alpha_{15}\alpha_{19})$$

With the expressions of T_{to}, T_{pco} and T_{pci}, let

$$a_1 = \frac{\alpha_6}{\alpha_{11}} \; ; \quad a_2 = \alpha_5 + \alpha_6\alpha_{13} \; ; \quad a_3 = 1 + \alpha_1 + \alpha_1\alpha_2 \; ; \quad a_4 = \frac{\alpha_1}{1+\alpha_1}(1 + \alpha_1 + \alpha_1\alpha_2) + (\alpha_7 + \alpha_6\alpha_{14})$$

Eq. 3-17 is rearranged into the following equation:

$$a_1 T_f + a_2 T_g + a_3 T_h + a_4 T_1 = 0 \qquad \text{Eq. 3-29}$$

Let $\lambda = \dfrac{k_e}{\rho_e c_e}$, Eq. 3-16 is simplified as:

$$\frac{1}{r}\frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial r} = \frac{1}{\lambda}\frac{\partial T}{\partial t} \qquad \text{Eq. 3-30}$$

Eq. 3-30 is valid for formation, thawed and frozen regions in permafrost, each with different thermal properties.

In conclusion, the heat transfer model is further simplified as Eq. 3-27 ~ Eq. 3-30. The next section discusses the solving procedure.

3.3.4 Energy Balance Equations for Pressure Calculation

For the incompressible single-phase flow without shaft work device, the over-all pressure drop in the wellbore results from the potential energy "ΔP_{PE}", kinetic energy "ΔP_{KE}", and frictional loss "ΔP_F" [37], as Eq. 3-31.

$$\Delta P = \Delta P_{PE} + \Delta P_{KE} + \Delta P_F \qquad \text{Eq. 3-31}$$

Pressure drop due to the potential energy change is calculated by Eq. 3-32:

$$\Delta P_{PE} = \frac{g}{g_c}\rho_f \Delta z \qquad \text{Eq. 3-32}$$

Where g is gravity acceleration, ft/s^2; g_c is conversion factor, 32.17 lb$_m$-ft/lb$_f$ s^2.

Kinetic energy pressure drop is caused by the diameter change, which can be calculated using the following equation.

$$\Delta P_{KE} = \frac{8\rho_f q^2}{\pi^2 g_c}\left(\frac{1}{D_2^4} - \frac{1}{D_1^4}\right)$$

Eq. 3-33

For oilfield units of bbl/d for flow rate, lb_m/ft^3 for density, and in. for diameter, Eq. 3-33 can be converted into the following form.

$$\Delta P_{KE} = 1.53\times10^{-8}\rho_f q^2\left(\frac{1}{D_2^4} - \frac{1}{D_1^4}\right)$$

Eq. 3-34

Where $D_1 \ and \ D_2$ are respectively inner diameters of flows in and out.

The pressure drop caused by friction is shown by Eq. 3-35.

$$\Delta P_F = \frac{2f_f\rho_f u^2 L}{g_c D}$$

Eq. 3-35

Here u is fluid velocity, ft/s; f_f is Fanning friction factor.

For Laminar flow: $f_f = \frac{16}{Re}$ $(Re \leq 2300)$

Turbulent flow [38]: $\frac{1}{\sqrt{f_f}} = -4\log\left\{\frac{\varepsilon}{3.7065} - \frac{5.0452}{Re}\log\left[\frac{\varepsilon^{1.1098}}{2.8257} + \left(\frac{7.149}{Re}\right)^{0.8981}\right]\right\}$ $(Re > 2300)$

In the above equation, ε is the relative pipe roughness, dimensionless.

We assume the reservoir is infinite and the inflow performance relationship of the reservoir can be expressed in Eq. 3-36 [37].

$$P_{wf} = P_i - \frac{162.6qB\mu}{kh}\left(\log t + \log\frac{k}{\phi\mu c_t r_w^2} - 3.23\right)$$

Eq. 3-36

Here P_i, P_{wf} are initial reservoir pressure and bottomhole pressure, respectively, psi; B is formation volume factor, bbl/STB; μ is fluid viscosity, cp; k is permeability, md; h is

36

reservoir thickness, ft; ϕ is porosity, faction; c_t is total system compressibility, psi^{-1}; r_w is wellbore radius, ft.

Given the reservoir properties as listed in Appendix A, the relationship between q and P_{wf} depends on time. Eq. 3-37 is used to calculate the bottomhole pressure when the well is flowing at a constant rate q, or the flow rate under constant bottomhole pressure.

$$P_{wf} = 5651 - 0.82 \cdot q \cdot \left(\log t + 4.39 \right) \qquad \text{Eq. 3-37}$$

3.3.5 Numerical Calculation Procedures

The model simulation includes data initialization, pressure prediction, temperature calculation and wax deposition estimation which are plotted in Figure 3-12. Wax deposition calculation will be introduced in chapter IV.

Data initialization deals with the preparation of production/injection information, gas and liquid thermal properties, WAT, wellbore configuration and thermal properties, formation and permafrost thermal properties.

The pressure calculation checks if the reservoir energy is large enough to produce the oil at the constant flowrate, and estimates the maximum production rate under constant surface or bottomhole pressure condition. The pressure is calculated using Eq. 3-37 at each time step.
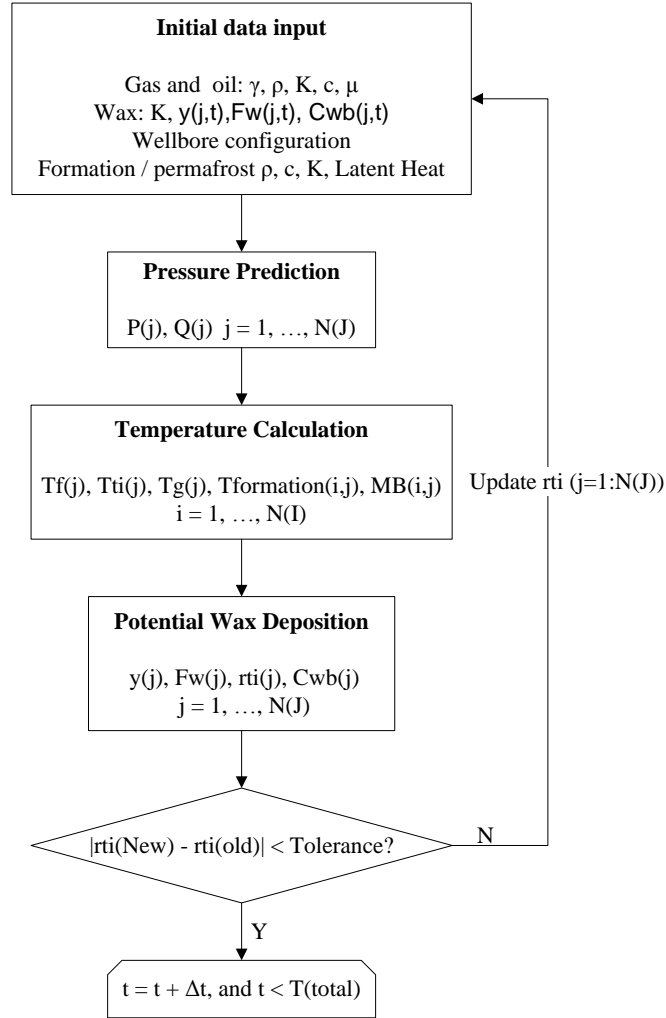
Fig. 3-12 Flow chart of simulation

Temperature calculation follows the pressure check. Finite-difference method is adopted to solve Eq. 3-27 ~ Eq. 3-30. Figure 3-13 shows numbering of the grid system illustrated in Figure 3-5. Cylindrical grid system is used for the formation/permafrost in the simulation. The grids are spaced logarithmically away from the wellbore, and the relationship of grid size is defined by Eq. 3-38 [39].

$$r_j = a_{\lg}^j r_h$$

Eq. 3-38

Here $a_{\lg} = (r_e / r_h)^{1/N_j}$ N_j is the total number of gridblocks in the r direction.
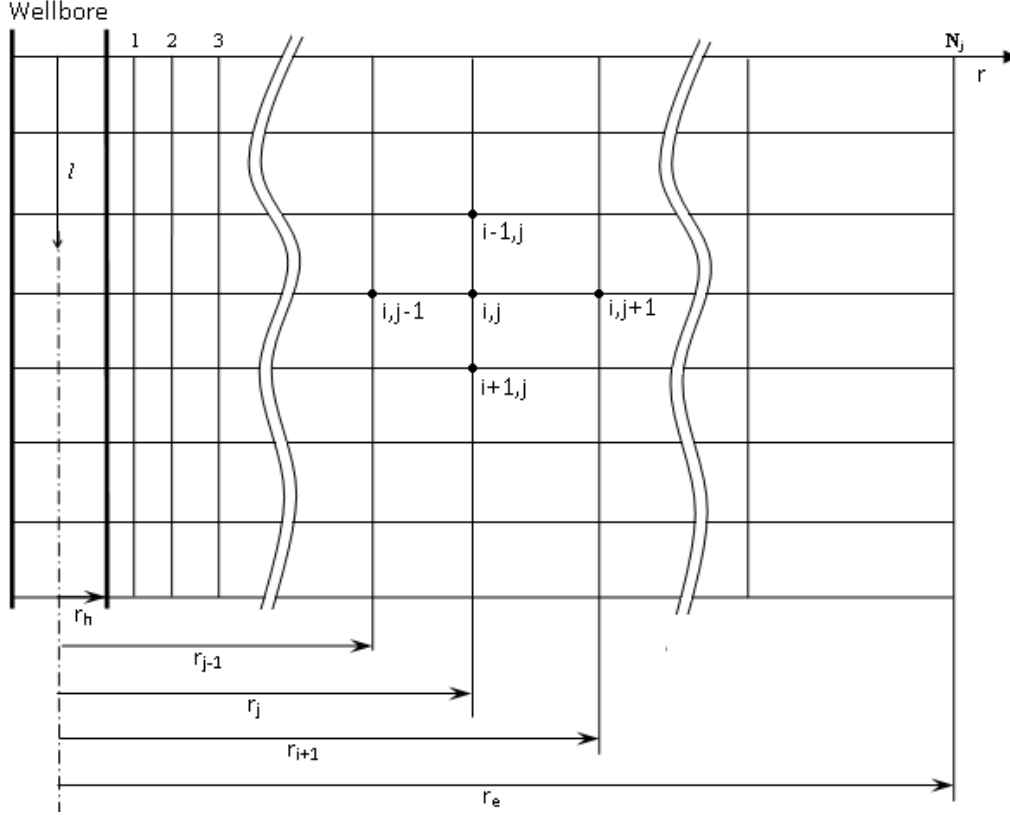
Fig. 3-13 Numbering of cylindrical grid system

Finite difference discretization of Eq. 3-27 ~ Eq. 3-30 are re-formatted as:

$$\frac{T_{f,i}^{n+1} - T_{f,i}^{n}}{\Delta l} = AT_{f,i}^{n+1} + BT_{g,i}^{n+1} + CT_{h,i}^{n+1}$$

Eq. 3-39

$$\frac{T_{g,i}^{n+1} - T_{g,i}^{n+1}}{\Delta l} = DT_{f,i}^{n+1} + ET_{g,i}^{n+1} + FT_{h,i}^{n+1}$$

Eq. 3-40

$$a_1 T_{f,i}^{n+1} + a_2 T_{g,i}^{n+1} + a_3 T_{h,i}^{n+1} + a_4 T_{1,i}^{n+1} = 0$$

Eq. 3-41

$$\left[\frac{2}{(\Delta r_{j+1} + \Delta r_j)\Delta r_j} - \frac{1}{(\Delta r_{j+1} + \Delta r_j)r_j} \right] T_{i,j-1}^{n+1} - \left[\frac{2}{(\Delta r_{j+1} + \Delta r_j)\Delta r_{j+1}} + \frac{2}{(\Delta r_{j+1} + \Delta r_j)\Delta r_j} + \frac{\rho c}{\tau} \right] T_{i,j}^{n+1} + \left[\frac{2}{(\Delta r_{j+1} + \Delta r_j)\Delta r_{j+1}} + \frac{1}{(\Delta r_{j+1} + \Delta r_j)r_j} \right] T_{i,j+1}^{n+1}$$

Eq. 3-42

$$= -\frac{\lambda}{\tau} T_{i,j}^{n}$$

Here $\Delta r_j = r_j - r_{j-1}, \Delta r_{j+1} = r_{j+1} - r_j$

Eq.3- 42 represents a series of equations at each grid in the formation/permafrost. When the wellbore is located in permafrost region, the moving boundary location needs to be estimated with the temperature results and determine if the permafrost in the vicinity of the well stars to defrost. The solving procedure of temperature calculation is shown by Figure 3-14.
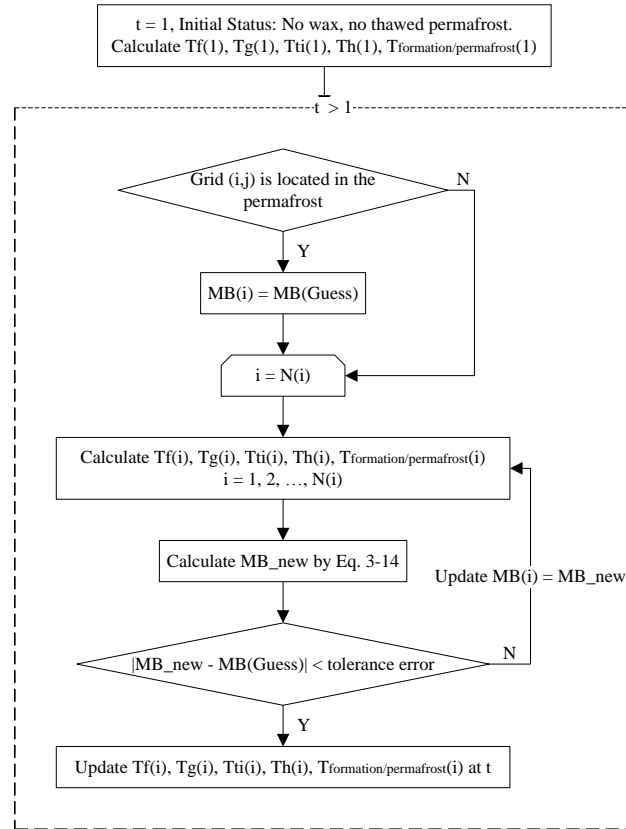
Fig. 3-14 Flow chart of temperature calculation

The initial wellbore temperature distributions are calculated at "$t = 0$". At the beginning, we assume no permafrost thawed and stationary moving boundary at wellbore/formation interface, that is, $r_{f_0} = r_h$, $u_{f_0} = 0$. Oil, gas and formation/permafrost temperatures are then calculated.

If the grid is in the permafrost, assume the moving boundary stays at $r_{f_1}^{(0)}$ after one time step $\triangle$t. The moving front velocity is obtained as $u_{f_1}^{(0)} = r_{f_1}^{(0)} / \Delta t$. Solve Eq. 3-39 ~ Eq. 3-42 with

40

boundary conditions introduced in the section 3.3.2 to update the temperature distributions in the wellbore and the formation/permafrost at "$t = t + \Delta t$".

A new moving front velocity $u_{f_1}^{(1)}$ is calculated according to Eq. 3-43 with the temperature gradients at the location $r_{f_1}^{(0)}$ in both thawed and frozen regions. Figure 3-15 shows an example of temperature gradients at the moving boundary. In Figure 3-15, line "1" represents the temperature gradient "$\partial T_L / \partial r$" in the thawed region and line "2" is that "$\partial T_S / \partial r$" in the frozen region.

$$u_{f_1}^{(1)} = \frac{1}{L_H \rho} \left[k_S \frac{\partial T_S}{\partial r} - k_L \frac{\partial T_L}{\partial r} \right]$$

Eq. 3-43

With $u_{f_1}^{(1)}$, we have the new location of the moving front after $\triangle$t,

$$r_{f_1}^{(1)} = r_{f_0} + \Delta t \cdot \left(u_{f_1}^{(0)} + u_{f_1}^{(1)} \right) / 2$$

Eq. 3-44
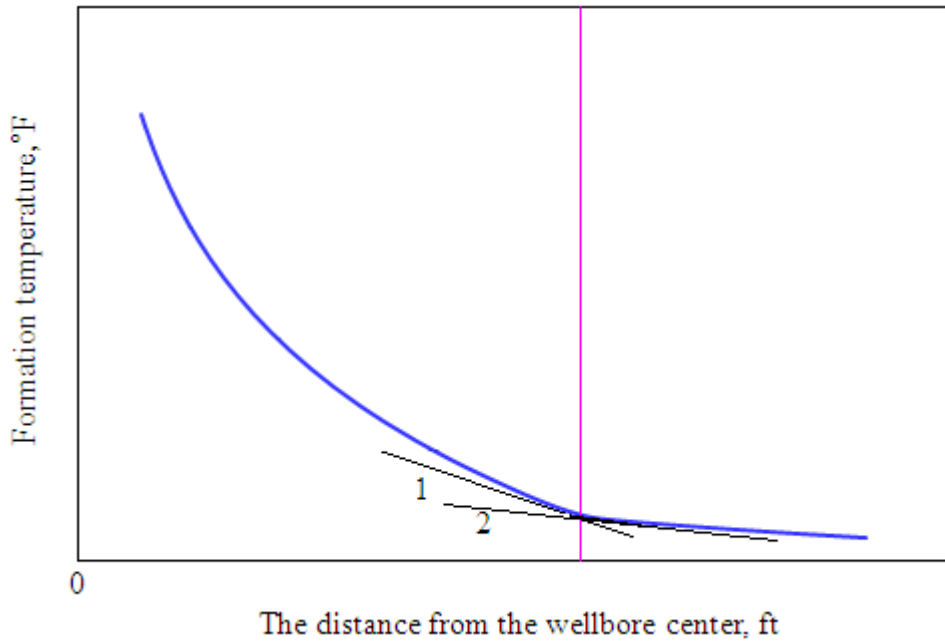
Fig. 3-15 Temperature gradients at the moving boundary

The simulation will compare $r_{f_1}^{(0)}$ and $r_{f_1}^{(1)}$. If the difference between the two values is beyond the tolerance error, $r_{f_1}^{(0)}$ is replaced by $r_{f_1}^{(1)}$ as the moving boundary location after $\triangle$t. Recalculate the temperature distribution using Eq. 3-39 ~ Eq. 3-42.

This process is iterated until the difference between two adjacent iteration times of t_1 is less than the error limit. $\Delta r_{f_1} = r_{f_1}^{(n)} - r_{f_0}$ is used to update the moving boundary location at next time step. After the position of the moving boundary is determined, the entire procedure is repeated to obtain the solution of the temperatures (fluid, gas and formation/permafrost) and so on for the next time instant. Note, at t_{n+1}, the initial values of $r_{f_{n+1}}^{(0)}$ and $v_{f_{n+1}}$ are estimated by:

$$r_{f_{n+1}} = r_{f_n} + \Delta r_{f_n} \quad \text{and} \quad v_{f_{n+1}}^{(0)} = \left(r_{f_n} - r_{f_{n-1}} \right)/\Delta t .$$

If injection well or the production well doesn't have GLV, the gas temperature can be estimated by averaging the outer tubing surface temperature and the inner production casing surface temperature.

4. Wax Deposition Calculation

4.1 Introduction

In this chapter, we discuss the calculation of wax deposition. Singh's model was used to predict the wax thickness and wax fraction in the deposition. A mathematical model was developed to simulate the variation of wax content in the solution over time. Since wax deposition is strongly related to the thermal behavior of the fluid flowing under production conditions, the calculation of wax deposition is coupled with heat transfer calculation.

4.1.1 Singh's Model for Wax Deposit Calculation

Wax starts to precipitate when the temperature of the flowing oil is at or less than WAT. According to Singh's model, the driving force leading to the wax deposition is the temperature gradient along the cross-sectional area. During the process of wax build-up, the temperature at oil/wax interface increases and the effective inner radius of tubing reduces. Wax only stops growing when the temperature at oil/wax interface is higher than WAT or the temperature gradient along the cross-sectional area vanishes.

The model is reviewed here based on Singh's research [21]. Aging rate and growth rate of wax deposition are calculated by the following equations with the definition of dimensionless thickness y, $y = 1 - r_i / R$.

Aging rate dF_w/dt:

$$\frac{y}{1-y}\left(1-\frac{y}{2}\right)\frac{dF_w}{dt} = \frac{D_e}{R\rho_{gel}}\frac{dC_{ws}}{dT}\frac{dT}{dy}\bigg|_i \qquad \text{Eq. 4-1}$$

Growth rate dy/dt:

$$\frac{dy}{dt} = \frac{k_l}{F_w\rho_{gel}R}\left[C_{wb} - C_{ws}(T_i)\right] + \frac{D_e}{F_w\rho_{gel}R}\frac{dC_{ws}}{dT}\frac{dT}{dy}\bigg|_i \qquad \text{Eq. 4-2}$$

Eqs. 4-1 and 4-2 can be solved numerically by using Runge-Kutta algorithms for the wax thickness and wax weight fraction in the deposition over time.

The following equations are used to calculate parameters mentioned in Eq. 4-1 and Eq. 4-2, such as wax solubility, oil/wax interfacial temperature, deposition thermal conductivity and wax effective diffusivity.

The solubility of wax in the oil solvent:

$$C_{ws}(T_i) = a(T_i + b)^c$$

Eq. 4-3

In simulation, a, b, c took 4.9×10^{-9} kg/m^{3}K^6, 17.8 ℃, and 6 respectively [21], which were obtained from the sample oil in Singh's experiments.

The temperature at oil/wax interface can be derived from the energy balance:

$$T_i = \frac{r_i h_{ti} T_f + \dfrac{k_{dep}}{\ln(R/r_i)} T_{wall}}{\dfrac{k_{dep}}{\ln(R/r_i)} + r_i h_{ti}}$$

Eq. 4-4

Where T_{wall} represents the tubing temperature.

The temperature gradient at the oil/gel interface is:

$$\left. \frac{dT}{dy} \right|_i = \frac{T_i - T_{wall}}{R(1-y)\ln(1-y)}$$

Eq. 4-5

The thermal conductivity of wax deposit is

$$k_{dep} = \frac{2k_{wax} + k_{oil} + (k_{wax} - k_{oil})F_w}{2k_{wax} + k_{oil} - 2(k_{wax} - k_{oil})F_w} k_{oil}$$

Eq. 4-6

where $k_{wax}, k_{oil}, k_{dep}$ are thermal conductivity of wax, oil and wax deposition, respectively, W/m/K.

44

The effective diffusivity of wax molecules into the gel:

$$D_e = \frac{D_{wo}}{1 + \alpha^2 F_w^2 / (1 - F_w)}$$

Eq. 4-7

Here, D_{wo} is the molecular diffusivity of wax in oil, m^2/s; average aspect ratio of the wax crystal α is 8 [21].

4.1.2 Mass Balance for Wax Content Prediction

The calculation of wax concentration introduced in Singh's model was derived from the mass balance for a closed flow loop, which may not be valid for the wellbore. A new solution to calculate wax content is introduced here.

Figure 4-1 shows the change of wax content in the solution. The bulk concentration of wax C_{wb} changes with time and location.
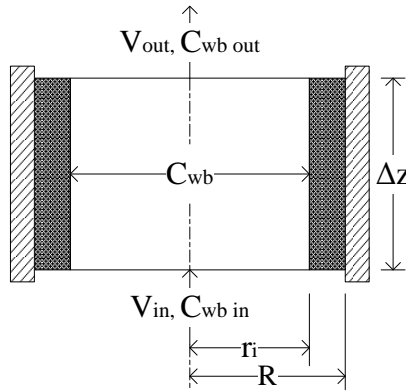
Fig. 4-1 Sketch of the change of wax content

▨▨▨ Steel tubing   ▨▨▨ Wax deposition

Assumptions

1. Thermal properties of wax-gel are uniform along the pipe length direction;
2. At adjacent time steps, the change of control volume due to wax deposition is negligible, that

45

is, $\Delta V \approx 0$ within Δt.

Initial and Boundary Conditions

At time t = 0:

$$C_{wb}(z,t) = C_{wb}(z,0) = C_{wbo}$$

The wax starts to grow at the location "Z", where the interfacial temperature is equal to WAT

$$C_{wb}(Z,t) = C_{wbo}$$

At the wellhead:

$$\partial C_{wb} / \partial z = 0$$

Mass Balance for Wax Content

Mass balance holds on the wax, the control volume shown in Figure 4-1 is $\pi r^2 \Delta z$.

Mass of wax change during Δt

$$= \text{Mass of wax change by bulk flow} \quad - \quad \text{Mass of wax change in wax deposition}$$

$$\Delta C_{wb} \cdot \pi r^2 \Delta z = \pi r^2 \cdot u C_{wb}\,|_z \cdot \Delta t - \pi r^2 \cdot u C_{wb}\,|_{z+\Delta z} \cdot \Delta t - \left[\pi\left(R^2 - r^2\right)\Delta z \rho_{gel} \cdot F_w\,|_{t+\Delta t} - \pi\left(R^2 - r^2\right)\Delta z \rho_{gel} \cdot F_w\,|_t \right] \qquad \text{Eq. 4-8}$$

Divided by $\pi r^2 \Delta z \Delta t$ on both sides of Eq. 4-8,

$$\frac{\partial C_{wb}}{\partial t} = \frac{\partial\left(u C_{wb}\right)}{\partial z} - \rho_{gel} \frac{\left(R^2 - r_i^2\right)}{r_i^2} \frac{\partial F_w}{\partial t} \qquad \text{Eq. 4-9}$$

Eq. 4-9 was adopted to compute the wax concentration in the tubing fluid and the numerical calculation procedures are presented in Section 4.1.3.

4.1.3 Procedures for Wax Deposit Calculation

Detail of the calculation procedure of Singh's model can be found in Wu's research [40]. Here, we discuss how to couple the calculation of wax deposition with temperature calculation. We first start by computing oil temperature distribution along the well length under the condition of no wax deposition. Then compare the oil temperature at inner tubing surface with WAT from the

46

well bottom to the surface until the first location j where wax starts to deposit is found. Wax calculation will be performed on all the points from location j to the surface.

The flowchart of the simulation is shown in Figure 4-2. Note that the wax concentration along the well length is equal to C_{wbo} at the initial status.
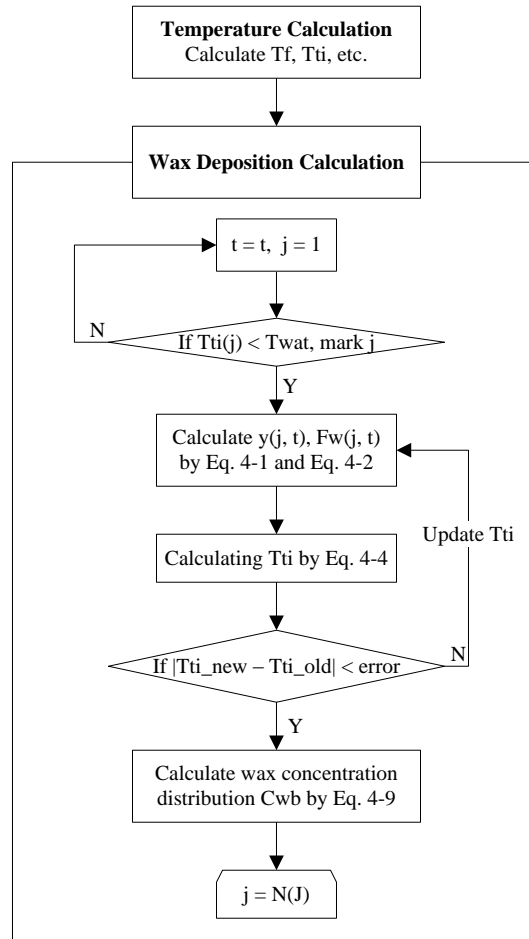
Fig. 4-2 Flowchart for wax deposition calculation

It is assumed there is a very thin slim of wax with thickness y_{t_0} and wax concentration $F_{w_{t_0}}$ deposited on the tubing inner surface at the beginning. We also assume the temperature at oil/wax interface $(T_i)_{t_1}^{(0)}$ is equal to the temperature at the inner tubing surface after one time

step Δt. The wax thickness $y_{t_1}{}^{(1)}$ and wax concentration in the deposition $F_{w_{t_1}}{}^{(1)}$ are calculated following Eq. 4-1 and Eq. 4-2.

Then, the interfacial temperature $(T_i)_{t_1}{}^{(1)}$ can be estimated by Eq. 4-4. Compare $(T_i)_{t_1}{}^{(0)}$ with $(T_i)_{t_1}{}^{(1)}$, if the difference is beyond the tolerance error, update $(T_i)_{t_1}{}^{(0)}$ by $(T_i)_{t_1}{}^{(1)}$ and recalculate $y_{t_1}{}^{(2)}$ and $F_{w_{t_1}}{}^{(2)}$. This process is repeated until the difference of $(T_i)_{t_1}$ between two adjacent iteration times is less than the error limit.

Repeat above procedures to get wax thickness and concentration for all points from location j to the surface. In the end, we predict $(C_{wb})_{t_1}$ distribution according to Eq. 4-9.

After the wax deposition calculation, the simulation updates the effective tubing inside diameter before calculating the pressure and heat transfer for the next time instant.

5. Graphic User Interface User Guide

The software pack TempCal is developed to calculate and visualize the temperature, pressure and wax deposition of typical well configurations. The design of the software is divided into several modules e.x. a user interface designed by MS Visual Basic; a back-end computation engine implemented in MATLAB and data exchange mechanism enabled by MS ActiveX technology.

Functionally, the software package does the numerical analysis in pressure/temperature and wax deposition while taking into account following conditions,

(1) Production well or injection well;

(2) Vertical well or inclined well;

(3) Gas lift during the operation or not;

(4) The well is located in the permafrost region or not;

(5) Wax deposition is considered or not.

This chapter is an introduction on user guide of TempCal involving how to input the parameters to define the well computation; how to interpret the output and results returned by MATLAB. The chapter, as a user guide, can serve as a reference for anyone who is interested in the numerical study on wellbore conditions. In what follows, the chapter is organized into two parts on software requirements: parameters input; results output and visualization. Command and operation procedure will be offered for the convenience of the readers to verify the functions.

5.1 Software setup and startup

5.1.1 Software requirements

Before starting up the software, there are several software tools to be installed. A recommended list is provided as below.

1) Microsoft Windows XP with service pack 3.0;

2) Microsoft Visual Basic 6.0

3) Mathworks MATLAB student version 2011

4) Dplot Jr. free version

5.1.2 General operating flows

1) Creating a new calculation

In the main window (see figure below), first select "File" in the menu bar.
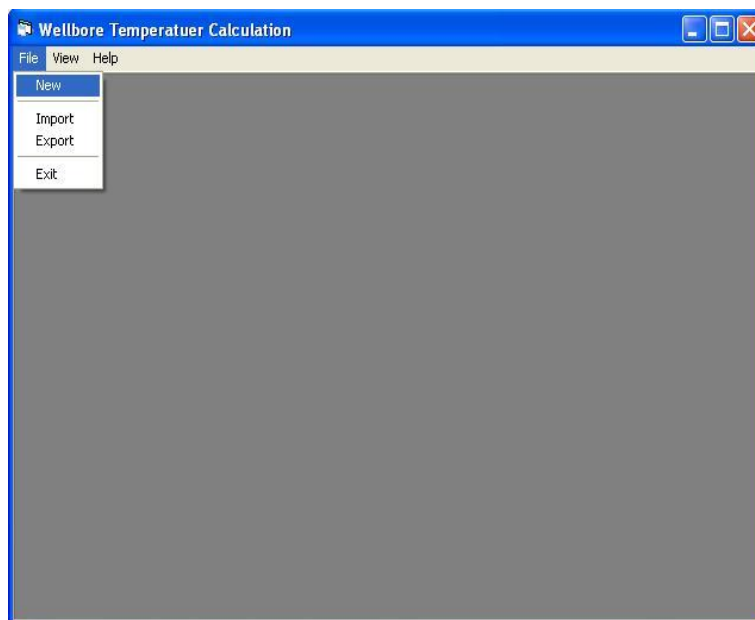
After a drop-down menu with four options pops up, right-click "New" to start a new calculation. A pop-up message box will be displayed if "New" is selected. Click "Yes" to start parameter input and "No" otherwise.
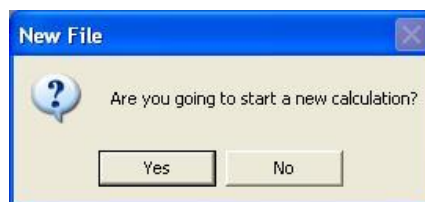
2) Export and Import a configuration file

The software pack also provides "Import" and "Export" functions designed to simplify the data input and saving. When finishing the input of all the parameters, you may choose the "Export" such that all the parameter will be saved as a configuration file as a template file to ease the parameter input process for the next time. Click "Import" and select an existing configuration file, the saved data will be read from the file to initialize the parameters to define the calculation.

3) Defining the calculation

To create a new calculation project after clicking "New", a form for the parameters input is presented as the figure below. In general, calculation parameters are categorized into five groups: well configuration, fluid property, geothermal temperature, wax deposition and reservoir.

You may click the corresponding button to complete the data input. The simulation does not start until all the necessary data entries are specified and the "Calculate" button is clicked.
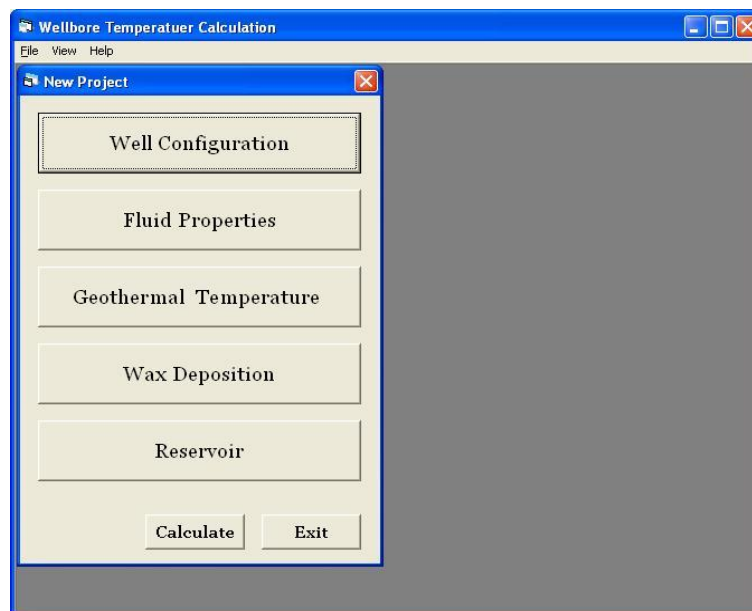
Note that the form of "Operation Time" is displayed before the calculation starts. You need to click the button "Run" to perform the simulation.
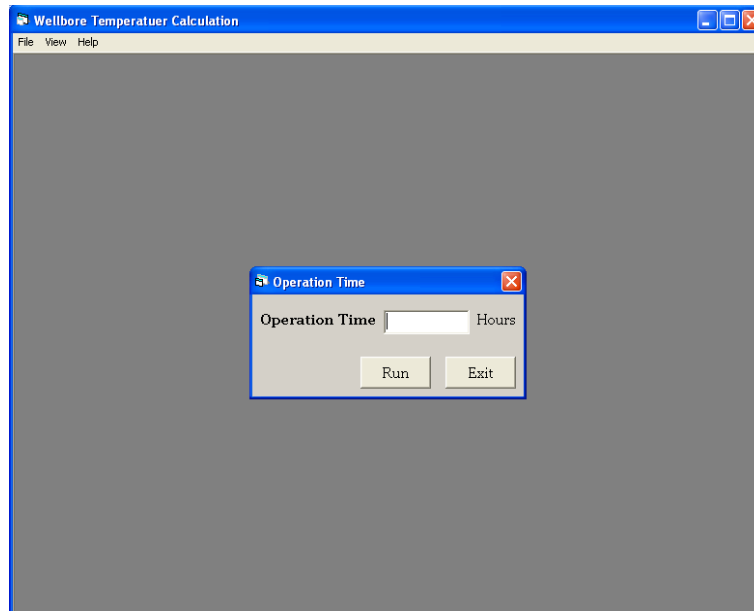
5.2 Parameters Input

Following the "New project" window mentioned above, the user needs to specify the parameters as input to the numerical computation. Note that typical values are provided when field data is not available as a default case. Only numbers are allowed in the data entry, and an alert message box will remind the users when invalid or mistaken stings are given. When "Exit" button is clicked, the software will check the all the necessary parameters in the data entries from different input windows. In this section, input parameters are grouped and introduced.

5.2.1 Well Configuration

1) Well
Click "Well Configuration" → "Well": the parameters of a well involve flow rate, initial temperatures and location of gas lift valves.

Note that the checkbox "Gas Lift" is not activated until the option "Production Well" is selected.

The study considers constant flowrate and constant surface/bottomhole pressure. According to the requirement of the software, at most ten gas-lift valve values are allowed to input in the simulation.

To use default values, select injection or production well first. Check "Use default or imported" after the well type is confirmed. Namely, for a production well, the default condition is no gas injection during the production.
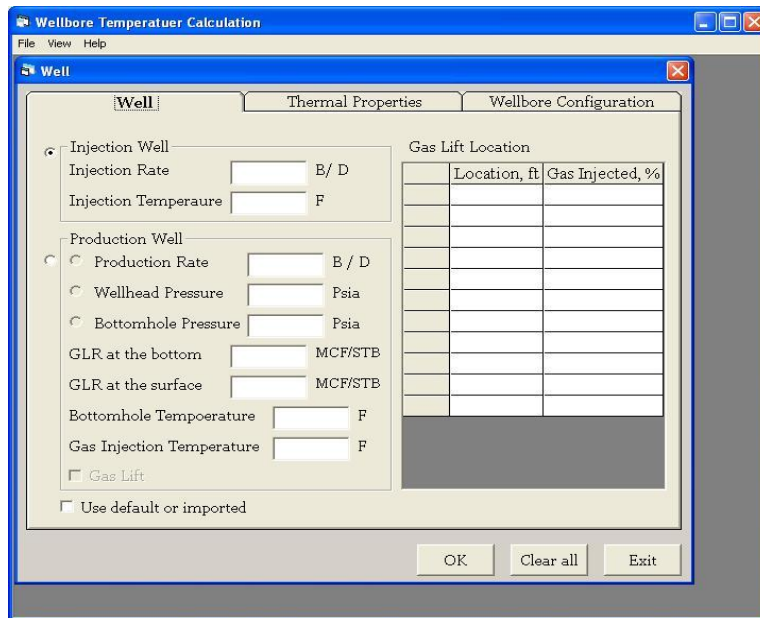
2) Thermal Properties

Choose "Well Configuration" → "Thermal Properties": wellbore and formation/permafrost thermal properties shall be input in the window popped up.

In the figure below which corresponds to the window of "Thermal properties", the depth refers to the permafrost thickness from the ground surface to the permafrost bottom. To adopt default

values of the permafrost, you may select the checkbox "Permafrost" first, and then click "Default". Otherwise, it is assumed the well located in the non-permafrost region.
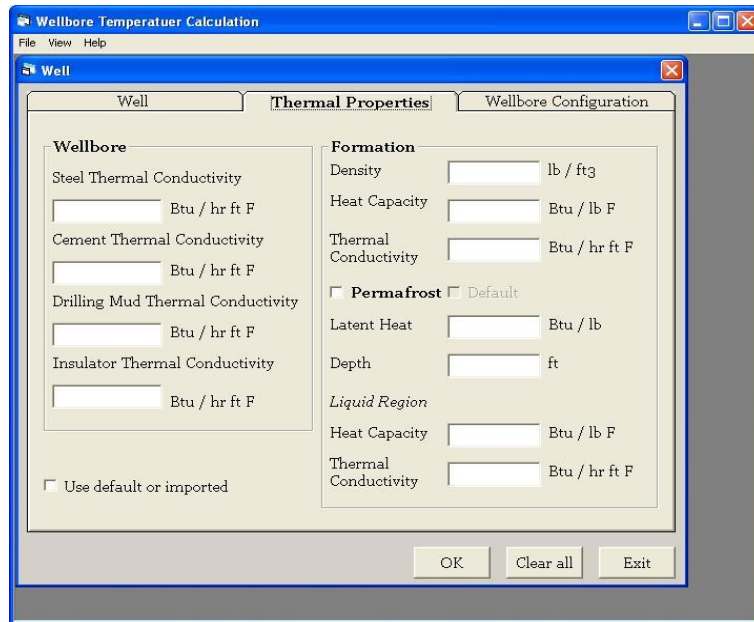
3) Wellbore Configuration

A third window is provided to take the input of the diameter and length information of the tubing, casing, insulator and drill hole. Recall that a typical wellbore structure in permafrost and non-permafrost has been introduced in Chapter III.
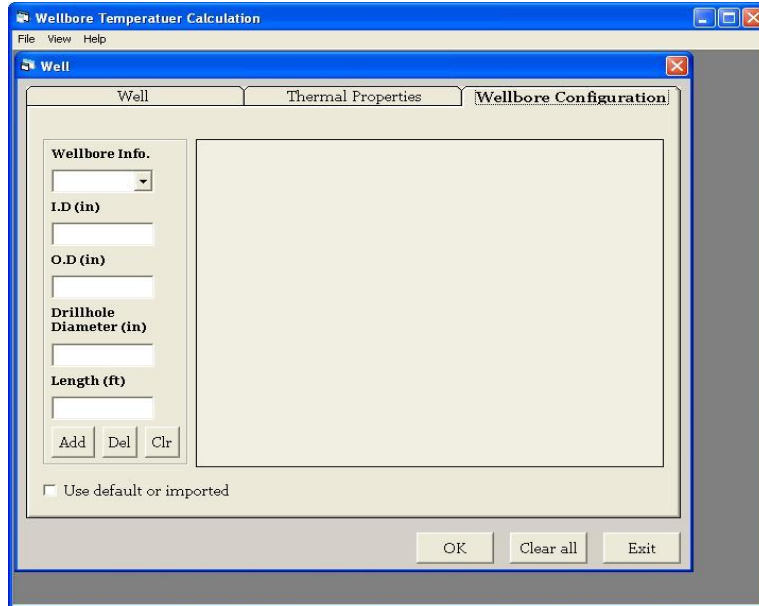
5.2.2 Fluid Properties

The form of "Fluid Properties" is designed for the user to input both oil and gas properties, which include fluid heat capacity, thermal conductivity, density and viscosity. By gas, air is considered in the context.
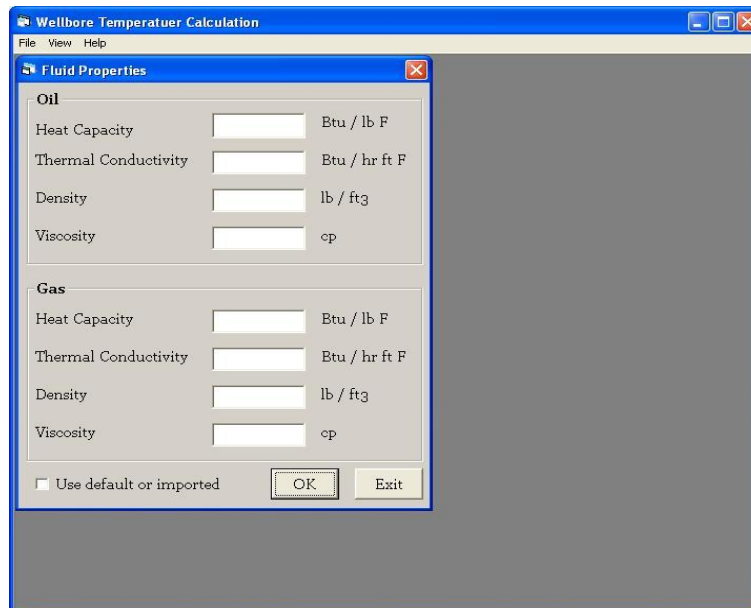
5.2.3 Geothermal Temperature

The software provides a third parameter input form to define the geothermal temperature either through the geothermal gradient and the ground surface temperature; or through the depth and corresponding geothermal temperature. For the latter, at least five pairs of (depth, temperature) are necessary for the software to obtain the relationship between the geothermal temperature and the vertical depth through curve fitting.

Default values of geothermal gradient and ground temperature are given by the program.
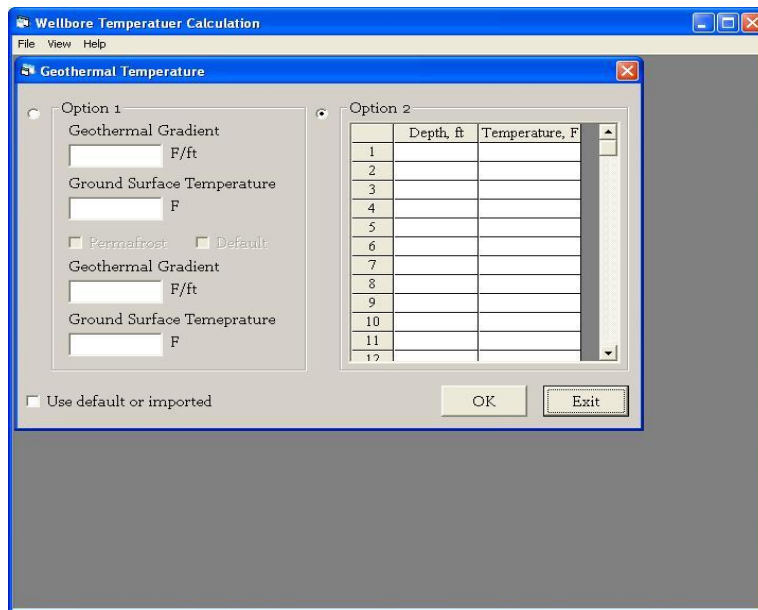
5.2.4 Wax Deposition

You can click the button "Wax Deposition" to open the form for wax thermal properties.

There are two options for the calculation: considering the wax deposition or not. For the scenario with the wax deposition, nine parameters are needed: wax thermal conductivity, gel density, wax

concentration in the oil, constants a, b, c for solubility, molecular diffusivity, aspect ratio and wax appearance temperature where a, b, c and aspect ratio from Singh's experiments are adopted as default values.
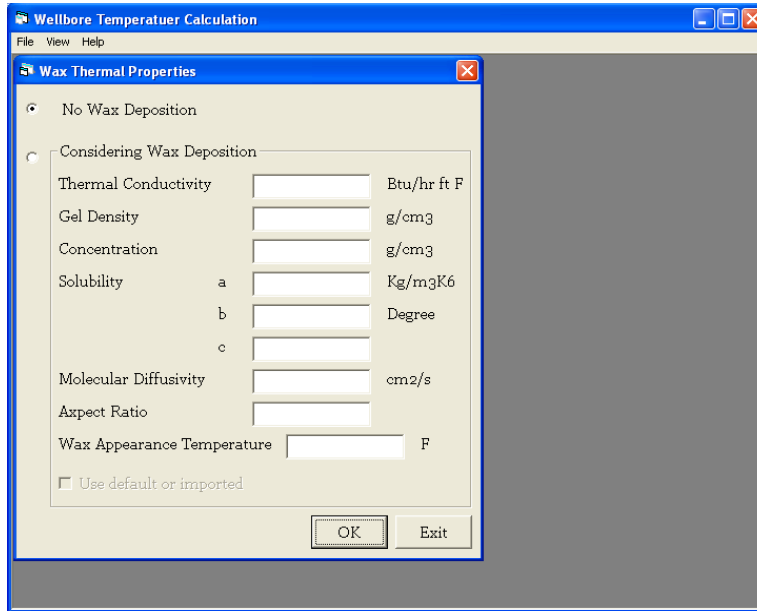
5.2.5 Reservoir

The form of "Reservoir" is used for the reservoir information input. Formation volume factor, total system compressibility, reservoir thickness, permeability, initial reservoir pressure, porosity, and wellbore radius are required for the calculation.

You could select the checkbox "Use default or imported" to import certain initial values either given by the program or the saved parameter template.

Similar to above parameter input windows, you may press "OK" to validate the input.
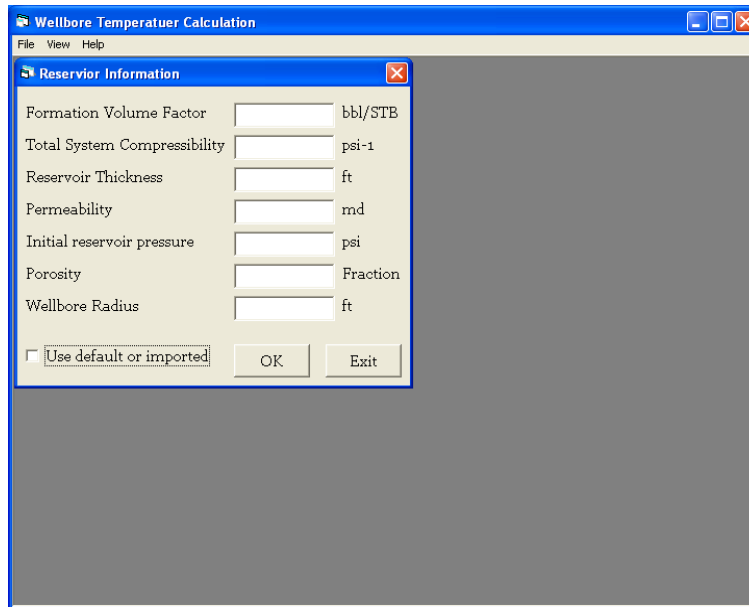
5.3 Results Output

On completion of parameter specification, you could start the calculation process by clicking "Calculate". The VB user interface will return once MATLAB computation completes and you may choose to illustrate or summarize the results.

5.3.1 Summary of Input Parameters

The parameters are summarized after information input is complete. A file includes all parameters is provided. Users can browse the summary by clicking "View" button in the menu bar and then choose "Parameter Summary".
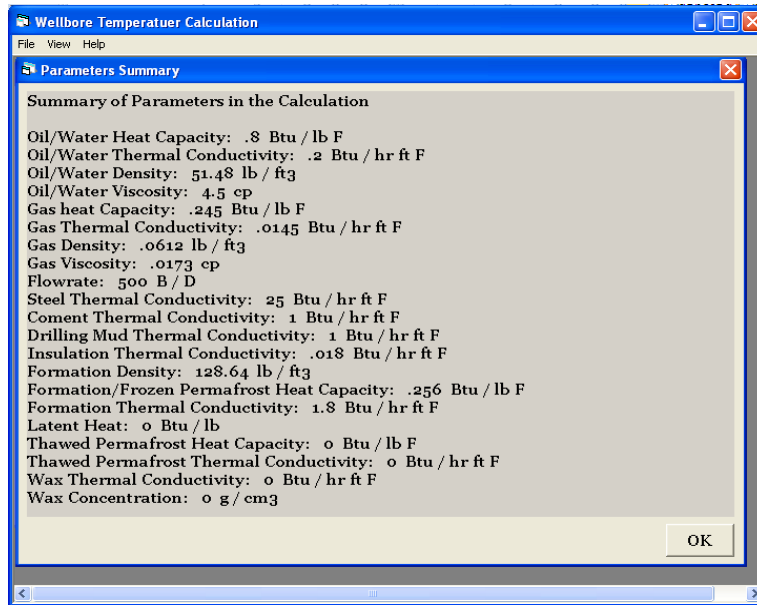
The program also provides the wellbore profile if the data is available and saved as Excel file. Users can browse the profile by following the procedures: click "View" in the menu bar, and then select "Wellbore Profile".
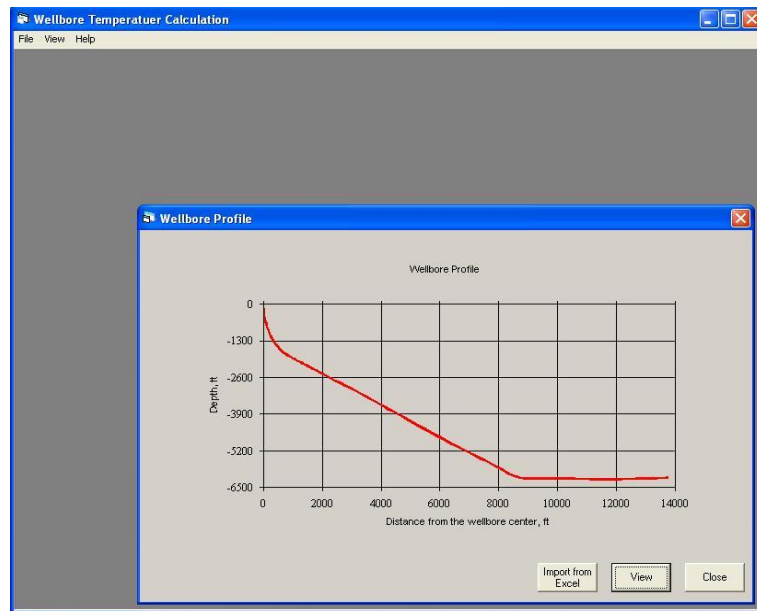
If the profile has been saved in an excel file, click button "Import from Excel" and choose the

file. You may then illustrate the wellbore profile when the importing is completed.

5.3.2 Calculation Results

The output results include: temperature and pressure distributions, moving boundary location, and wax deposition.

The fluid/gas temperature and pressure change with the wellbore length can be plotted in a figure while the fluid temperature is shown in a table after the calculation.
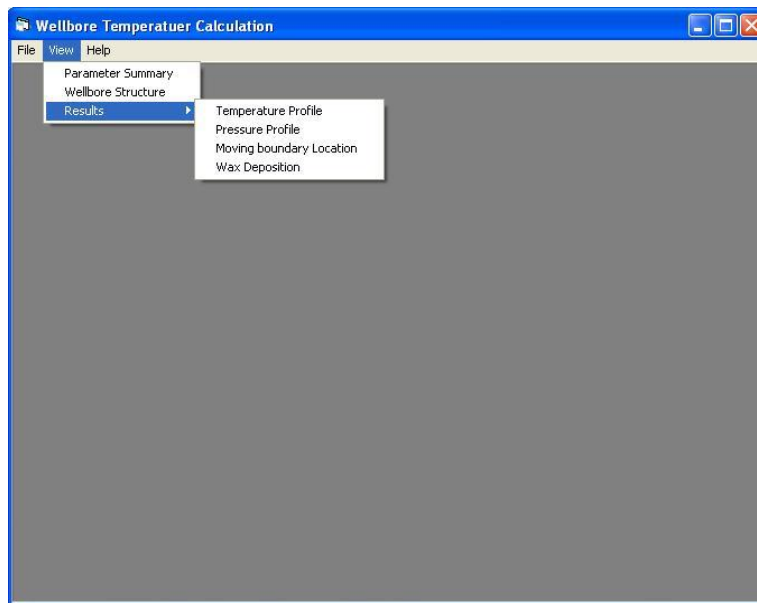
Examples of the temperature profile and moving boundary location are given below by selecting from the menu "View" → "Temperature Profile" → "Moving boundary location".
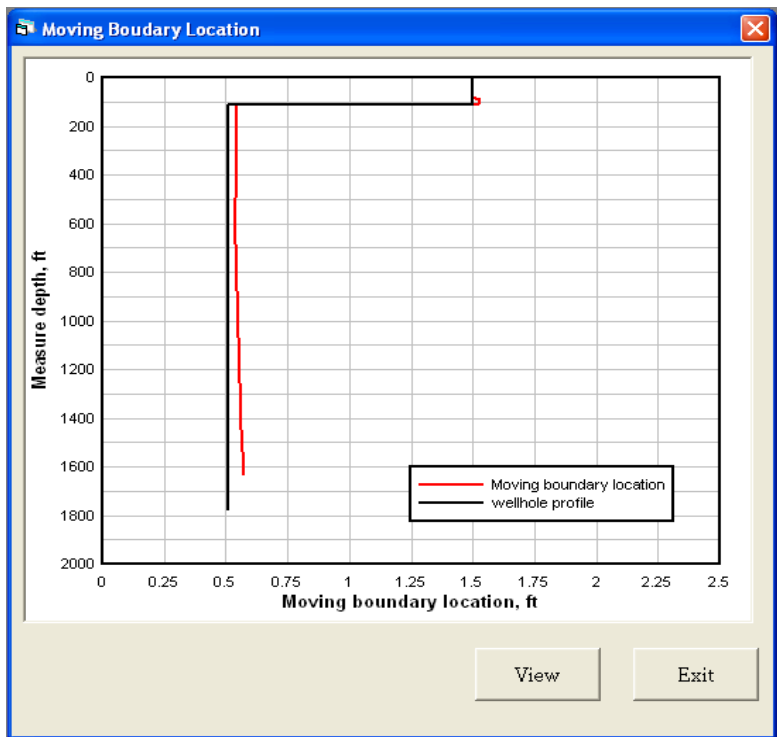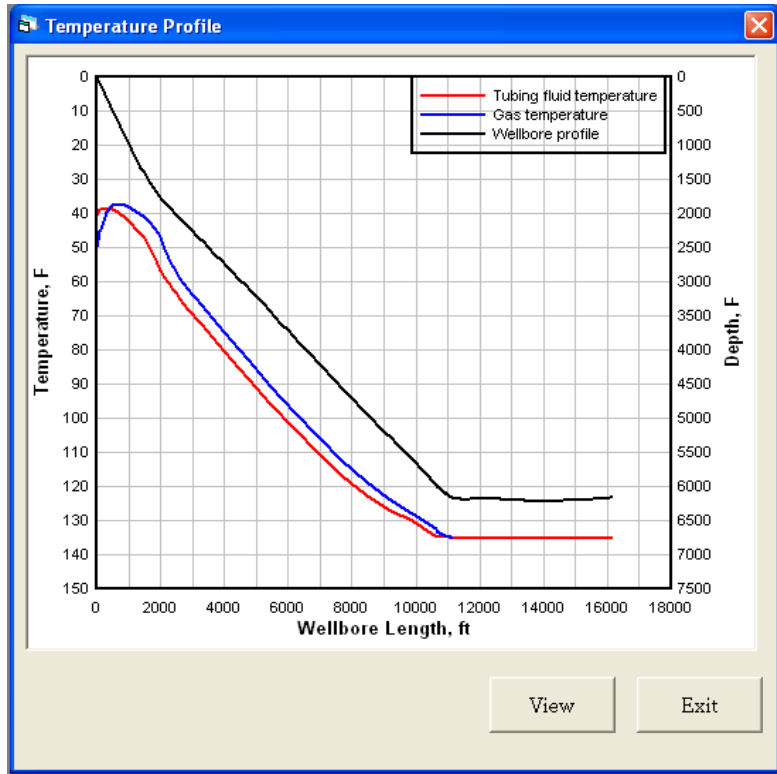
6. Results

This chapter illustrates the capability of the models developed in this thesis to estimate temperature, pressure and wax deposition during production and injection. Section 6.1.1 illustrates estimation of temperature distributions in injection and production wells. In production wells, the effects of insulation and gas-lift valve on wellbore heat transfer in a wellbore containing a permafrost region are illustrated. Section 6.1.2 presents the pressure calculation under constant flowrate and constant surface (bottomhole) pressure conditions. Wax deposition based on Singh's model is simulated in section 6.1.3. All calculated results are tabulated in Appendix B.

6.1 Results

6.1.1 Temperature Calculation

This section illustrates the calculation of injection well and production well temperature distributions under several operation conditions.

6.1.2.1 Injection Well

Application of the model for an injection well assumes the well is located in a non-permafrost region and the wellbore structure is shown by Figure 3-3. The annulus between tubing and production casing was filled with gas. Water is injected down through the tubing at 500 ℉ at the rate of 200 ft^3/hr. The temperature profile of water after 30 days is shown in Figure 6-1.
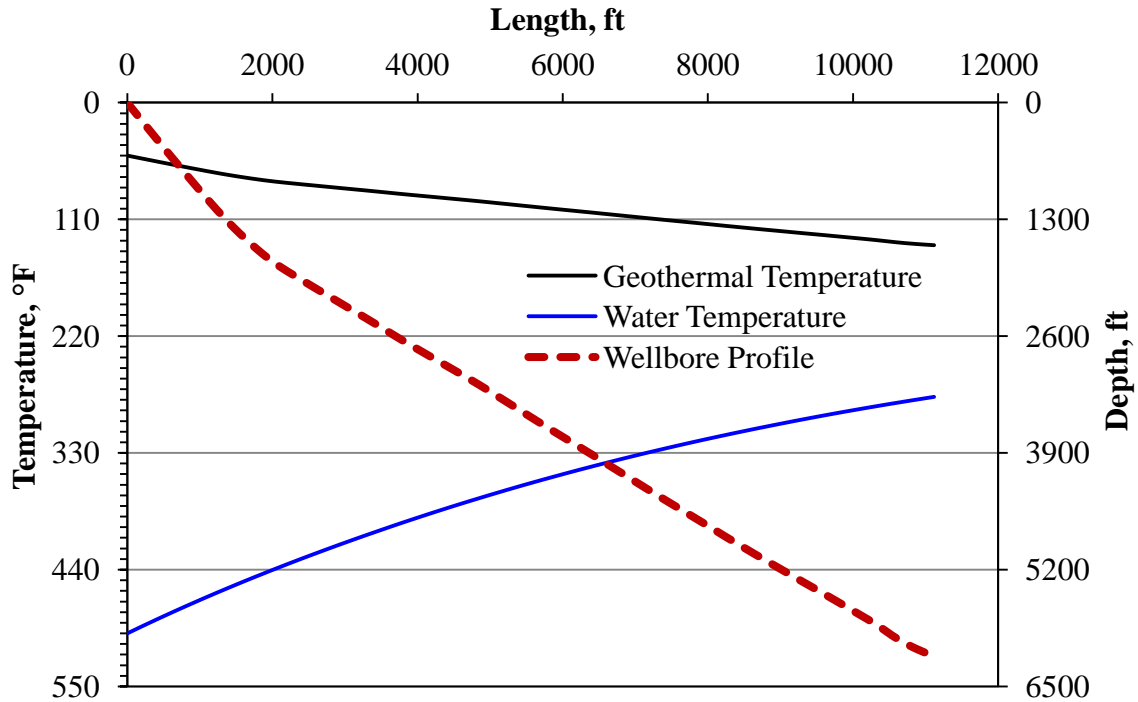
Fig. 6-1 Calculated water temperature in the well after 30 days injection

6.1.2.2 Production Well

Estimation of temperature distribution in a production well considers the wellbore located in a permafrost region and the wellbore structure described by Figure 3-2. The oil production rate is 500 B/D. The annular between tubing and production casing was filled with gas. We first consider insulation effect and then include gas lift valves (GLV) into the temperature model.

Effect of Insulation on Temperature Distributions

Three cases were simulated assuming no gas was injected into the tubing through GLV during the production

Case I: The wellbore was not insulated.

Case II: The conductor was insulated by an 80 ft length of insulation.

Case III: The tubing in the upper part of the permafrost (1700 feet) was insulated with insulation which was 0.65 inch thick.

Temperature profiles are shown in Figure 6-3 for Cases I-III. In Figure 6-3, red dashed lines show the locations of 80 ft and 1700 ft along the wellbore length. The wellhead temperatures in Case II and Case III (with insulation) are higher than that in Case I (without insulation). The difference of wellhead temperature between Case I and Case II is not significant since only an 80-ft long insulator exists around the conductor. In Case III (1700 ft insulated) the wellhead temperature was about 10 ℉ higher than Cases I and II. Based on the calculation, it shows that insulating the well prevents the heat loss from the wellbore to the formation.
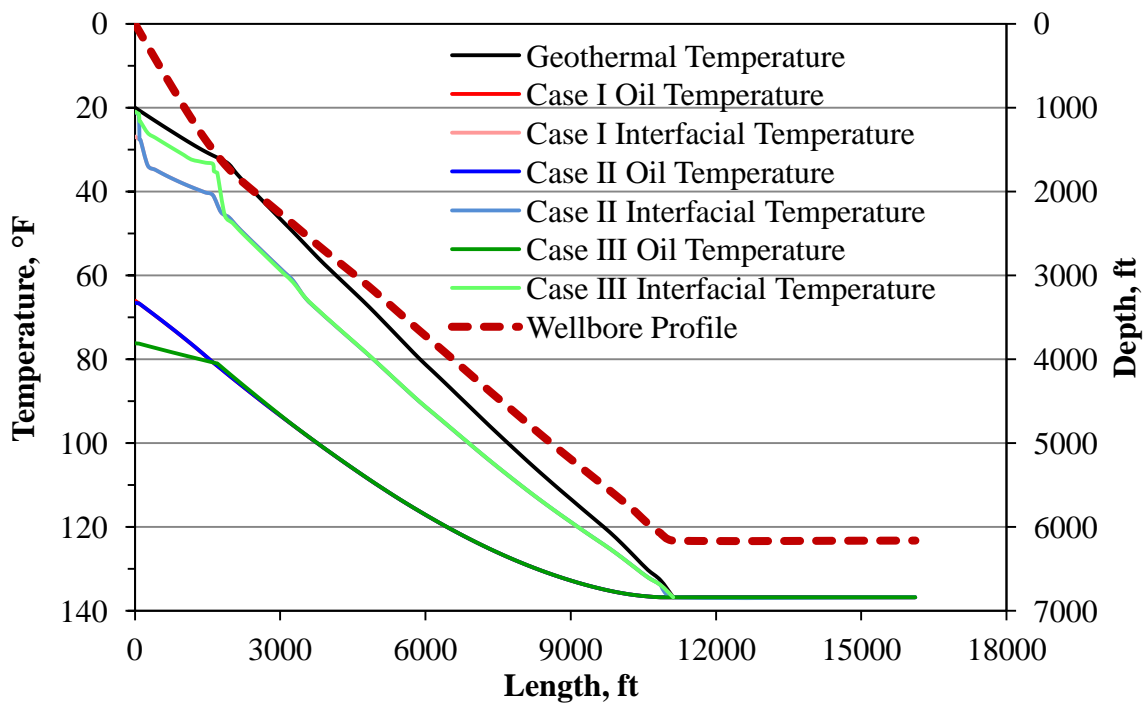
Fig. 6-2 Calculated oil and wellbore/formation interfacial temperature at 500 B/D after 30 days production with or without insulating the well
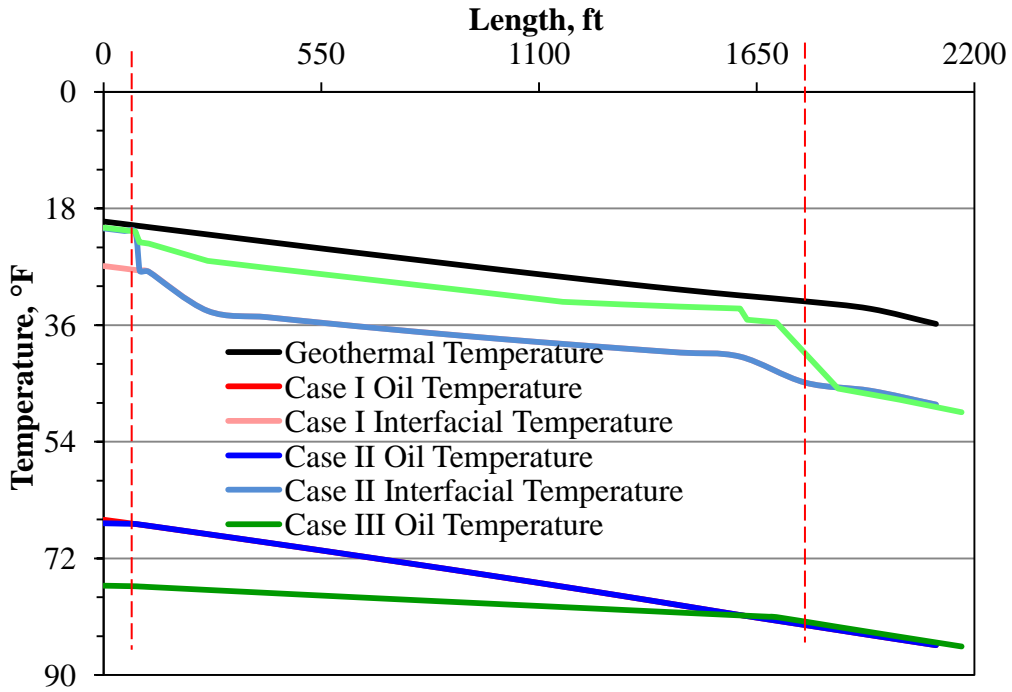
Fig. 6-3 Zoomed view of temperature distribution in Figure 6-2 from length 0 to 2200 ft
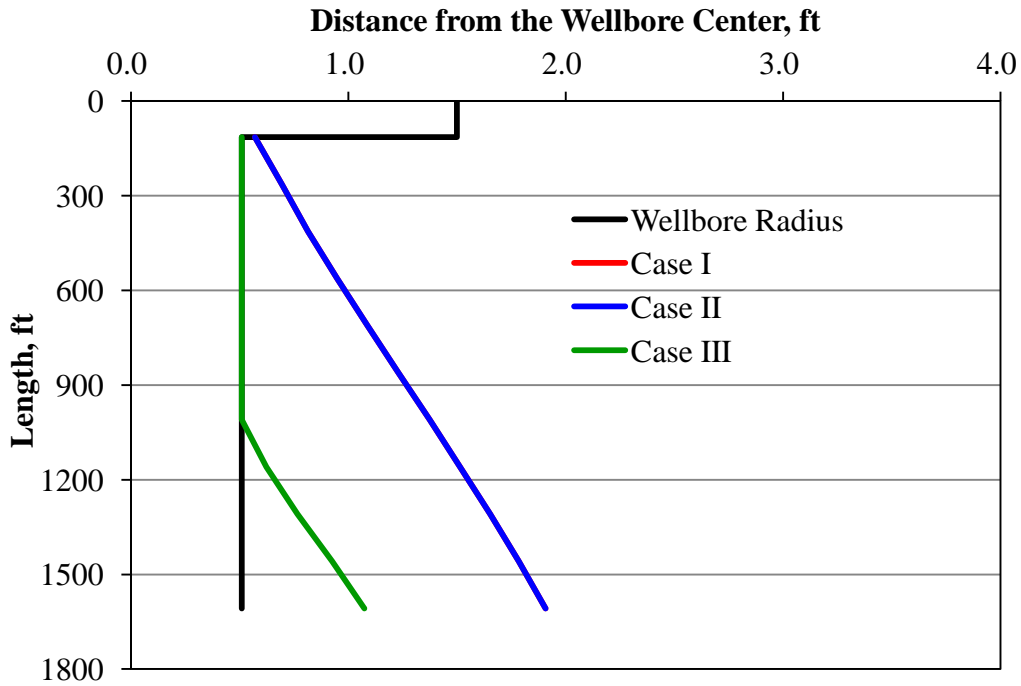
Fig. 6-4 Calculated moving boundary location at 500 B/D after 30 days production with and without insulating the well

The temperature at wellbore/formation (permafrost) interface follows the same trend as the wellhead temperature. In permafrost region, reducing heat loss to the permafrost is crucial to prevent permafrost from thawing. Insulating the conductor prevents the permafrost thawing near the ground surface. Insulating the tubing delays the movement of the moving boundary in the permafrost region, as illustrated in Figure 6-4.

For the simulation time of one year, temperature profiles are shown in Figure 6-5. The results show the same trend as Cases of 30 days. In Case III (1700 ft insulated) the wellhead temperature was about 9 ℉ higher than Cases I and II.
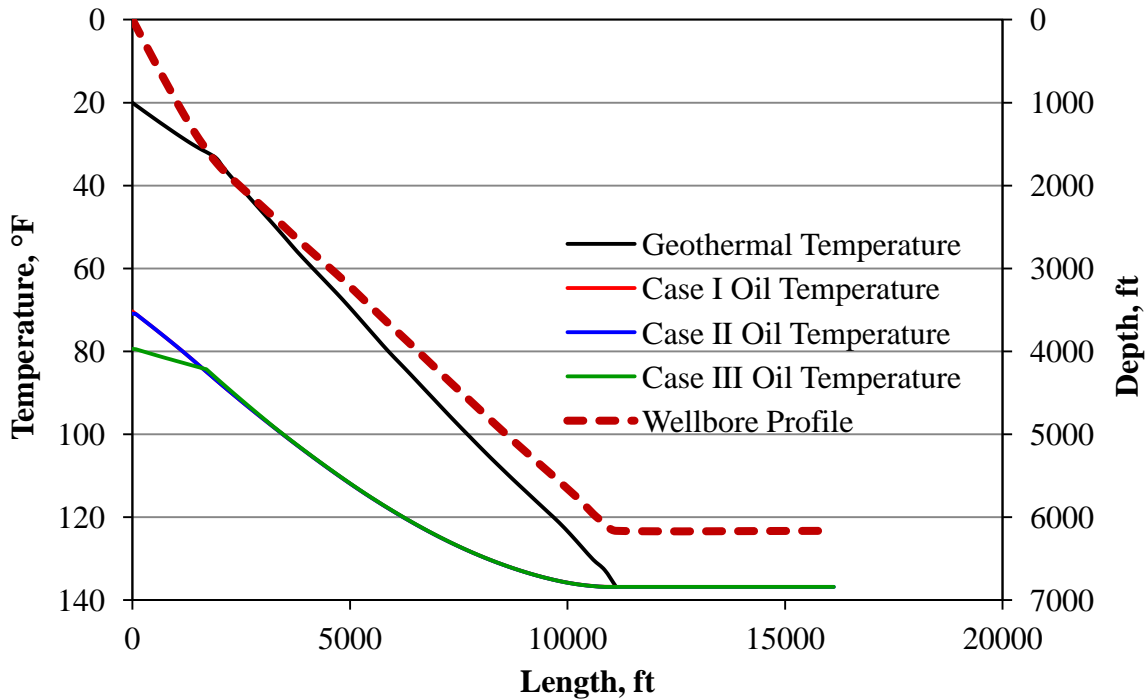
Fig. 6-5 Calculated oil and wellbore/formation interfacial temperature at 500 B/D after one year production with or without insulating the well

Effect of Gas Lift Valves on Temperature Distributions

We consider three cases:

Case II: No gas was injected during the production.

Case IV: Gas was injected down through the annulus at 50 ℉, maintaining a gas/liquid ratio of 12000 SCF/STB at the wellhead. Gas was injected through the gas-lift valve located at 11118 ft along the well length direction.

Case V: The annulus was evacuated. Heat transfers from the outside tubing surface to the inside production casing surface only by radiation.

Temperature profiles are shown in Figure 6-6. In Case IV, the wellhead temperature was 24 ℉ lower than Case II. The fluid temperature decreases because gas injection accelerates the heat loss both from the fluid in the tubing to the surrounding and from the tubing to the production casing. The wellbore heat loss can be reduced by evacuating the annulus. The wellhead temperature in Case V (annulus evacuated) was 8 ℉ higher than Cases II.

The effect of gas injection on the location of the moving boundary is shown in Figure 6-7. The moving front proceeded much faster when gas was injected and the trend is illustrated in Figure 6-8. With gas injection, the temperature distribution of the permafrost was higher than that without any gas injection at the same location.
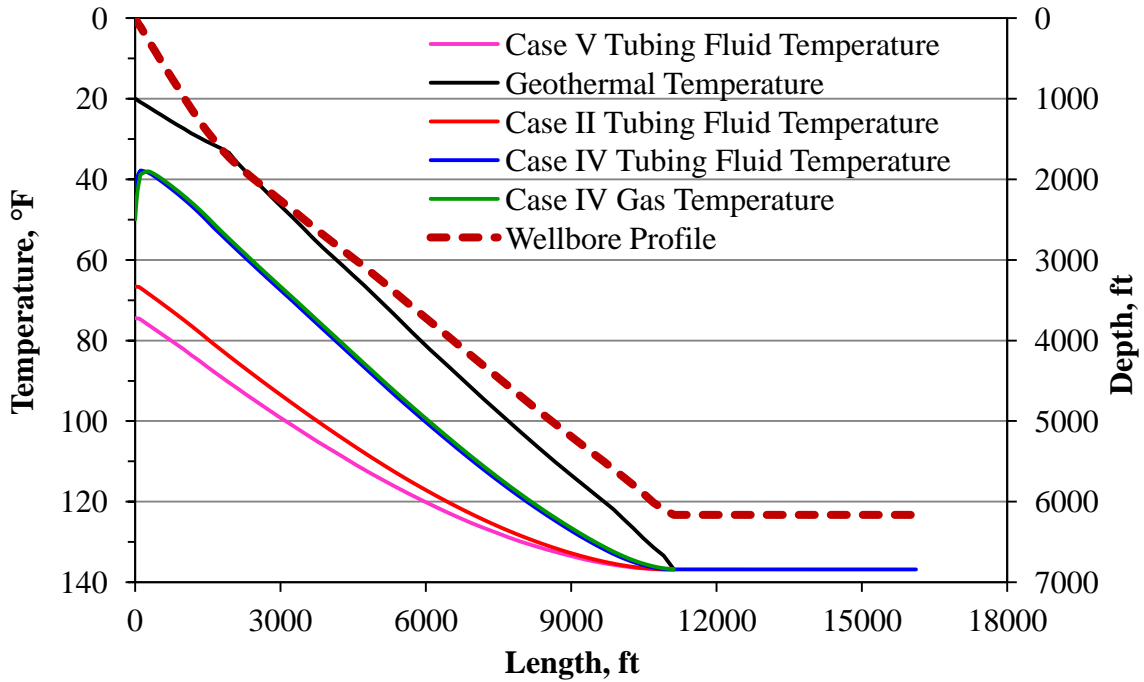
Fig. 6-6 Calculated tubing fluid and gas temperature at 500 B/D after 30 days production with or without GLV
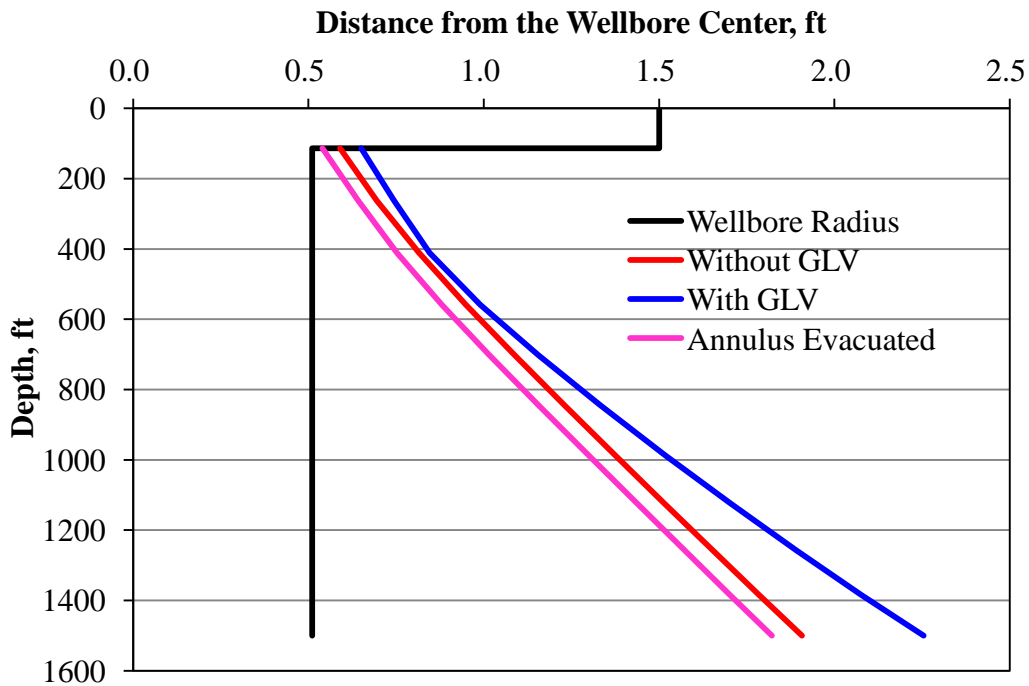
Fig. 6-7 Calculated moving boundary location at 500 B/D after 30 days production with or without GLV

68
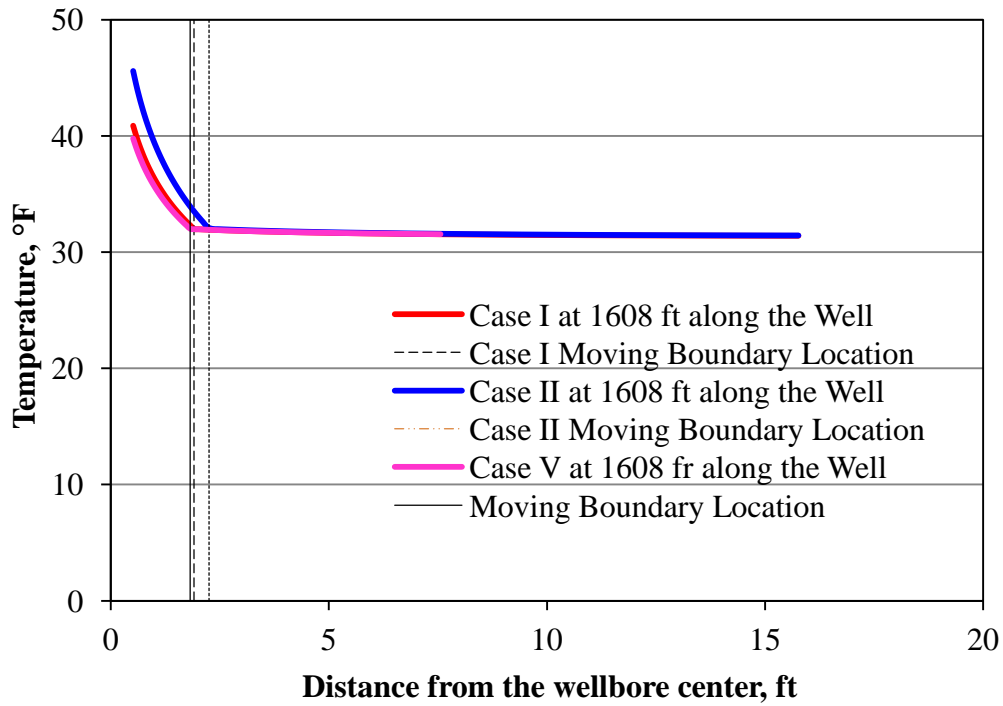
Fig. 6-8 Calculated permafrost temperature at 500 B/D after 30 days production with or without GLV

For the simulation time of one year, temperature profiles are shown in Figure 6-9. The results show the same trend as Cases of 30 days.
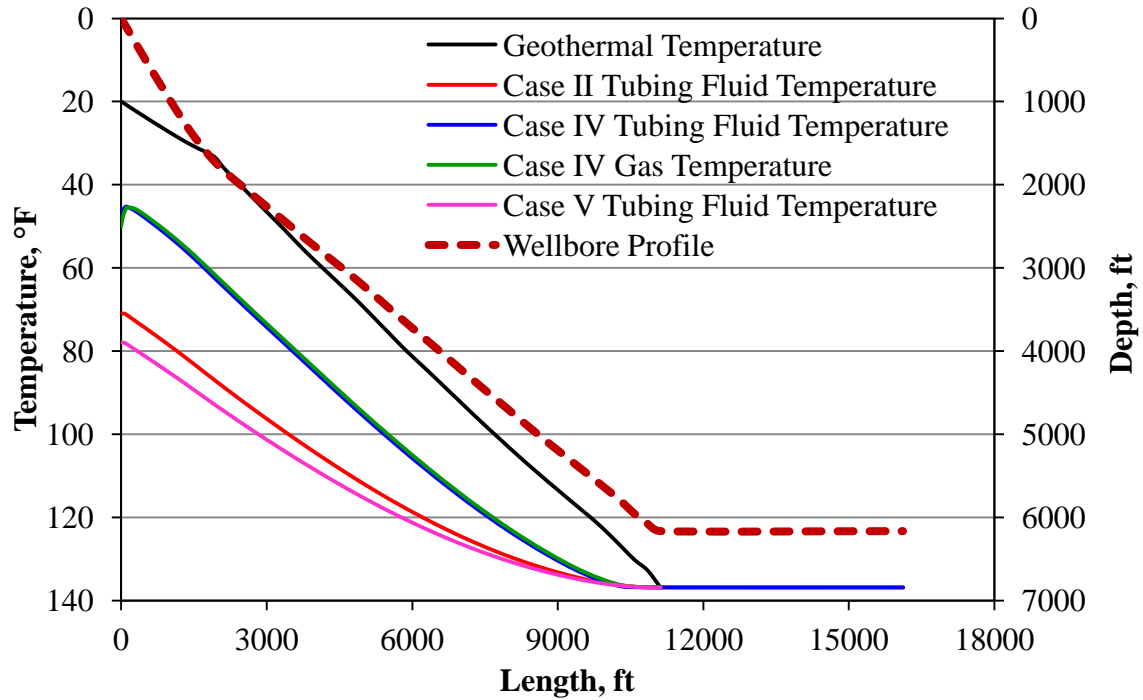
Fig. 6-9 Calculated tubing fluid and gas temperature at 500 B/D after one year production with or without GLV

6.1.2 Pressure Drop Calculation

Pressure distributions during production were estimated for three cases. No gas was injected and no water was produced during the production.

The three cases calculated were:

Case I_P: Oil was produced at a constant flowrate 500 B/D

Case II_P: Oil was produced under a constant surface pressure 300 psig.

Case III_P: Oil was produced under a constant bottomhole pressure 2400 psig.

The simulation results are presented in Figure 6-10 ~ Figure 6-12. Figure 6-10 shows the pressure distribution in the tubing under different production conditions, while Figure 6-11 characterizes the flowrate changes with time. Oil is produced under a constant bottomhole pressure 2400 psig at the biggest production rate and the lowest wellhead pressure. According to

70

the calculation, the bottomhole pressure should be higher than 2210 psig to produce the oil from the reservoir.

The flowrate calculation was based on pressure drop equations and transit inflow performance relationship as introduced in Section 3.3.4. In transit state, the flowrate decreases over time under constant surface (bottomhole) pressure conditions.

The comparison of moving boundary locations under different production conditions is shown in Figure 6-12. It is reported in the analytical study of Stefan problem that the moving boundary location is proportional to the square root of the time [16, 41], and the relationship is captured by our simulation results. Figure 6-12 illustrates the moving boundary in a linear relationship with the value of square time.

The production rate in Case III_P is higher than Case I_P and II_P. The moving boundary moved fastest under constant bottomhole pressure condition because the heat loss from the wellbore to the permafrost was accelerated at higher production rate.
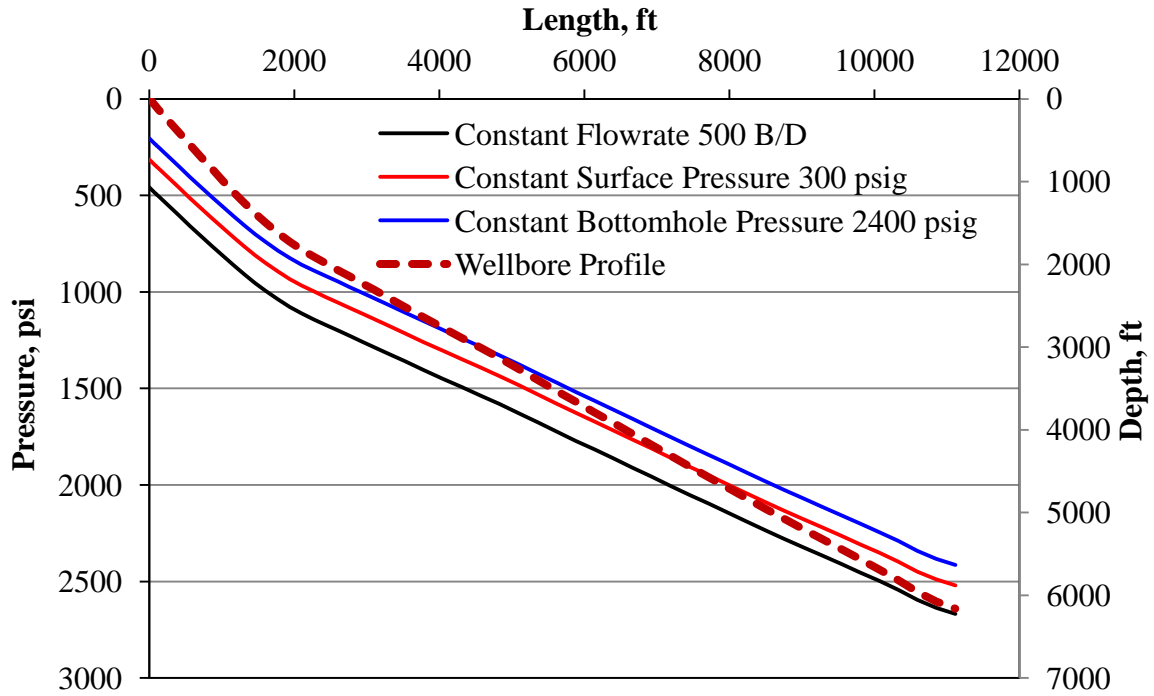
Fig. 6-10 Calculated pressure under different production conditions after 30 days

Note: Pressure gradients are linear, but have different slopes between permafrost and non permafrost region.
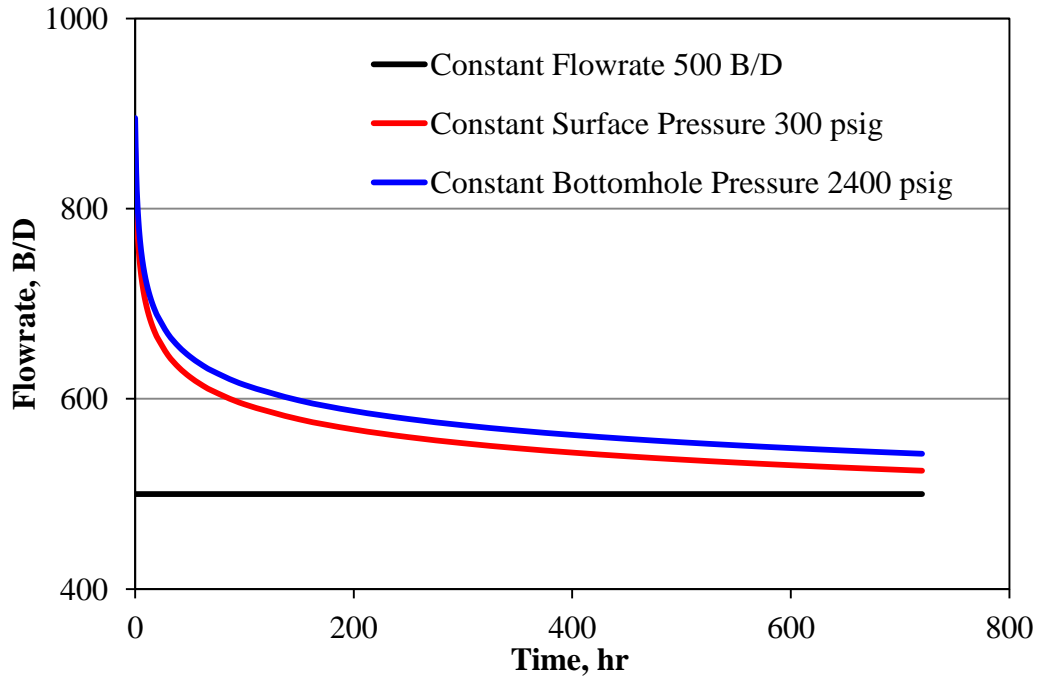
Fig. 6-11 Calculated flowrate changes in 30 days under different production conditions
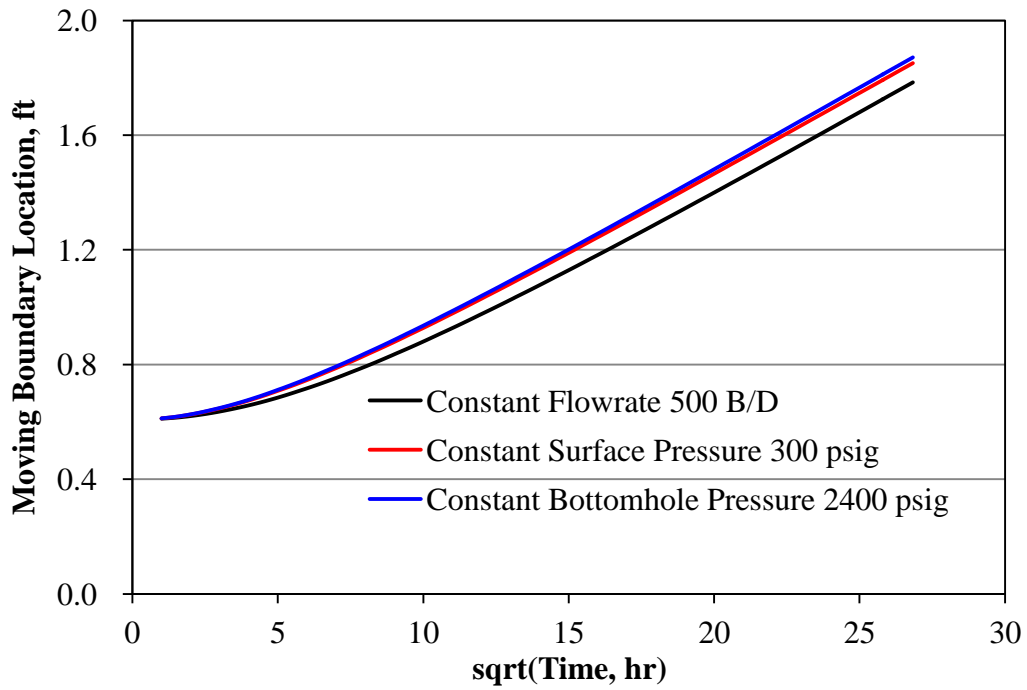
Fig. 6-12 Calculated moving boundary location under different production conditions after 30 days at the length of 1458 ft

6.1.3 Wax Deposition Calculation

Singh's model is valid for laminar flow (Re < 2300). Therefore, the model is valid only when the production rate is less than 168 B/D for the well configuration studied in this section. Also, the parameters in the wax solubility equation (Eq. 4-3) are based upon the sample oil in Singh's experiments. The examples were developed for a well located in permafrost region, and the wellbore structure is shown in Figure 3-2. No gas was injected during the production. Oil temperature at the bottom is equal to the formation temperature.

This section presents the results of wax deposition calculation, and evaluates the effect of wax deposition on wellbore heat transfer and pressure profile.

6.1.3.1 Wax Deposition at Different Flowrates

We consider production rate at 168, 300, 350, 400, and 500 B/D, respectively. Effects of flowrate on wax deposition are illustrated by Figure 6-13 and Figure 6-14. The temperatures at different flowrates are listed in Appendix B.

No wax deposition occurred when the production rate was 500 B/D. Figure 6-13 shows the effective tubing radius after the well was produced for 10 days at each flowrate. Wax deposition appears at deeper location on the tubing wall as the production rate decreased.
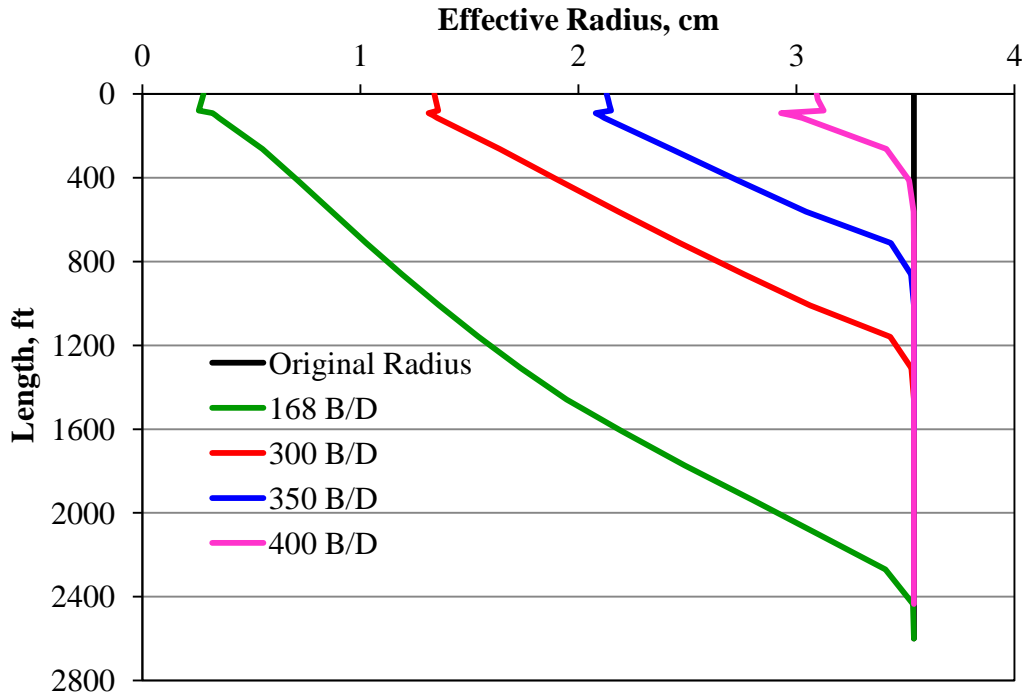
Fig. 6-13 Calculated effective inner tubing radius at different flowrates after 10 days

Figure 6-14 shows the temperature at wax/oil interface at different flowrates after 10 days. The black dashed line represents WAT, the wax appearance temperature (57 °F). Above the line, wax stops growing. At the production rates 168, 300 and 350 B/D, wax deposition keeps growing since the temperature distributions near the wellhead are still lower than WAT. At production rate 400 B/D, the temperature at wax/oil interface along the well is above the line. It means wax stops growing after 10 days.
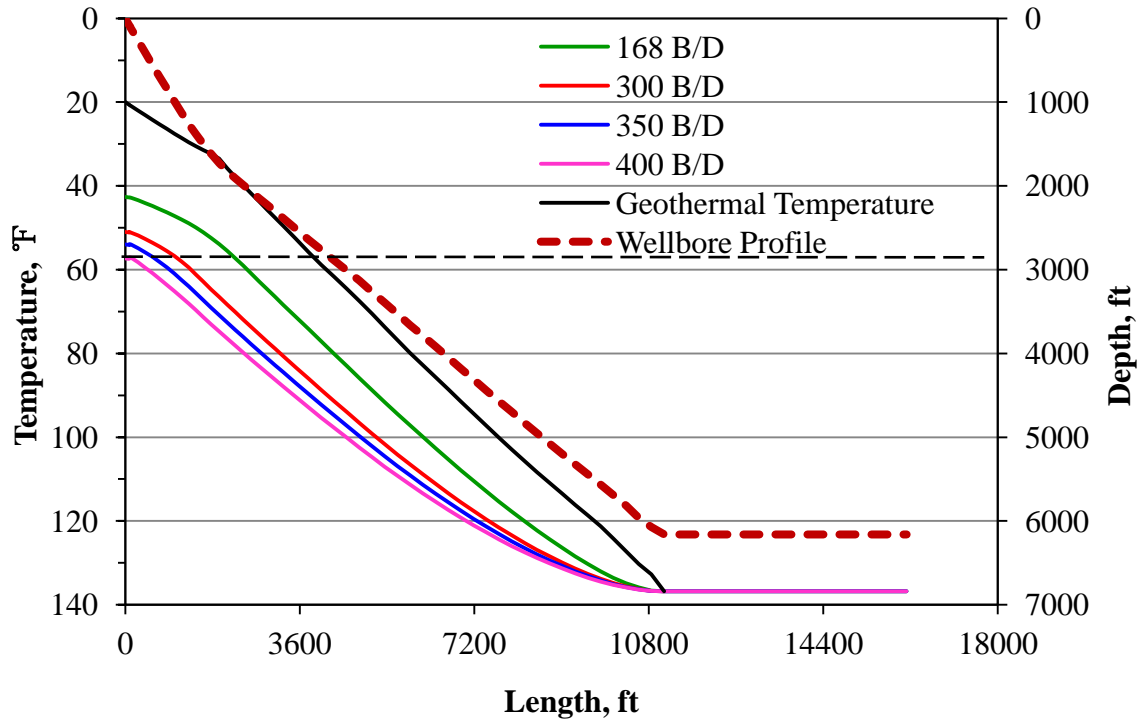
Fig. 6-14 Calculated temperature at wax/oil interface at different flowrates after 10 days

Figure 6-15 shows the oil temperature at 400B/D after 10 days and one year. The oil temperature at wellhead after one year is 4 ℉ higher than that after 10 days.
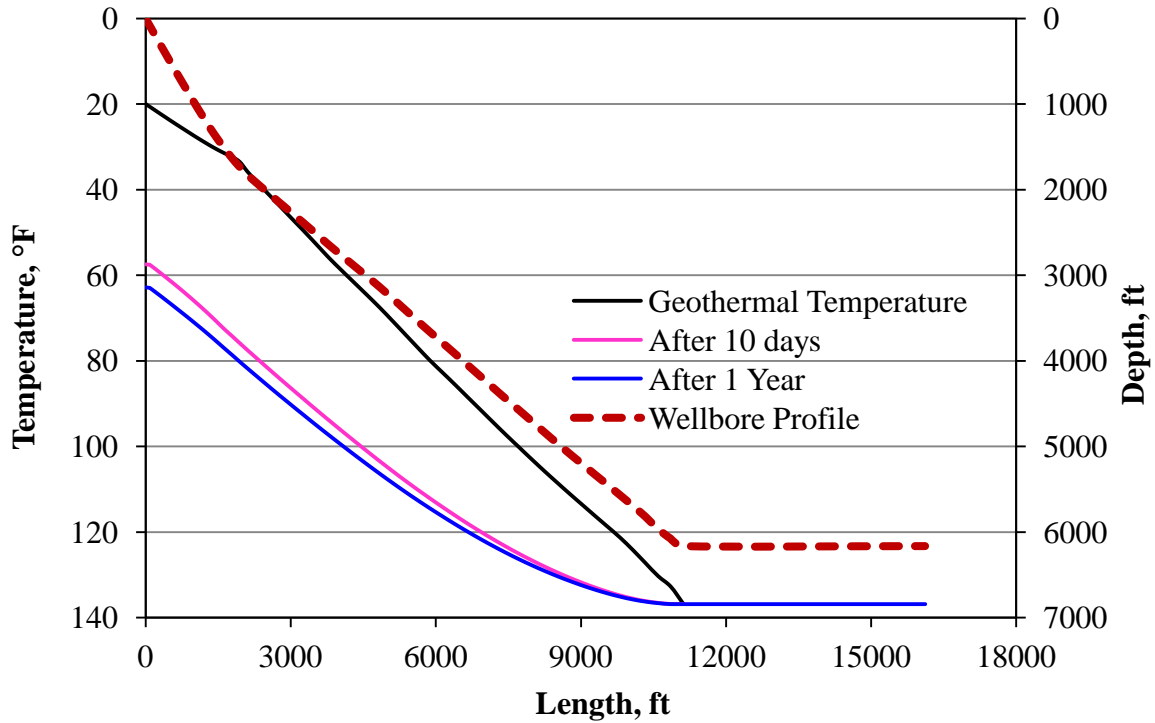
Fig. 6-15 Calculated oil temperature at 400 B/D after 10 days and one year

Figure 6-16 shows the pressure distributions at 168 B/D after 10 days, 300 B/D after 12 days and 350 B/D after 16days. When the production rate drops to 168 B/D, wax deposition almost blocks the whole cross-sectional area after 10 days (shown in Figure 6-13) and the wellhead pressure drops under 0 psi. It means the reservoir energy is not sufficient to sustain the production rate following the wax deposition and wellbore is shut in. At production rate 300 B/D and 350 B/D, the well is shut in after 12 days and 16 days respectively.
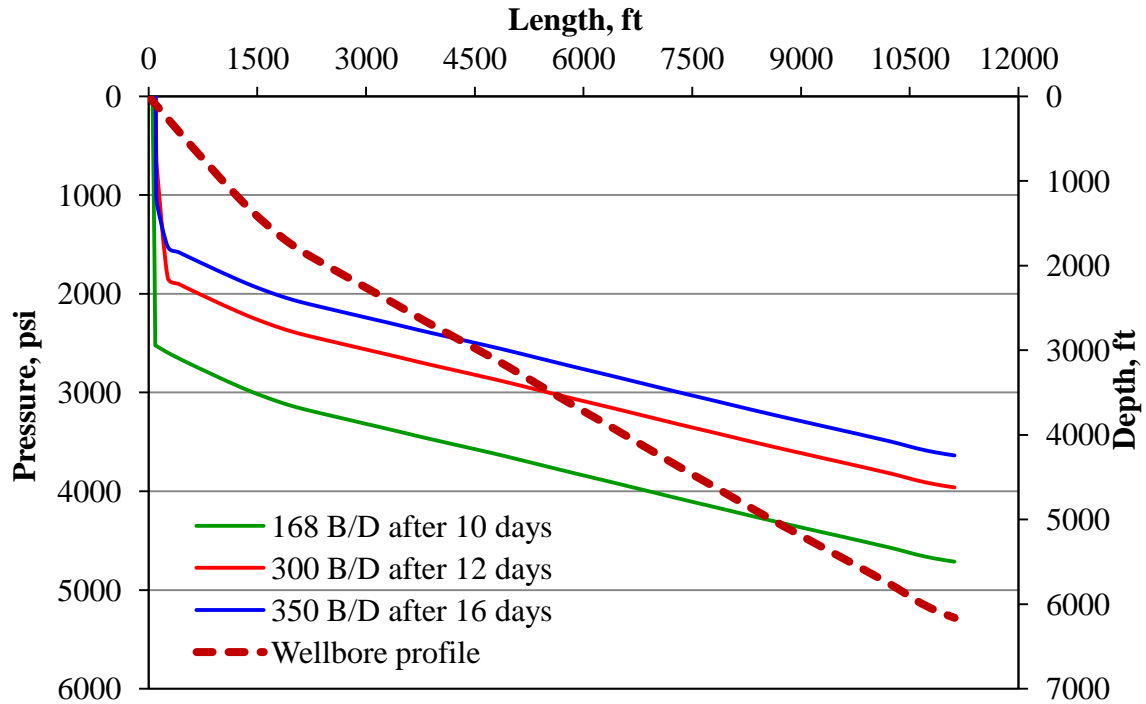
Fig. 6-16 Calculated pressure distributions at different flowrates

6.1.3.2 Effect of Wax Deposition on Oil Temperature and Pressure Calculation

The effect of wax deposition on temperature and pressure distributions is illustrated for the case where the production rate was 350 B/D. Effective tubing inner radius after 16 days is shown in Figure 6-17. The longitudinal coordinates represents the center of the wellbore. Black dash line represents the location of 80 ft under the surface along the wellbore length. The insulator around the wellbore conductor reduces the heat loss from the well to the formation; therefore the location with the thickest wax deposition is away from the wellhead.

78
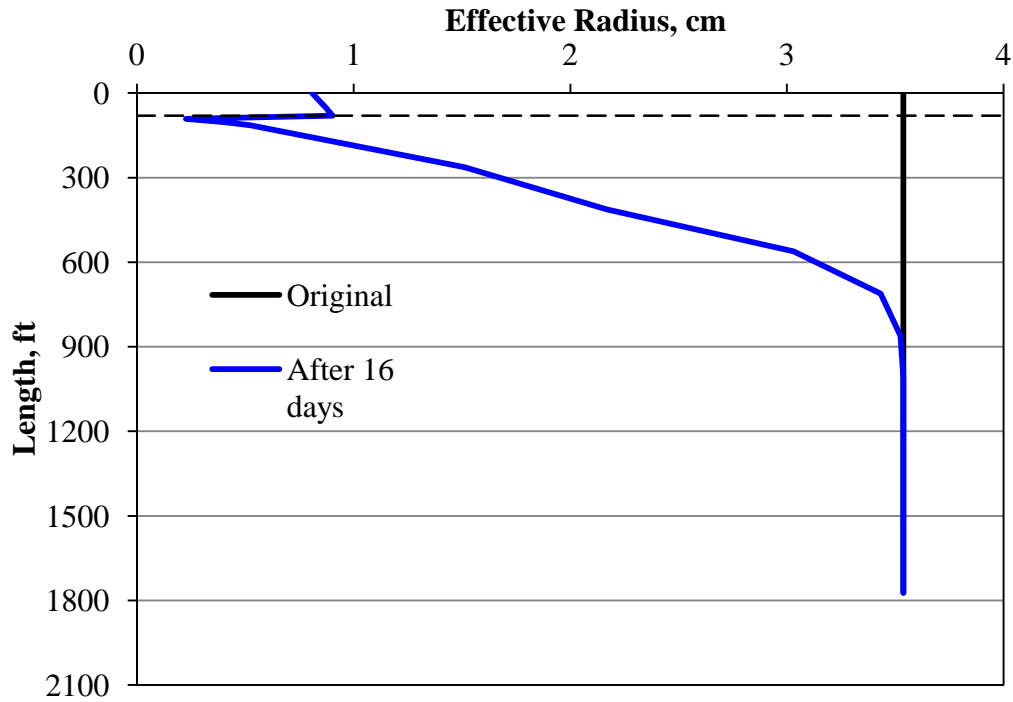
Fig. 6-17 Calculated effective inner tubing radius at 350 B/D after 16 days

Figure 6-18 compares the temperature calculation on the inner tubing surface with and without considering wax deposition after 16 days. The temperature decreases with thicker wax deposit because the wax insulator decreases the heat loss from the oil to the tubing wall. The temperature is higher near the wellhead because the insulator around the wellbore conductor decreases the heat loss from the well to the formation.
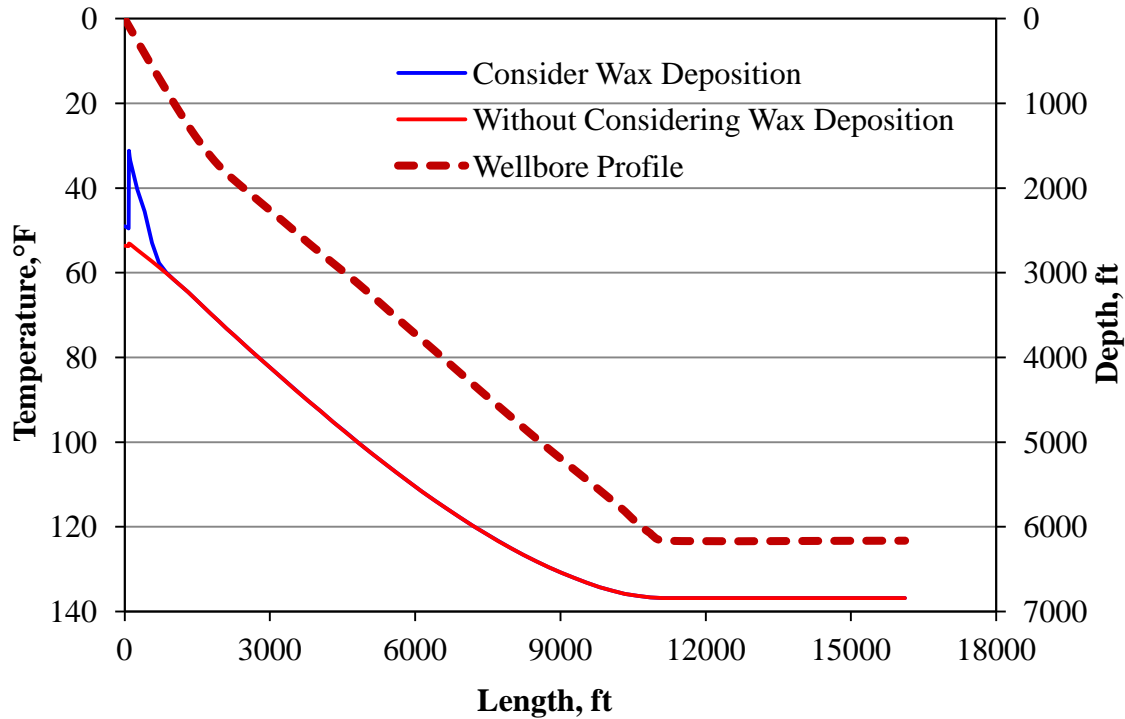
Fig. 6-18 Calculated temperature at inner tubing surface after 16 days

The pressure distribution after 16 days is shown in Figure 6-19. The pressure drop with wax deposition is larger near the wellhead because most of the cross-sectional area is blocked. The frictional pressure drop increases significantly as the effective tubing inner radius decreases, which is the major reason causing the increase in pressure drop after wax deposition.

Details of pressure calculation are presented in the following paragraphs.

1) The frictional pressure drop

If there was no wax deposition on the tubing wall, the pressure drop caused by friction is

$$u = \frac{4q}{\pi D^2} = \frac{4 \times \left(\dfrac{350\,^{B}\!/_{D} \times 5.615\,^{ft^3}\!/_{B}}{86400\,^{sec}\!/_{D}} \right)}{3.1416 \times (0.232\,ft)^2} = 0.54\,^{ft}\!/_{s}$$

$$N_{Re} = \frac{uD\rho}{\mu} = \frac{0.54\,^{ft}\!/_{s} \times 0.232\,ft \times 51.48\,^{lb_m}\!/_{ft^3}}{2cp \times 6.7197 \times 10^{-4}\,^{lb_m}\!/_{ft \cdot s \cdot cp}} = 4799 \quad f_f = 0.0097$$

80

$$\Delta P_{F1} = \frac{2 f_f \rho u^2 L}{g_c D} = \frac{2 \times 0.0097 \times 51.48^{\,lb_m}/_{ft^3} \times \left(0.54^{\,ft}/_s\right)^2 \times 1010\,ft}{32.174^{\,ft-lb_m}/_{lb_f-sec^2} \times 0.232\,ft \times 144^{\,in^2}/_{ft^2}} = 2.7 \times 10^{-4}\,psi\,/\,ft$$

If the tubing radius was reduced to 0.06 ft (0.85 cm), the pressure drop caused by friction is

$$u = 7.9^{\,ft}/_s \quad , \quad N_{\mathrm{Re}} = 18157, \quad f_f = 0.0071; \qquad \Delta P_{F1} = 0.164\,psi\,/\,ft$$

If the tubing radius was reduced to 0.018 ft (0.27 cm), the pressure drop caused by friction is

$$u = 72.4^{\,ft}/_s, \quad N_{\mathrm{Re}} = 49919, \quad f_f = 0.0059; \qquad \Delta P_{F1} = 38.2\,psi\,/\,ft$$

From the above calculations, it is clear that the frictional pressure drop increases significantly as the effective tubing inner radius decreases.

2) Pressure drop due to kinetic energy change

If the tubing radius decreases from 0.232 ft (2.78 in) to 0.018 ft (0.21 in), the pressure drop caused by kinetic energy change is

$$\Delta P_{KE} = 1.53 \times 10^{-8} \times 51.48 \times 350^2 \times \left(\frac{1}{0.21^4} - \frac{1}{2.78^4} \right) = 49.6\,psi$$

3) Pressure drop due to potential energy change

The total potential energy change from the location where wax appears on the tubing wall to the surface is

$$\Delta P_{PE} = \frac{g}{g_c}\rho\Delta z = \left(\frac{32.174^{\,ft}/_{s^2}}{32.174^{\,lb_m \cdot ft}/_{lb_f \cdot s^2}} \right) \times 51.48^{\,lb_m}/_{ft^3} \times 986.6\,ft \times {1ft^2}/_{144in^2} = 352.7\,psi$$
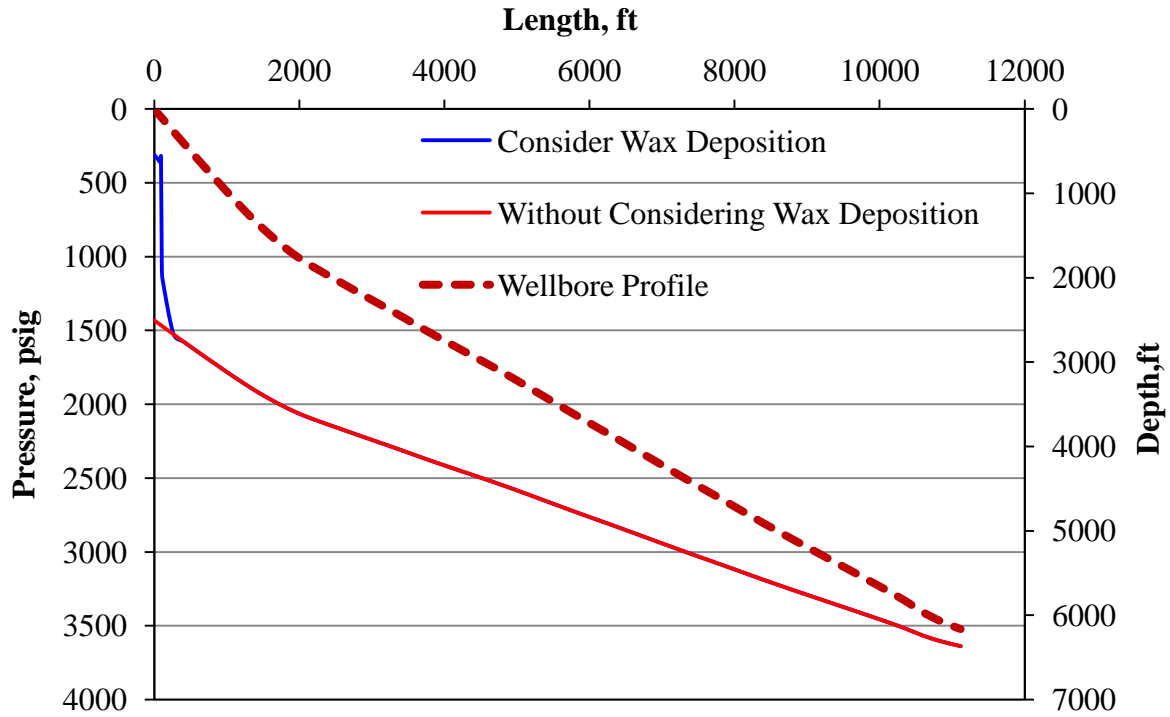
Fig. 6-19 Calculated oil pressure profile at 350 B/D after 16 days

Table 6-1 Calculated pressure drop along the well length from 1010 ft to the surface

Length (ft)	Depth (ft)	r_{in} (cm)	r_{out} (cm)	ΔP_F (psi)	ΔP_{KE}, (psi)	ΔP_{PE} (psi)	P (psi)
0		0.82					309.7
0 ~ 27	26.7	0.85	0.82	7.59	0.08	9.5	
27 ~ 53	53.3	0.89	0.85	6.32	0.06	9.5	
53 ~ 80	80	0.91	0.89	5.33	0.05	9.5	
80 ~ 91	91.3	0.27	0.91	1.93	-47.07	4.1	
91 ~ 103	102.7	0.43	0.27	711.81	39.90	4.1	
103 ~ 114	114	0.54	0.43	76.00	4.67	4.17	
114 ~ 263	263.3	1.52	0.54	310.03	2.81	53.4	
263 ~ 413	411.6	2.17	1.52	2.24	0.04	53.0	
413 ~ 562	558.4	3.03	2.17	0.41	0.01	52.5	
562 ~ 711	702	3.43	3.03	0.08	0	51.3	
711 ~ 861	844.8	3.52	3.43	0.05	0	51.1	
861 ~ 1010	986.6	3.54	3.52	0.04	0	50.7	
1010			3.54				1784.8

7. Conclusions and Future Work

7.1 Conclusions

1. We developed a model that can be used to predict the temperature distribution of an injection or production well, with or without gas-lift valves, in both permafrost and non-permafrost region, with or without considering wax deposition.

2. Insulating the wellbore effectively reduces the wellbore heat loss. For a well in permafrost region, insulating the conductor prevents the permafrost thawing near the ground surface. The model is capable of calculating the wellbore heat loss when only part of the tubing is insulated. Reducing the heat lost by gas convection in annulus is also a way to reduce the wellbore heat loss as shown in the thesis. Under the same production conditions, evacuating the annulus increases oil temperature at the wellhead 7 °F for a well without gas injection in the permafrost region after one year production.

3. Involving wax deposition, Singh's model is derived for laminar flow. Therefore, the model is valid only when the production rate is less than 168 B/D. At production rate 168B/D, the well is shut in after 10 days because the pressure is not sufficient to sustain the production rate following the wax deposition.

7.2 Future work

1. Since two-phase flow commonly occurs in most oil and gas wells, the study of pressure and temperature for two-phase flow shall improve the prediction accuracy and extend the model application.

2. Friction loss and kinetic energy play a role in temperature when fluid in turbulent flow or fluid velocity changes. Calculation of the change in enthalpy and kinetic energy will require the calculation of wellbore pressures from wellbore hydraulics considerations as well as an thorough understanding of gas and liquid composition and PVT behavior of the oil and gas system. It will be more realistic to utilize reservoir data together with available engineering correlations to calculate enthalpy and kinetic energy terms.

3. Due to the limitations of Singh's model, the model is valid only when the production rate is less than 168 B/D. Also, the constants in the wax solubility equation are from the sample oil in Singh's experiments. To apply this model for other oil samples, lab experiments and measurements are necessary to ensure more accurate results.

References

[1]   H. RAMEY JR, "Wellbore Heat Transmission," *Journal of Petroleum Technology,* vol. 14, pp. 427-435, 1962.

[2]   K. H. Bendiksen*, et al.*, "The Dynamic Two-Fluid Model OLGA: Theory and Application," *SPE Production Engineering,* 01/01/1991 1991.

[3]   G. H. Couto*, et al.*, "An Investigation of Two-Phase Oil/Water Paraffin Deposition," *SPE Production & Operations,* vol. 23, 2008.

[4]   http://en.wikipedia.org/wiki/Flow_assurance.

[5]   G. P. Willhite, "Over-all Heat Transfer Coefficients in Steam And Hot Water Injection Wells," *SPE Journal of Petroleum Technology,* vol. 19, 1967.

[6]   J. W. Galate and R. F. Mitchell III, "Downward Two-Phase Flow Effects in Heat-Loss and Pressure-Drop Modeling of Steam Injection Wells," presented at the SPE California Regional Meeting, Bakersfield, California, 1985.

[7]   A. R. Hasan and C. S. Kabir, "Aspects of Wellbore Heat Transfer During Two-Phase Flow (includes associated papers 30226 and 30970)," *SPE Production & Operations,* vol. 9, 1994.

[8]   A. R. Hasan and C. S. Kabir, "A Mechanistic Model for Computing Fluid Temperature Profiles in Gas-Lift Wells," *SPE Production & Operations,* vol. 11, 1996.

[9]   A. R. Hasan and C. S. Kabir, "Modeling Two-Phase Fluid and Heat Flows in Geothermal Wells," presented at the SPE Western Regional Meeting, San Jose, California, 2009.

[10]   I. N. Alves*, et al.*, "A Unified Model for Predicting Flowing Temperature Distribution. in Wellbores and Pipelines," *SPE Production Engineering,* vol. 7, 1992.

[11]   R. Sagar*, et al.*, "Predicting Temperature Profiles in a Flowing Well," *SPE Production Engineering,* vol. 6, 1991.

[12]   A. Sulaimon*, et al.*, "A proactive approach for predicting and preventing wax deposition in production tubing strings: A Niger Delta experience," *Journal of Petroleum and Gas Engineering Vol,* vol. 1, pp. 26-36, 2010.

[13]   Y. Sharma*, et al.*, "Simulation of Downhole Heater Phenomena in the Production of Wellbore Fluids," *SPE Production Engineering,* vol. 4, 1989.

[14]   H. S. Carslaw and J. C. Jaeger, "Conduction of heat in solids," *Oxford: Clarendon Press, 1959, 2nd ed.,* vol. 1, 1959.

[15]   http://cgc.rncan.gc.ca/permafrost/whatis_e.php.

[16]   M. Sengul, "Numerical solution of heat conduction with phase change in cylindrical systems," 1977.

[17]   E. Couch and J. Watts, "Permafrost thawing around producing oil wells," *Journal of Canadian Petroleum Technology,* vol. 9, 1970.

[18]   R. Merriam*, et al.*, "Insulated hot oil-producing wells in permafrost," *Journal of Petroleum Technology,* vol. 27, pp. 357-365, 1975.

[19]   C. K. Ekweribe*, et al.*, "Interim Report on Pressure Effect on Waxy-Crude Pipeline-Restart Conditions Investigated by a Model System," *SPE Projects, Facilities & Construction,* vol. 4, 2009.

[20] P. Singh*, et al.*, "Existence of a critical carbon number in the aging of a wax-oil gel," 2001.

[21] P. Singh, "Gel deposition of cold surfaces," University of Michigan, 2000.

[22] P. Singh*, et al.*, "Formation and aging of incipient thin film wax oil gels," *AIChE journal,* vol. 46, pp. 1059-1074, 2000.

[23] P. Singh*, et al.*, "Morphological evolution of thick wax deposits during aging," *AIChE journal,* vol. 47, pp. 6-18, 2001.

[24] J. Hsu and J. Brubaker, "Wax deposition scale-up modeling for waxy crude production lines," 1995.

[25] J. J. C. Hsu and J. P. Brubaker, "Wax Deposition Measurement and Scale-Up Modeling for Waxy Live Crudes under Turbulent Flow Conditions," presented at the International Meeting on Petroleum Engineering, Beijing, China, 1995.

[26] J. J. C. Hsu*, et al.*, "Wax Deposition of Waxy Live Crudes Under Turbulent Flow Conditions," presented at the SPE Annual Technical Conference and Exhibition, New Orleans, Louisiana, 1994.

[27] H. S. Lee, "Computational and rheological study of wax deposition and gelation in subsea pipelines," The University of Michigan, 2008.

[28] J. F. Keating and R. A. Wattenbarger, "The Simulation of Paraffin Deposition and Removal in Wellbores," presented at the SPE Western Regional Meeting, Long Beach, California, 1994.

[29] J. J. C. Hsu*, et al.*, "Validation of Wax Deposition Model by a Field Test," presented at the SPE International Oil and Gas Conference and Exhibition in China, Beijing, China, 1998.

[30] P. Singh*, et al.*, "An Application of Vacuum Insulation Tubing for Wax Control in an Arctic Environment," *SPE Drilling & Completion,* vol. 22, 2007.

[31] R. Bagatin*, et al.*, "Wax Modeling: There is Need for Alternatives," presented at the SPE Russian Oil and Gas Technical Conference and Exhibition, Moscow, Russia, 2008.

[32] C. Labes-Carrier*, et al.*, "Wax Deposition in North Sea Gas Condensate and Oil Systems: Comparison Between Operational Experience and Model Prediction," presented at the SPE Annual Technical Conference and Exhibition, San Antonio, Texas, 2002.

[33] F. Jessen and J. N. Howell, "Effect of flow rate on paraffin accumulation in plastic, steel, and coated pipe," *Petroleum Transactions, AIME,* vol. 213, pp. 80-84, 1958.

[34] C. S. Kabir*, et al.*, "Heat-Transfer Models for Mitigating Wellbore Solids Deposition," *SPE Journal,* vol. 7, 2002.

[35] A. J. Mansure and K. M. Barker, "Insights Into Good Hot Oiling Practices," presented at the SPE Production Operations Symposium, Oklahoma City, Oklahoma, 1993.

[36] D. E. Kenyon, "Model for Hot Oil Jobs," presented at the SPE Mid-Continent Operations Symposium, Oklahoma City, Oklahoma, 1999.

[37] M. J. Economides*, et al.*, "Petroleum production systems," 1994.

[38] N. H. Chen, "An explicit equation for friction factor in pipe," *Industrial & Engineering Chemistry Fundamentals,* vol. 18, pp. 296-297, 1979.

[39] T. Ertekin*, et al.*, *Basic applied reservoir simulation*: Society of Petroleum Engineers, 2001.

[40]   Y. Wu, "Study of Asphaltene on Wax Precipitation and Deposition," University of Kansas, 2012.

[41]   S. Paterson, "Propagation of a boundary of fusion," 1952, pp. 42-47.

Appendix A Parameters in Simulation

Reservoir Information

$c_t = 1.29 \times 10^{-5}$ psi^{-1}

$k = 8.2$ md

$h = 53$ ft

$P_i = 5651$ psi

$B = 1.1$ res bbl/STB

$\phi = 0.19$

$r_w = 0.2$ ft

Default Parameters in Simulation

Formation/Permafrost Properties

    Thermal conductivity of formation/frozen permafrost = 1.8, Btu/hr ft °F

    Heat capacity of the formation = 0.256, Btu/lb °F

    Formation density = 128.6, lb/ft^3

    Thermal conductivity of the thawed permafrost = 1.2, Btu/hr ft °F

    Heat capacity of the thawed permafrost = 0.347, Btu/lb °F

    Thawing temperature = 32, °F

    Latent Heat = 26, Btu/lb

Wellbore Properties

    Length = 16118, ft (The horizontal part stars from 11118ft)

    Vertical depth = 6162.4, ft

    Pipe relative roughness = 0.001

    Thermal conductivity of the cement = 1.0, Btu/hr ft °F

    Thermal conductivity of tubing = 25, Btu/hr ft °F

    Thermal conductivity of casing = 25, Btu/hr ft °F

The inside radius of the tubing = 0.116, ft

The outside radius of the tubing = 0.146, ft

The inside radius of the production casing = 0.262, ft

The outside radius of the production casing = 0.292, ft

The inside radius of the surface casing = 0.368, ft

The outside radius of the surface casing = 0.401, ft

The inside radius of the conductor = 0.625, ft

The outside radius of the conductor = 0.667, ft

The inside radius of the casing = 1.392, ft

The outside radius of the casing = 1.417, ft

The radius of the wellbore hole = 1.5, ft

The length of the insulator around the conductor is 80 ft

Thermal conductivity of insulator = 0.018, Btu/hr ft °F

Tubing Fluid Properties

Heat capacity = 0.8, Btu/lb °F

Thermal conductivity = 0.2, Btu/hr ft °F

Density = 51.48, lb/ft^3

Viscosity = 2, cp

Gas Properties

Heat capacity = 0.245, Btu/lb °F

Thermal conductivity = 0.0145, Btu/hr ft °F

Density = 0.0612, lb/ft^3

Viscosity = 0.0173, cp

Wax Properties

Thermal conductivity = 0.145, Btu/hr ft °F

WAT = 57, °F

Gel Density = 0.9, g/cm^3

Molecular diffusivity of wax in oil $= 1.48 \times 10^{-6}$, m^2/s

Production Information

Gas liquid ratio at the surface = 12000, SCF/STB

Initial gas liquid ratio = 990, SCF/STB

Other Initial Values

Production well:

Tubing fluid temperature at the wellbore bottomhole $T_f = 136.8$, ℉

Injected gas temperature in the annulus $T_g = 50$, ℉

Injection well:

Injected water temperature $T_f = 500$, ℉

Appendix B Simulation Results

B.1 Example of Temperature Calculation

Calculation of Injection Well

Table B-1 Calculated temperature distribution in the tubing
when 500 °F water is injected down tubing at the rate of 200 ft^3/hr after 30 days

Length (ft)	Vertical Depth (ft)	Twater (°F)	Te (°F)	Length (ft)	Vertical Depth (ft)	Twater (°F)	Te (°F)
0	0.0	500.0	50.0	4570	3009.1	378.4	91.2
163	163.0	494.7	52.2	4832	3133.1	372.9	92.9
326	325.5	489.4	54.5	5094	3263.7	367.5	94.7
489	486.8	484.2	56.7	5355	3396.2	362.3	96.5
652	645.1	479.1	58.8	5617	3529.5	357.2	98.4
815	801.0	474.1	61.0	5879	3660.9	352.2	100.2
978	956.2	469.2	63.1	6141	3789.4	347.4	101.9
1141	1107.4	464.4	65.2	6403	3918.3	342.7	103.7
1304	1253.8	459.7	67.2	6665	4050.4	338.1	105.5
1467	1390.7	455.0	69.1	6927	4181.6	333.7	107.3
1630	1516.7	450.4	70.8	7189	4313.5	329.3	109.1
1793	1634.6	445.9	72.4	7451	4443.2	325.1	110.9
1956	1741.2	441.4	73.9	7713	4570.2	321.0	112.6
2119	1832.9	437.0	75.1	7975	4698.5	317.0	114.4
2282	1915.2	432.7	76.2	8237	4828.6	313.1	116.2
2445	1994.6	428.5	77.3	8499	4955.2	309.3	117.9
2608	2073.0	424.3	78.4	8761	5080.1	305.7	119.6
2771	2151.5	420.2	79.5	9023	5201.8	302.1	121.3
2934	2230.0	416.1	80.6	9546	5443.9	295.3	124.6
3097	2308.0	412.1	81.6	9808	5566.0	292.0	126.3
3260	2385.2	408.2	82.7	10070	5689.4	288.8	127.9
3522	2513.2	401.9	84.4	10332	5816.8	285.8	129.7
3784	2642.3	395.8	86.2	10594	5960.6	282.8	131.7
4046	2765.4	389.9	87.9	10856	6074.9	279.9	133.2
4308	2887.5	384.1	89.6	11118	6162.4	277.1	134.4

Calculation of Production Well

I. Effect of Insulation on Temperature

Table B-2 Calculated oil and wellbore/formation interfacial temperature at 500 B/D
after 30 days production when the well is not insulated (Case I)

Length (ft)	Tf (℉)	Th (℉)	Length (ft)	Tf (℉)	Th (℉)
0	66.0	26.9	4308	104.5	73.8
23	66.2	27.1	4570	106.6	76.4
46	66.4	27.2	4832	108.7	79.1
68	66.6	27.4	5094	110.7	81.9
91	66.8	27.6	5355	112.6	84.6
114	67.0	27.8	5617	114.5	87.4
263	68.3	33.8	5879	116.3	90.2
413	69.6	34.8	6141	118.1	92.7
562	70.9	35.7	6403	119.8	95.3
711	72.2	36.6	6665	121.4	97.8
861	73.6	37.4	6927	123.0	100.4
1010	75.0	38.2	7189	124.5	102.9
1160	76.4	38.9	7451	125.9	105.4
1309	77.8	39.6	7713	127.3	107.8
1458	79.3	40.2	7975	128.6	110.1
1608	80.7	40.9	8237	129.8	112.5
1773	82.3	44.9	8499	130.9	114.7
1938	83.9	46.1	8761	131.9	116.9
2103	85.4	48.3	9023	132.9	119.0
2269	86.9	50.2	9284	133.8	121.1
2434	88.4	52.1	9546	134.6	123.2
2599	89.9	53.9	9808	135.2	125.2
2764	91.4	55.7	10070	135.8	127.4
2930	92.9	57.6	10332	136.3	129.8
3095	94.3	59.4	10594	136.6	132.0
3260	95.7	61.2	10856	136.8	133.9
3522	98.0	65.4	11118	136.8	136.8
3784	100.2	68.4	16118	136.8	136.8
4046	102.4	71.1			

Table B-3 Calculated oil and wellbore/formation interfacial temperature at 500 B/D
after 30 days production when 80ft long conductor is insulated (Case II)

Length (ft)	Tf (°F)	Th (°F)	Length (ft)	Tf (°F)	Th (°F)
0	66.6	21.1	4308	104.5	73.8
23	66.6	21.3	4570	106.6	76.4
46	66.7	21.5	4832	108.7	79.1
68	66.7	21.7	5094	110.7	81.9
91	66.8	27.6	5355	112.6	84.6
114	66.9	27.7	5617	114.5	87.4
263	67.0	27.8	5879	116.3	90.2
413	68.3	33.8	6141	118.1	92.7
562	69.6	34.8	6403	119.8	95.3
711	70.9	35.7	6665	121.4	97.8
861	72.2	36.6	6927	123.0	100.4
1010	73.6	37.4	7189	124.5	102.9
1160	75.0	38.2	7451	125.9	105.4
1309	76.4	38.9	7713	127.3	107.8
1458	77.8	39.6	7975	128.6	110.1
1608	79.3	40.2	8237	129.8	112.5
1773	80.7	40.9	8499	130.9	114.7
1938	82.3	44.9	8761	131.9	116.9
2103	83.9	46.1	9023	132.9	119.0
2269	86.9	50.2	9284	133.8	121.1
2434	88.4	52.1	9546	134.6	123.2
2599	89.9	53.9	9808	135.2	125.2
2764	91.4	55.7	10070	135.8	127.4
2930	92.9	57.6	10332	136.3	129.8
3095	94.3	59.4	10594	136.6	132.0
3260	95.7	61.2	10856	136.8	133.9
3522	98.0	65.4	11118	136.8	136.8
3784	100.2	68.4	16118	136.8	136.8
4046	102.4	71.1			

Table B-4 Calculated oil and wellbore/formation interfacial temperature at 500 B/D after 30 days production when 80ft long conductor and 1700ft long tubing are insulated (Case III)

Length (ft)	Tf (℉)	Th (℉)	Length (ft)	Tf (℉)	Th (℉)	Length (ft)	Tf (℉)	Th (℉)
0	76.2	21.0	1700	81.0	35.6	6141	118.1	92.7
27	76.2	21.2	1856	82.6	45.8	6403	119.8	95.2
53	76.3	21.4	2012	84.1	47.5	6665	121.4	97.8
80	76.3	21.6	2168	85.6	49.4	6927	123.0	100.3
91	76.3	23.2	2324	87.1	51.2	7189	124.5	102.9
103	76.4	23.3	2480	88.5	53.0	7451	125.9	105.3
114	76.4	23.4	2636	90.0	54.7	7713	127.3	107.7
263	76.9	26.1	2792	91.4	56.5	7975	128.5	110.1
413	77.3	27.2	2948	92.9	58.2	8237	129.7	112.4
562	77.8	28.2	3104	94.3	59.9	8499	130.9	114.7
711	78.2	29.3	3260	95.7	61.6	8761	131.9	116.9
861	78.7	30.3	3522	98.0	65.4	9023	132.9	119.0
1010	79.1	31.3	3784	100.2	68.3	9284	133.8	121.1
1160	79.5	32.4	4046	102.4	71.0	9546	134.5	123.2
1309	79.9	32.8	4308	104.5	73.7	9808	135.2	125.2
1458	80.4	33.2	4570	106.6	76.4	10070	135.8	127.4
1608	80.8	33.4	4832	108.7	79.0	10332	136.3	129.8
1626	80.8	35.2	5094	110.7	81.8	10594	136.6	132.0
1645	80.9	35.3	5355	112.6	84.6	10856	136.8	133.9
1663	80.9	35.4	5617	114.5	87.4	11118	136.8	136.8
1682	81.0	35.5	5879	116.3	90.1	16118	136.8	136.8

Table B-5 Calculated moving boundary location at 500 B/D after 30 days production

Length (ft)	Moving Boundary Location (ft)			Length (ft)	Moving Boundary Location (ft)		
	Case I	Case II	Case III		Case I	Case II	Case III
0	1.50	1.50	1.50	861	1.23	1.23	0.51
114	0.57	0.57	0.51	1010	1.37	1.37	0.51
263	0.69	0.69	0.51	1160	1.52	1.52	0.62
413	0.82	0.82	0.51	1309	1.65	1.65	0.77
562	0.95	0.95	0.51	1458	1.78	1.78	0.93
711	1.09	1.09	0.51	1608	1.91	1.91	1.07

Table B-6 Calculated oil and wellbore/formation interfacial temperature at 500 B/D
after one year production when 80ft long conductor insulated or not

Length (ft)	Tf (°F)		Length (ft)	Tf (°F)	
	Case I	Case II		Case I	Case II
0	70.3	70.9	4308	106.7	106.7
23	70.5	70.9	4570	108.7	108.7
46	70.7	71.0	4832	110.7	110.7
68	70.9	71.0	5094	112.5	112.5
91	71.1	71.1	5355	114.4	114.4
114	71.2	71.2	5617	116.1	116.1
263	72.5	72.5	5879	117.8	117.8
413	73.7	73.7	6141	119.5	119.5
562	74.9	74.9	6403	121.1	121.1
711	76.2	76.2	6665	122.6	122.6
861	77.4	77.5	6927	124.1	124.1
1010	78.7	78.7	7189	125.5	125.5
1160	80.0	80.0	7451	126.8	126.8
1309	81.3	81.3	7713	128.1	128.1
1458	82.7	82.7	7975	129.2	129.2
1608	84.1	84.1	8237	130.4	130.4
1773	85.5	85.5	8499	131.4	131.4
1938	87.0	87.0	8761	132.3	132.3
2103	88.5	88.5	9023	133.2	133.2
2269	90.0	90.0	9284	134.0	134.0
2434	91.4	91.4	9546	134.8	134.8
2599	92.8	92.8	9808	135.4	135.4
2764	94.2	94.3	10070	135.9	135.9
2930	95.6	95.6	10332	136.3	136.3
3095	97.0	97.0	10594	136.6	136.6
3260	98.4	98.4	10856	136.8	136.8
3522	100.5	100.5	11118	136.8	136.8
3784	102.6	102.6	16118	136.8	136.8
4046	104.7	104.7			

Table B-7 Calculated oil temperature at 500 B/D after 30 days production when 80ft long conductor and 1700ft long tubing are insulated (Case III)

Length (ft)	Tf (°F)	Length (ft)	Tf (°F)	Length (ft)	Tf (°F)
0	79.4	1700	84.3	6141	119.4
27	79.4	1856	85.8	6403	121.0
53	79.4	2012	87.2	6665	122.6
80	79.5	2168	88.6	6927	124.0
91	79.5	2324	90.1	7189	125.4
103	79.5	2480	91.5	7451	126.8
114	79.6	2636	92.8	7713	128.0
263	80.0	2792	94.2	7975	129.2
413	80.5	2948	95.6	8237	130.3
562	81.0	3104	96.9	8499	131.4
711	81.4	3260	98.3	8761	132.3
861	81.9	3522	100.4	9023	133.2
1010	82.3	3784	102.5	9284	134.0
1160	82.7	4046	104.6	9546	134.8
1309	83.2	4308	106.6	9808	135.4
1458	83.6	4570	108.6	10070	135.9
1608	84.0	4832	110.6	10332	136.3
1626	84.1	5094	112.5	10594	136.6
1645	84.1	5355	114.3	10856	136.8
1663	84.2	5617	116.1	11118	136.8
1682	84.2	5879	117.8	16118	136.8

II. Effect of GLV on Temperature

Case II: No gas was injected during the production

Case IV: Gas was injected during the production

Case V: The annulus was evacuated

Table B-8 Calculated tubing fluid and wellbore/formation interfacial temperature
at 500 B/D after 30 days

Length (ft)	Case II Tf (℉)	Case IV Tf (℉)	Case IV Tg (℉)	Case V Tf (℉)	Length (ft)	Case II Tf (℉)	Case IV Tf (℉)	Case IV Tg (℉)	Case V Tf (℉)
0	66.6	42.1	50.0	74.4	3885	102.4	77.3	76.4	107.1
27	66.6	40.4	45.6	74.5	4094	104.5	79.6	78.7	109.0
53	66.7	39.2	42.7	74.5	4302	106.6	81.9	81.0	110.9
80	66.7	38.4	40.7	74.6	4510	108.7	84.2	83.3	112.7
91	66.8	38.2	39.9	74.6	4719	110.7	86.5	85.6	114.5
103	66.9	38.0	39.3	74.7	4927	112.6	88.8	87.9	116.2
114	67.0	37.8	38.8	74.8	5136	114.5	91.1	90.2	117.8
263	68.3	38.3	38.1	76.0	5344	116.3	93.4	92.5	119.4
413	69.6	39.4	38.8	77.3	5714	118.1	97.3	96.4	120.9
562	70.9	40.7	40.0	78.5	6083	119.8	101.1	100.3	122.4
711	72.2	42.1	41.3	79.7	6453	121.4	104.8	104.0	123.8
861	73.6	43.5	42.7	81.0	6822	123.0	108.5	107.7	125.2
1010	75.0	45.0	44.2	82.3	7192	124.5	112.0	111.2	126.5
1160	76.4	46.6	45.7	83.6	7931	127.3	118.7	118.0	128.9
1309	77.8	48.2	47.4	84.9	8670	129.8	124.7	124.1	131.0
1458	79.3	50.0	49.1	86.2	9040	130.9	127.5	126.8	131.9
1608	80.7	51.9	50.9	87.6	9248	131.9	129.0	128.3	132.8
1773	82.3	53.8	52.9	89.0	9456	132.9	130.4	129.8	133.6
2103	85.4	57.6	56.7	91.8	9663	133.8	131.7	131.1	134.3
2269	86.9	59.4	58.5	93.2	9871	134.6	132.9	132.4	135.0
2599	89.9	63.1	62.2	95.9	10287	135.8	135.0	134.6	136.0
2764	91.4	64.9	64.1	97.3	10495	136.3	135.7	135.4	136.4
3095	94.3	68.6	67.7	99.9	10702	136.6	136.3	136.1	136.6
3260	95.7	70.3	69.5	101.2	10910	136.8	136.7	136.6	136.8
3522	98.0	72.7	71.8	103.2	11118	136.8	136.8	136.8	136.8
3784	100.2	75.0	74.1	105.2	16118	136.8	136.8	136.8	136.8

Table B-9 Calculated moving boundary location at 500 B/D after 30 days production

Length (ft)	Moving Boundary Location (ft)		
	Case II	Case IV	Case V
0	1.50	1.50	1.50
114	0.59	0.65	0.54
263	0.69	0.74	0.64
413	0.82	0.85	0.75
562	0.95	0.99	0.88
711	1.09	1.16	1.02
861	1.23	1.33	1.16
1010	1.37	1.52	1.30
1160	1.52	1.70	1.44
1309	1.65	1.89	1.57
1458	1.78	2.08	1.70
1608	1.91	2.25	1.82

Table B-10 Calculated permafrost temperature at 1608 ft along the wellbore
after 30 days with or without GLV (production rate 500 B/D)

Distance from the wellbore center (ft)	T (°F)			Distance from the wellbore center(ft)	T (°F)		
	Case II	Case IV	Case V		Case II	Case IV	Case V
0.51	45.6	40.9	39.8	2.33	32.0	31.9	31.9
0.71	42.5	38.6	37.8	2.45	32.0	31.9	31.9
0.91	40.3	37.0	36.2	2.70	31.9	31.9	31.9
1.17	37.9	35.3	34.7	2.84	31.9	31.9	31.8
1.30	36.9	34.5	34.0	3.13	31.9	31.8	31.8
1.50	35.7	33.6	33.2	4.00	31.8	31.7	31.7
1.65	34.8	32.9	32.6	4.20	31.8	31.7	31.7
1.92	33.4	32.0	32.0	6.21	31.6	31.6	31.6
2.01	33.0	32.0	32.0	10.65	31.5	31.5	31.5
2.22	32.1	31.9	31.9	15.76	31.4	31.4	31.4

Table B- 11 Calculated tubing fluid temperature at 500 B/D after one year (Case II)

Length (ft)	Tf (℉)	Length (ft)	Tf (℉)	Length (ft)	Tf (℉)
0	70.9	2269	90.0	6927	124.1
27	70.9	2434	91.4	7189	125.5
53	71.0	2599	92.8	7451	126.8
80	71.0	2764	94.3	7713	128.1
91	71.1	2930	95.6	7975	129.2
103	71.2	3095	97.0	8237	130.4
114	71.3	3260	98.4	8499	131.4
263	72.5	3522	100.5	8761	132.3
413	73.7	3784	102.6	9023	133.2
562	74.9	4046	104.7	9284	134.0
711	76.2	4308	106.7	9546	134.8
861	77.5	4570	108.7	9808	135.4
1010	78.7	4832	110.7	10070	135.9
1160	80.0	5094	112.5	10332	136.3
1309	81.3	5355	114.4	10594	136.6
1458	82.7	5617	116.1	10856	136.8
1608	84.1	5879	117.8	11118	136.8
1773	85.5	6141	119.5	16118	136.8
1938	87.0	6403	121.1		
2103	88.5	6665	122.6		

Table B-12 Calculated tubing fluid and gas temperature at 500 B/D after one year (Case IV)

Length (ft)	Tf ($^\circ$F)	Tg ($^\circ$F)	Length (ft)	Tf ($^\circ$F)	Tg ($^\circ$F)
0	46.8	50.0	2103	64.4	63.5
27	46.1	48.2	2269	66.2	65.3
53	45.7	47.0	2434	68.0	67.2
80	45.4	46.3	2599	69.9	69.0
91	45.3	45.9	2764	71.7	70.8
103	45.2	45.6	2930	73.4	72.6
114	45.2	45.4	3095	75.2	74.4
263	46.1	45.6	3260	77.0	76.2
413	47.2	46.6	4046	85.5	84.6
562	48.5	47.8	4832	93.9	93.1
711	49.8	49.1	5617	101.9	101.1
861	51.2	50.5	6403	109.6	108.8
1010	52.6	51.9	7189	116.7	116.0
1160	54.1	53.3	7975	123.1	122.5
1309	55.6	54.8	8761	128.8	128.2
1458	57.2	56.4	9546	133.5	133.1
1608	58.8	58.0	10332	136.6	136.3
1773	60.7	59.8	11118	136.8	136.8
1938	62.5	61.6	16118	136.8	136.8

Table B-13 Calculated tubing fluid temperature at 500 B/D after one year (Case IV)

Length (ft)	Tf (°F)	Length (ft)	Tf (°F)	Length (ft)	Tf (°F)
0	77.9	2269	95.6	6927	126.0
27	77.9	2434	96.9	7189	127.2
53	77.9	2599	98.2	7451	128.3
80	78.0	2764	99.5	7713	129.4
91	78.1	2930	100.7	7975	130.4
103	78.2	3095	102.0	8237	131.4
114	78.2	3260	103.2	8499	132.3
263	79.4	3522	105.1	8761	133.1
413	80.6	3784	107.0	9023	133.8
562	81.7	4046	108.9	9284	134.5
711	82.9	4308	110.7	9546	135.1
861	84.1	4570	112.5	9808	135.6
1010	85.2	4832	114.2	10070	136.1
1160	86.5	5094	115.9	10332	136.4
1309	87.7	5355	117.5	10594	136.6
1458	88.9	5617	119.0	10856	136.8
1608	90.2	5879	120.5	11118	136.8
1773	91.5	6141	122.0	16118	136.8
1938	92.9	6403	123.4		
2103	94.2	6665	124.7		

B.2 Example of Pressure Drop Calculation

Case I$_P$: Oil was produced at a constant flowrate 500 B/D

Case II$_P$: Oil was produced under a constant surface pressure 300 psig.

Case III$_P$: Oil was produced under a constant bottomhole pressure 2400 psig

Table B-14 Calculated pressure under different production conditions after 30 days

Length (ft)	Pressure (psi)			Length (ft)	Pressure (psi)		
	Case I$_P$	Case II$_P$	Case III$_P$		Case I$_P$	Case II$_P$	Case III$_P$
0	458.9	314.7	205.2	4046	1449.5	1304.6	1196.2
27	468.4	324.2	214.8	4308	1493.3	1348.3	1240.0
53	478.0	333.8	224.3	4570	1537.0	1391.9	1283.6
80	487.5	343.3	233.9	4832	1581.4	1436.3	1328.1
91	491.6	347.4	237.9	5094	1628.2	1483.1	1374.9
103	495.7	351.4	242.0	5355	1675.7	1530.5	1422.4
114	499.7	355.5	246.0	5617	1723.5	1578.2	1470.2
263	553.1	408.9	299.5	5879	1770.6	1625.3	1517.4
413	606.3	462.0	352.6	6141	1816.7	1671.3	1563.5
562	658.8	514.5	405.1	6403	1862.9	1717.5	1609.7
711	710.2	565.9	456.6	6665	1910.3	1764.8	1657.1
861	761.4	617.0	507.7	6927	1957.3	1811.8	1704.2
1010	812.1	667.7	558.5	7189	2004.6	1859.0	1751.4
1160	861.4	717.0	607.8	7451	2051.1	1905.5	1798.0
1309	909.3	764.8	655.7	7713	2096.6	1950.9	1843.5
1458	954.2	809.8	700.7	7975	2142.6	1996.9	1889.5
1608	996.0	851.5	742.4	8237	2189.3	2043.5	1936.2
1773	1039.2	894.7	785.7	8499	2234.7	2088.8	1981.6
1938	1078.3	933.7	824.8	8761	2279.4	2133.6	2026.4
2103	1112.3	967.7	858.7	9023	2323.1	2177.2	2070.1
2269	1142.3	997.7	888.8	9284	2366.7	2220.7	2113.7
2434	1171.3	1026.6	917.8	9546	2409.9	2263.9	2156.9
2764	1228.3	1083.6	974.8	9808	2453.7	2307.6	2200.7
2930	1256.8	1112.1	1003.3	10070	2497.9	2351.8	2245.0
3095	1285.2	1140.4	1031.7	10332	2543.6	2397.4	2290.7
3260	1313.2	1168.4	1059.8	10594	2595.1	2448.9	2342.2
3522	1359.1	1214.3	1105.7	10856	2636.1	2489.9	2383.3
3784	1405.4	1260.5	1152.0	11118	2667.5	2521.2	2414.7

Table B-15 Calculated flowrate in 30 days under different production conditions

Time (hr)	Q (B/D)		Time (hr)	Q (B/D)	
	Case II$_P$	Case III$_P$		Case II$_P$	Case III$_P$
0	864.3	895.4	20	665.1	688.2
1	809.1	837.9	30	646.0	668.4
2	780.0	807.6	40	633.0	654.9
3	760.6	787.4	50	623.2	644.7
4	746.2	772.4	60	615.4	636.6
5	734.8	760.6	70	608.9	629.8
6	725.4	750.9	100	594.3	614.8
7	717.5	742.6	150	578.6	598.4
8	710.7	735.5	200	567.9	587.3
9	704.7	729.3	250	559.8	579.0
10	699.3	723.7	300	553.4	572.3
11	694.5	718.7	350	548.1	566.8
12	690.1	714.2	400	543.6	562.1
13	686.1	710.0	450	539.7	558.1
14	682.5	706.2	500	536.2	554.5
15	679.1	702.7	550	533.1	551.3
16	675.9	699.4	600	530.3	548.4
17	672.9	696.3	650	527.7	545.7
18	670.1	693.4	700	525.4	543.3
19	667.5	690.7	720	524.5	542.4

Table B-16 Calculated moving boundary location at constant flowrate 500 B/D

Time (hr)	MB (ft)			Time (hr)	MB (ft)		
	Case I$_P$	Case II$_P$	Case III$_P$		Case I$_P$	Case II$_P$	Case III$_P$
1	0.61	0.61	0.61	60	0.78	0.82	0.83
2	0.61	0.62	0.62	70	0.81	0.85	0.86
3	0.62	0.62	0.62	80	0.83	0.88	0.88
4	0.62	0.63	0.63	90	0.86	0.90	0.91
5	0.62	0.63	0.63	100	0.88	0.93	0.94
10	0.64	0.65	0.65	200	1.08	1.14	1.15
15	0.66	0.67	0.67	300	1.25	1.32	1.33
20	0.67	0.69	0.69	400	1.40	1.47	1.48
30	0.70	0.73	0.73	500	1.53	1.60	1.62
40	0.73	0.76	0.76	600	1.65	1.72	1.74
50	0.76	0.79	0.80	720	1.78	1.85	1.87

B.3 Example of Wax Deposition Calculation

Wax Deposition at Different Flowrate

Table B-17 Calculated effective inner tubing radius at different flowrates after 10 days

Length (ft)	Effective Radius (cm)			
	168 (B/D)	300 (B/D)	350 (B/D)	400 (B/D)
0	0.28	1.34	2.13	3.09
27	0.27	1.34	2.14	3.10
53	0.27	1.35	2.14	3.11
80	0.26	1.36	2.15	3.12
91	0.32	1.31	2.08	2.93
103	0.34	1.33	2.10	2.98
114	0.35	1.35	2.12	3.03
263	0.55	1.64	2.42	3.41
413	0.71	1.91	2.73	3.52
562	0.87	2.18	3.04	3.54
711	1.03	2.47	3.43	3.54
861	1.19	2.76	3.52	3.54
1010	1.36	3.06	3.54	3.54
1160	1.54	3.43	3.54	3.54
1309	1.73	3.53	3.54	3.54
1458	1.95	3.54	3.54	3.54
1608	2.20	3.54	3.54	3.54
1773	2.49	3.54	3.54	3.54
1938	2.80	3.54	3.54	3.54
2103	3.10	3.54	3.54	3.54
2269	3.41	3.54	3.54	3.54
2434	3.53	3.54	3.54	3.54
2599	3.54	3.54	3.54	3.54

Table B-18 Calculated temperature at wax/oil interface at different flowrates after 10 days

Length (ft)	Temperature (°F)				Length (ft)	Temperature (°F)			
	168 (B/D)	300 (B/D)	350 (B/D)	400 (B/D)		168 (B/D)	300 (B/D)	350 (B/D)	400 (B/D)
0	42.6	51.0	54.0	57.3	4046	77.2	88.7	92.3	95.3
27	42.7	51.0	54.0	57.4	4308	80.1	91.4	94.8	97.8
53	42.7	51.1	54.1	57.4	4570	83.0	94.0	97.3	100.1
80	42.7	51.1	54.1	57.4	4832	85.9	96.6	99.8	102.5
91	42.7	51.0	53.9	57.0	5094	88.8	99.2	102.2	104.8
103	42.8	51.1	53.9	57.1	5355	91.6	101.7	104.6	107.0
114	42.8	51.1	54.0	57.1	5617	94.4	104.1	106.9	109.2
263	43.4	51.8	54.9	58.2	5879	97.2	106.5	109.1	111.3
413	44.1	52.7	55.9	59.5	6141	99.9	108.9	111.3	113.4
562	44.8	53.5	57.0	60.9	6403	102.6	111.1	113.5	115.4
711	45.5	54.5	58.1	62.2	6665	105.3	113.4	115.5	117.4
861	46.3	55.5	59.5	63.6	6927	107.9	115.5	117.6	119.2
1010	47.1	56.7	60.9	65.1	7189	110.4	117.6	119.5	121.0
1160	48.0	58.0	62.4	66.6	7451	112.9	119.6	121.4	122.8
1309	48.9	59.5	64.0	68.1	7713	115.4	121.6	123.1	124.4
1458	49.9	61.1	65.6	69.7	7975	117.7	123.4	124.8	126.0
1608	51.1	62.8	67.3	71.3	8237	120.0	125.2	126.5	127.5
1773	52.5	64.7	69.1	73.1	8499	122.2	126.9	128.0	128.9
1938	53.9	66.5	70.8	74.7	8761	124.4	128.5	129.4	130.2
2103	55.5	68.3	72.6	76.4	9023	126.4	130.0	130.8	131.4
2269	57.2	70.1	74.3	78.1	9284	128.4	131.4	132.0	132.5
2434	59.0	71.9	76.1	79.8	9546	130.3	132.6	133.2	133.5
2599	60.9	73.7	77.8	81.4	9808	132.0	133.8	134.2	134.5
2764	62.8	75.4	79.5	83.1	10070	133.6	134.8	135.1	135.3
2930	64.7	77.2	81.1	84.7	10332	134.9	135.6	135.8	135.9
3095	66.5	78.9	82.8	86.3	10594	135.9	136.3	136.3	136.4
3260	68.4	80.6	84.5	87.9	10856	136.6	136.7	136.7	136.7
3522	71.3	83.3	87.1	90.4	11118	136.8	136.8	136.8	136.8
3784	74.2	86.0	89.7	92.9	16118	136.8	136.8	136.8	136.8

Table B-19 Calculated oil temperature at 400 B/D after 10 days and one year

Length (ft)	Temperature (°F)		Length (ft)	Temperature (°F)	
	10 days	One year		10 days	One year
0	62.8	59.0	4046	99.6	97.3
27	62.9	59.0	4308	101.9	99.7
53	62.9	59.1	4570	104.1	102.0
80	63.0	59.1	4832	106.3	104.3
91	63.0	59.2	5094	108.4	106.5
103	63.1	59.3	5355	110.5	108.7
114	63.2	59.3	5617	112.5	110.8
263	64.5	60.6	5879	114.5	112.9
413	65.8	62.0	6141	116.3	114.9
562	67.1	63.3	6403	118.2	116.8
711	68.4	64.7	6665	119.9	118.7
861	69.8	66.2	6927	121.6	120.5
1010	71.1	67.6	7189	123.3	122.2
1160	72.5	69.1	7451	124.8	123.9
1309	74.0	70.6	7713	126.3	125.4
1458	75.4	72.2	7975	127.7	126.9
1608	76.9	73.8	8237	129.0	128.3
1773	78.5	75.4	8499	130.2	129.7
1938	80.1	77.1	8761	131.4	130.9
2103	81.8	78.8	9023	132.4	132.0
2269	83.3	80.4	9284	133.4	133.1
2434	84.9	82.1	9546	134.3	134.0
2599	86.5	83.7	9808	135.1	134.9
2764	88.0	85.3	10070	135.7	135.6
2930	89.6	86.9	10332	136.2	136.2
3095	91.1	88.5	10594	136.6	136.5
3260	92.6	90.0	10856	136.8	136.8
3522	95.0	92.5	11118	136.8	136.8
3784	97.3	94.9	16118	136.8	136.8

Effect of Wax Deposition on Oil Temperature and Pressure Calculation

Case I_W: Without considering wax deposition

Case II_W: Consider wax deposition

Table B-20 Calculated effective tubing radius and temperature at inner tubing surface at 350 B/D after 16 days

Length (ft)	Effective radius (cm)	Temperature (°F)		Length (ft)	Effective radius (cm)	Temperature (°F)	
		Case I_W	Case II_W			Case I_W	Case II_W
0	0.81	53.6	49.0	4046	3.54	92.7	92.7
27	0.84	53.6	49.2	4308	3.54	95.2	95.2
53	0.87	53.7	49.4	4570	3.54	97.7	97.7
80	0.90	53.7	49.5	4832	3.54	100.2	100.2
91	0.23	53.1	31.2	5094	3.54	102.6	102.6
103	0.40	53.2	32.6	5355	3.54	104.9	104.9
114	0.52	53.3	33.5	5617	3.54	107.2	107.2
263	1.51	54.6	40.3	5879	3.54	109.5	109.5
413	2.17	56.0	45.5	6141	3.54	111.6	111.6
562	3.03	57.3	53.0	6403	3.54	113.8	113.8
711	3.43	58.7	57.7	6665	3.54	115.8	115.8
861	3.52	60.1	60.0	6927	3.54	117.8	117.8
1010	3.54	61.6	61.6	7189	3.54	119.7	119.7
1160	3.54	63.1	63.1	7451	3.54	121.6	121.6
1309	3.54	64.6	64.6	7713	3.54	123.3	123.3
1458	3.54	66.2	66.2	7975	3.54	125.0	125.0
1608	3.54	67.9	67.9	8237	3.54	126.6	126.6
1773	3.54	69.7	69.7	8499	3.54	128.1	128.1
1938	3.54	71.4	71.4	8761	3.54	129.5	129.5
2103	3.54	73.2	73.2	9023	3.54	130.9	130.9
2269	3.54	74.9	74.9	9284	3.54	132.1	132.1
2434	3.54	76.6	76.6	9546	3.54	133.2	133.2
2599	3.54	78.3	78.3	9808	3.54	134.2	134.2
2764	3.54	80.0	80.0	10070	3.54	135.1	135.1
2930	3.54	81.7	81.7	10332	3.54	135.8	135.8
3095	3.54	83.3	83.3	10594	3.54	136.3	136.3
3260	3.54	85.0	85.0	10856	3.54	136.7	136.7
3522	3.54	87.6	87.6	11118	3.54	136.8	136.8
3784	3.54	90.2	90.2	16118	3.54	136.8	136.8

Table B-21 Calculated oil pressure at 350 B/D after 16 days

Length (ft)	Pressure (psi)		Length (ft)	Pressure (psi)	
	Case I$_W$	Case II$_W$		Case I$_W$	Case II$_W$
0	1432	310	4046	2422	2422
27	1442	327	4308	2466	2465
53	1451	343	4570	2509	2509
80	1461	358	4832	2554	2553
91	1465	317	5094	2600	2600
103	1469	1072	5355	2648	2647
114	1473	1157	5617	2695	2695
263	1526	1523	5879	2742	2742
413	1579	1579	6141	2788	2788
562	1632	1632	6403	2835	2834
711	1683	1683	6665	2882	2882
861	1734	1734	6927	2929	2929
1010	1785	1785	7189	2976	2976
1160	1834	1834	7451	3023	3022
1309	1882	1882	7713	3068	3068
1458	1927	1927	7975	3114	3114
1608	1969	1968	8237	3161	3160
1773	2012	2012	8499	3206	3206
1938	2051	2051	8761	3251	3250
2103	2085	2085	9023	3294	3294
2269	2115	2115	9284	3338	3337
2434	2144	2144	9546	3381	3381
2599	2172	2172	9808	3425	3424
2764	2201	2201	10070	3469	3468
2930	2229	2229	10332	3514	3514
3095	2258	2257	10594	3566	3566
3260	2286	2285	10856	3607	3607
3522	2332	2331	11118	3638	3638
3784	2378	2377			

Appendix C

```
======================================================================
                Main program for Heat Transfer Calculation
======================================================================
function Result = TempSimu(B)

global Kcem Ktub Kcas Kins rh r_inTubing r_outTubing r_inProdcasing r_outProdcasing
r_inInsulator r_outInsulator Ks Kl Cs Cl Form_Dens Latent_H Ko Co Massrate_o Cg
Massrate_g Dens_g Visco_o Dens_o Visco_g Kg C_mix Massrate_mix Q_inj Timepace Length
Lenpace Delta_depth VDepth Depth Num_z Num_r Num GLV Num_GLV WoNGLV Welltype
Tw Te Tf Tg Uco hci hr hti hto rto Uwi Kwax Flowrate_mix WoNWax ResPVara_1
ResPVara_2 GelDens Cwbo Sa Sb Sc Dwo alpha Qg2 WAT

x = xlsread('WellboreProfile','Length');
y = xlsread('WellboreProfile','Depth');
[r,c] = size(x);
A = real(B);


Welltype = A(89); % "0" = Injection well; "1" = Production well
WoNGLV = A(90); % "0" = Without GLV; "1" = With GLV
WoNWax = A(125); % "0" = Without wax; "1" = With wax
if A(92) == 1 % "0" = in non-permafrost region; "1" = in permafrost region
    for i = 1:r
        if A(12) <= y(i)    PermLength = (x(i)-x(i-1)) * (A(12) - y(i-1)) / (y(i) - y(i-1)) + x(i-1);
            break; end
    end
end
%_____Initial Data_____
% Parameters: density of the formation, latent heat, thawing temperature, thermal conductivity
% and heat capacity of frozen/thawed regions, thermal conductivity of tubing, cement, mud, casing
% and insulation
Form_Dens = A(24); Latent_H = A(27); Tw = 32;
Ks = A(26); Cs = A(25); Kl = A(29); Cl = A(28);
Ktub = A(20); Kcem = A(21); Kmud = A(22); Kcas = A(20); Kins = A(23);
% Wellbore configurations: r_in (radius at inside surface), r_out (radius at outside surface)
r_inTubing = A(51); r_outTubing = A(52);
r_inProdcasing = A(55); r_outProdcasing = A(56);
r_inSurfcasing = A(59); r_outSurfcasing = A(60);
inConductor = A(63); r_outConductor = A(64);
r_inCasing = A(71); r_outCasing = A(72);
r_inInsulator = A(67); r_outInsulator = A(68);
```

```matlab
% Insulator around tubing
r_inAdd = A(75); r_outAdd = A(76); AddLength = A(78);
%_____
% Parameters of oil/gas: Heat capacity, thermal conductivity, density and viscosity
Co = A(1); Ko = A(2); Dens_o = A(3); Visco_o = A(4)*2.4191;
Cg = A(5); Kg = A(6); Dens_g = A(7); Visco_g = A(8)*2.4191;
%_____
%Heat transfer coefficient by radiation
hr = 1.1;
%_____
%Parameters for wax: Thermal conductivity, density, wax concentration, a, b, c
Kwax = A(116); GelDens = A(117); Cwbo = A(118);
Sa = A(119); Sb = A(120) ; Sc = A(121);
Dwo = A(122); alpha = A(123); WAT = (A(9)-32)*5/9;
%_____
%Parameters for reservoir pressure calculation
ResPress = A(130);
ResPVara_1 = 162.6 * A(126) * A(4) /(A(128) * A(129));
ResPVara_2 = log10(A(129)/(A(131)*A(4)*A(127)*A(81)^2)) - 3.23;

Time = A(30); % Simulation time, hr
Timepace = 1; ;
Num_t = Time / Timepace + 1;
%_____
% Find how many gas lift valves the wellbore has
i = 0;
for j = 1:10
    if A(30+j) ~= 0
        i = i + 1;
        GLVDepth(i) = A(30+j);
        GLVRatio(i) = A(40+j)/100;
    end
end
Num_GLV = i;
%_____
% Grid numbering depth direction
if A(92) == 1
    if A(74) ~= 0
        if A(74) ~= A(66)
            MDepth(1)= A(74); % Mdepth(casing);
            MDepth(2)= A(66); % Mdepth(conductor);
            MDepth(3)= PermLength; % Mdepth(permafrost);
            MDepth(4)= A(62); % Mdepth(surface casing);
                if WoNGLV == 1
```

```matlab
            if A(54) ~= GLVDepth(Num_GLV)
                numn = 5 + Num_GLV; MDepth(5:numn-1)= GLVDepth(1:Num_GLV);
            else
                numn = 4 + Num_GLV;
                MDepth(5:numn)= GLVDepth(1:Num_GLV);
            end
            N(1:2) = 3; N(3:numn) = 10;
            for i = 1:Num_GLV
                GLV(i) = sum(N(1:4+i))+1; % The total grid number from the top to the GLV
            end
        else
            numn = 5; N = [3 3 10 10 30];
        end
    else
        MDepth(1)= A(74); MDepth(2)= PermLength; MDepth(3)= A(62);
        if WoNGLV == 1
            if A(54) ~= GLVDepth(Num_GLV)
                numn = 4 + Num_GLV; MDepth(4:numn-1)= GLVDepth(1:Num_GLV);
            else
                numn = 3 + Num_GLV; MDepth(4:numn)= GLVDepth(1:Num_GLV);
            end
            N(1) = 6; N(2:numn) = 10;
            for i = 1:Num_GLV
                GLV(i) = sum(N(1:3+i))+1;
            end
        else
            numn = 4;
            N = [6 10 10 30];
        end
    end
end
else
    MDepth(1)= A(66); MDepth(2)= PermLength; MDepth(3)= A(62);
    if WoNGLV == 1
        if A(54) ~= GLVDepth(Num_GLV)
            numn = 4 + Num_GLV; MDepth(4:numn-1)= GLVDepth(1:Num_GLV);
        else
            numn = 3 + Num_GLV; MDepth(4:numn)= GLVDepth(1:Num_GLV);
        end
        N(1:numn) = 10;
        for i = 1:Num_GLV
            GLV(i) = sum(N(1:3+i))+1; % The total grid number from the top to the GLV
        end
    else
        numn = 4; N = [5 10 10 30];
```

```matlab
            end
        end
    else
        MDepth(1)= A(62);
        if WoNGLV == 1
            if A(54) ~= GLVDepth(Num_GLV)
                numn = 2 + Num_GLV; MDepth(2:numn-1)= GLVDepth(1:Num_GLV);
            else
                numn = 1 + Num_GLV; MDepth(2:numn)= GLVDepth(1:Num_GLV);
            end
            N(1:numn) = 10;
            for i = 1:Num_GLV
                GLV(i) = sum(N(1:1+i))+1;
            end
        else
            numn = 2;
            N = [20 30];
        end
    end
    MDepth(numn)= A(54); % the whole length

    if A(75) ~= 0 % Add insulation to tubing
        for i = 1:numn
            if AddLength <= MDepth(i)
                WHERE = i;
                if AddLength == MDepth(i)
                    AddDot = 0; break;
                else
                    for j = numn:-1:i
                        MD(j+1) = MDepth(j); N(j+1) = N(j);
                    end
                    MD(1:i-1) = MDepth(1:i-1); MD(i) = AddLength; AddDot = i; N(i) = 5;
                    numn = numn + 1; break;
                end
            end
        end
    else
        AddDot = 0;
    end

    if AddDot == 0     MD = MDepth;    end

    for i = 1:numn
        if i == 1 M(1) = N(1) + 1; else M(i)= N(i) + M(i-1); end
```

```
    if i == AddDot AddM = M(i); end
  end

Num_z = M(numn); Depth(1) = 0;
for i = 2:Num_z
  if i <= M(1)
    if i < M(1)
      Delta_depth(1:(M(1)-1)) = MD(1) / N(1); Depth(i) = Depth(i-1) + Delta_depth(i-1);
    else
      Depth(i) = MD(1);
    end
  else
    for j = 2:M(numn)
      if i <= M(j)
        if i < M(j)
          Delta_depth(M(j-1):M(j)) = (MD(j) - MD(j-1)) / N(j);
          Depth(i) = Depth(i-1) + Delta_depth(i-1);
        else
          Depth(i) = MD(j);
        end
        break;
      end
    end
  end
end
Depth(Num_z) = A(54);
%_____
% Geothermal temperature calculation
for i = 1: Num_z
  if Depth(i) <= A(62)
    if (A(92) == 1) && (Depth(i) <= A(66))
      rh(i) = A(65); % Drillhole radius around conductor
    else
      rh(i) = A(61); % Drillhole radius around surface casing
    end
  else
  rh(i) = A(57); % Drillhole radius around production casing
  end
end

if A(84) == 0 % "0" Using a/b mode to calculate geothermal T; "1" interpolation
  for i = 1:10
    if A(105+i) ~= 0
      GeoDepth(i) = A(95+i); GeoTemp(i) = A(105+i);
```

```matlab
            end
        end
    [a,GeoM] = size(GeoDepth);
end

VDepth(1) = y(1);
for i = 2:Num_z
    for j = 1:r
        if Depth(i) <= x(j)
            VDepth(i) = (y(j) - y(j-1)) * (Depth(i) - x(j-1)) / (x(j) - x(j-1)) + y(j-1);
            break;
        end
    end
end
Inflexion = 0;
for i = 1:Num_z
    if A(84) == 1
        Te(i) = A(10) * VDepth(i) + A(11); % Geothermal gradient and surface temperature;
            if A(85) == 1 % In permafrost region
                if VDepth(i) <= A(12)
                    if VDepth(i) == A(12)
                        Inflexion = i;
                    end
                Te(i) = A(82) * VDepth(i) + A(83); % Geothermal gradient and surface temperature
                end
            end
    else
        if VDepth(i) <= GeoDepth(1)
            Te(i) = (GeoTemp(2) - GeoTemp(1)) * (VDepth(i) - GeoDepth(1)) / (GeoDepth(2) -
GeoDepth(1)) + GeoTemp(1);
        elseif VDepth(i) > GeoDepth(GeoM)
            Te(i) = (GeoTemp(GeoM) - GeoTemp(GeoM-1)) * (VDepth(i) - GeoDepth(GeoM-1)) /
(GeoDepth(GeoM) - GeoDepth(GeoM-1)) + GeoTemp(GeoM-1);
        else
            for t = 1:GeoM
                if VDepth(i) <= GeoDepth(t)
                    Te(i) = (GeoTemp(t) - GeoTemp(t-1)) * (VDepth(i) - GeoDepth(t-1)) / (GeoDepth(t)
- GeoDepth(t-1)) + GeoTemp(t-1);
                    break;
                end
            end
        end
    end
end
```

```matlab
if A(84) == 1
   if Inflexion ~= 0
      tt = [VDepth(Inflexion-3), VDepth(Inflexion-2), VDepth(Inflexion-1), VDepth(Inflexion+1),
VDepth(Inflexion+2), VDepth(Inflexion+3)];
      pp = [Te(Inflexion-3), Te(Inflexion-2), Te(Inflexion-1), Te(Inflexion+1), Te(Inflexion+2),
Te(Inflexion+3)];
      Te(Inflexion) = interp1(tt, pp, A(12), 'spline');
   end
end
%_____
% Grid numbering in radial direction
Distance = 200; Num_r = 101;
Accterm = ((Distance + max(rh)) / max(rh)) ^ (1 / (Num_r-1));
Len(1) = max(rh); Lenp(1) = 0;
for j = 2:Num_r
   Len(j) = Len(1) * Accterm^(j-1); Lenp(j) = Len(j) - Len(j-1);
end

if A(92) == 1
   for i = 1:Num_z
      if (A(74) ~= A(66)) && (A(74) ~= 0)
         if Depth(i) <= MDepth(2)
            Length(i,1:Num_r) = Len(1:Num_r); Lenpace(i,1:Num_r) = Lenp(1:Num_r);
         elseif (Depth(i) > MDepth(2)) && (Depth(i) <= MDepth(4))
            Delta1 = (A(65) - A(61)) / 15; Length(i,1:15) = A(61) + Delta1 * (0:14);
            Lenpace(i,1) = Lenp(1); Lenpace(i,2:16) = Delta1;
            Length(i,16:Num_r) = Len(1:(Num_r-15));
            Lenpace(i,17:Num_r) = Lenp(2:(Num_r-15));
         else
            Delta2 = (A(61) - A(57)) / 5;
            Length(i,1:5) = A(57) + Delta2 * (0:4); Lenpace(i,1) = Lenp(1);
            Lenpace(i,2:6) = Delta2; Length(i,6:20) = A(61) + Delta1 * (0:14);
            Lenpace(i,7:21) = Delta1; Length(i,21:Num_r) = Len(1:(Num_r-20));
            Lenpace(i,22:Num_r) = Lenp(2:(Num_r-20));
         end
      else
         if Depth(i) <= MDepth(1)
            Length(i,1:Num_r) = Len(1:Num_r); Lenpace(i,1:Num_r) = Lenp(1:Num_r);
         elseif (Depth(i) > MDepth(1)) && (Depth(i) <= MDepth(3))
            Delta1 = (A(65) - A(61)) / 15; Length(i,1:15) = A(61) + Delta1 * (0:14);
            Lenpace(i,1) = Lenp(1); Lenpace(i,2:16) = Delta1;
            Length(i,16:Num_r) = Len(1:(Num_r-15));
            Lenpace(i,17:Num_r) = Lenp(2:(Num_r-15));
         else
```

```matlab
            Delta2 = (A(61) - A(57)) / 5; Length(i,1:5) = A(57) + Delta2 * (0:4);
            Lenpace(i,1) = Lenp(1); Lenpace(i,2:6) = Delta2;
            Length(i,6:20) = A(61) + Delta1 * (0:14); Lenpace(i,7:21) = Delta1;
            Length(i,21:Num_r) = Len(1:(Num_r-20));
            Lenpace(i,22:Num_r) = Lenp(2:(Num_r-20));
         end
      end
   end
else
   for i = 1:Num_z
      if Depth(i) <= MDepth(1)
         Length(i,1:Num_r) = Len(1:Num_r); Lenpace(i,1:Num_r) = Lenp(1:Num_r);
      else
         Delta1 = (A(61) - A(57)) / 8; Length(i,1:8) = A(57) + Delta1 * (0:7);
         Lenpace(i,1) = Lenp(1); Lenpace(i,2:9) = Delta1;
         Length(i,9:Num_r) = Len(1:(Num_r-8)); Lenpace(i,10:Num_r) = Lenp(2:(Num_r-8));
      end
   end
end
%_____
%Uco
if A(92) == 1
   if A(74) == 0
      R1 = log(A(65)/r_outConductor) + log(r_inConductor/r_outSurfcasing) +
log(r_inSurfcasing/r_outProdcasing);
      R2 = log(r_outConductor/r_inConductor) + log(r_outSurfcasing/r_inSurfcasing);
      R3 = log(A(61)/r_outSurfcasing) + log(r_inSurfcasing/r_outProdcasing);
      R4 = log(r_outSurfcasing/r_inSurfcasing);
      Uco(1:M(1)) = (r_outProdcasing * (R1/Kcem + R2/Kcas))^(-1);
      Uco(M(1)+1:M(3)) = (r_outProdcasing * (R3/Kcem + R4/Kcas))^(-1);
      Uco(M(3)+1:Num_z) = (r_outProdcasing * log(A(57)/r_outProdcasing) / Kcem)^(-1);
   else
      if A(74) ~= A(66)
       R1 = log(A(65)/r_outCasing) + log(r_inConductor/r_outSurfcasing) +
log(r_inSurfcasing/r_outProdcasing);
       R2 = log(r_outCasing/r_inCasing) + log(r_outConductor/r_inConductor) +
log(r_outSurfcasing/r_inSurfcasing);
       R3 = log(r_inCasing/r_outConductor);
       R4 = log(A(65)/r_outConductor) + log(r_inConductor/r_outSurfcasing) +
log(r_inSurfcasing/r_outProdcasing);
       R5 = log(r_outConductor/r_inConductor) + log(r_outSurfcasing/r_inSurfcasing);
       R6 = log(A(61)/r_outSurfcasing) + log(r_inSurfcasing/r_outProdcasing);
       R7 = log(r_outSurfcasing/r_inSurfcasing);
       Uco(1:M(1)) = (r_outProdcasing * (R1/Kcem + R2/Kcas + R3/Kins))^(-1);
```

```matlab
            Uco(M(1)+1:M(2)) = (r_outProdcasing * (R4/Kcem + R5/Kcas))^(-1);
            Uco(M(2)+1:M(4)) = (r_outProdcasing * (R6/Kcem + R7/Kcas))^(-1);
            Uco(M(4)+1:Num_z) = (r_outProdcasing * log(A(57)/r_outProdcasing) / Kcem)^(-1);
        else
            R1 = log(A(65)/r_outCasing) + log(r_inConductor/r_outSurfcasing) +
log(r_inSurfcasing/r_outProdcasing);
            R2 = log(r_outCasing/r_inCasing) + log(r_outConductor/r_inConductor) +
log(r_outSurfcasing/r_inSurfcasing);
            R3 = log(r_inCasing/r_outConductor);
            R4 = log(A(61)/r_outSurfcasing) + log(r_inSurfcasing/r_outProdcasing);
            R5 = log(r_outSurfcasing/r_inSurfcasing);
            Uco(1:M(1)) = (r_outProdcasing * (R1/Kcem + R2/Kcas + R3/Kins))^(-1);
            Uco(M(1)+1:M(3)) = (r_outProdcasing * (R4/Kcem + R5/Kcas))^(-1);
            Uco(M(3)+1:Num_z) = (r_outProdcasing * log(A(57)/r_outProdcasing) / Kcem)^(-1);
        end
    end
else
    R1 = log(A(61)/r_outSurfcasing) + log(r_inSurfcasing/r_outProdcasing);
    R2 = log(r_outSurfcasing/r_inSurfcasing);
    Uco(1:M(1)) = (r_outProdcasing * (R1/Kcem + R2/Kcas))^(-1);
    Uco((M(1)+1):Num_z) = (r_outProdcasing * log(A(57)/r_outProdcasing) / Kcem)^(-1);
end
%_____
% Calculation
if WoNGLV == 1 Num = Num_z * (Num_r+1) - 2;
else Num = Num_z * Num_r - 1; end

if A(92) == 1
    for i = 1:Num_z
        if VDepth(i) > A(12)
            MPerm = i-1;
            break;
        end
    end
    Check(1:MPerm) = 0; V_old(1:MPerm) = 0.2;
    MB(1:MPerm) = rh(1:MPerm) + 0.01; MDS(1:MPerm) = 0;
end
for i = 1:Num_t     MBLocation(1:Num_z,i) = rh(1:Num_z); end
for i = 1:Num_z     Formation_temp(i,1:Num_r) = Te(i); end
Dens_mix(1:Num_z) = Dens_o;
Visco_mix(1:Num_z) = Visco_o;
WAPInfom(1,:) = [0,r_inTubing * 30.5];
WaxConcentration(1:Num_z,1:Num_t) = Cwbo;
WaxThickness(1:Num_z,1:Num_t) = r_inTubing * (1-0.00001)*30.5;
```

117

```matlab
WaxFraction(1:Num_z,1:Num_t) = 0.0067;
Wax = [WaxConcentration(:,1),WaxThickness(:,1),WaxFraction(:,1)];
rti(1:Num_z) = r_inTubing;
S1(1:Num_z) = Length(:,1);
for n = 1:Num_t
   a = A(133); % 0:Constant Flowrate; 1: constant Surface Pressure; 2: Constant Bottomhole
Pressure
   t = n * Timepace;
   if Welltype == 0
      if A(3) == 0
         Flowrate_mix(1:Num_z) = A(13) * 1000 / 24; % Injection rate
         Flowrate_g(1:Num_z) = 0;
         Massrate_mix(1:Num_z) = Flowrate_mix(1:Num_z) * Dens_g; % mixture mass flow rate
         C_mix(1:Num_z) = A(5);
      else
         if A(3) == 62.4;
            Flowrate_mix(1:Num_z) = A(13);
         else
            Flowrate_mix(1:Num_z) = A(13) * 5.6 / 24;
         end
         Flowrate_g(1:Num_z) = 0; Massrate_mix(1:Num_z) = Flowrate_mix(1:Num_z) * A(3);
         C_mix(1:Num_z) = A(1);
      end
      Pressure(1:Num_z,n)=zeros; Tf = A(14); % Injection temperature
   else
      if WoNGLV == 0
         if a ~= 0
            if a == 1 Pknow = A(79); % Pressure at surface
            else Pknow = A(80); % Pressure at bottomhole
            end
            [P,Q] = FdMinQ(Pknow, rti, Dens_o, Visco_o, t, a); FRQ(n,1) = Q; Qo = Q;
         else
            Qo = A(13); [P,Q] = FdMinQ(Qo, rti, Dens_o, Visco_o, t, a);
         end
         Pressure(:,n) = P;
         if P(1) <= 0
            disp('The bottomhole pressure is not sufficient!'); break;
         end
      else
         Qo = A(13);
      end
      GLR2 = A(17); %Gas liquid ratio at the surface, SCF/STB
      GLR1 = A(16); %Gas liquid ratio at the bottom, SCF/STB
      Qg1 = (GLR2 - GLR1) * Qo / 24; Qg2 = GLR1 * Qo / 24;
```

```matlab
Flowrate_o = Qo * 5.615 / 24; Massrate_o = Flowrate_o * Dens_o; %oil mass flow rate
if WoNGLV == 1
    for i = 1: Num_z
        Q_inj = GLVRatio * Qg1; FRg(1) = Qg1;
        for ii = 2:Num_GLV
            FRg(ii) = FRg(ii-1) - Q_inj(ii-1); % Gas flowrate in the annulus
        end
        if i <= GLV(Num_GLV)
            for j = 1:Num_GLV
                if i <= GLV(j)
                    Flowrate_g(i) = FRg(j); Massrate_g(i) = Flowrate_g(i) * Dens_g;
                    break;
                end
            end
        else
            Flowrate_g(i) = 0; Massrate_g(i) = 0;
        end
        Flowrate_mix(i) = Flowrate_o + Flowrate_g(i) + Qg2;
        Massrate_mix(i) = - Massrate_o - Massrate_g(i) - Qg2 * Dens_g;
        Xo(i) = - Massrate_o / Massrate_mix(i);
    end
    Tf = Te(Num_z); % Bottomhole oil temperature
    Tg = A(19); % Injected gas temperature
    Xg = 1 - Xo; C_mix = Xo * Co + Xg * Cg; K_mix = Xo * Ko + Xg * Kg;
    Dens_mix = Xo * Dens_o + Xg * Dens_g;
    Visco_mix = Xo * Visco_o + Xg * Visco_g;
    [P,Q] = FdMinQ(Qo, rti, Dens_mix, Visco_mix, t, a); Pressure(:,n) = P;
    if P(1) <= 0
        disp('The bottomhole pressure is not sufficient!'); break;
    end
else
    Flowrate_g(1:Num_z) = 0; Massrate_g(1:Num_z) = zeros;
    Flowrate_mix(1:Num_z) = Flowrate_o + Qg2;
    Massrate_mix(1:Num_z) = -(Massrate_o + Qg2 * Dens_g);
    Xg(1:Num_z)    = - Qg2 * Dens_g ./ Massrate_mix(1:Num_z) ;
    Tf = A(18); % Production temperature at the bottomhole
    Xo = 1 - Xg; C_mix = Xo * Co + Xg * Cg;
    K_mix = Xo * Ko + Xg * Kg; Dens_mix = Xo * Dens_o + Xg * Dens_g;
    Visco_mix = Xo * Visco_o + Xg * Visco_g;
end
end
if WoNWax == 0
    if A(75) ~= 0
        rto(1:M(WHERE)) = r_outAdd; rto((M(WHERE)+1):Num_z) = r_outTubing;
```

```
          Uwi(1:M(WHERE)) = (rti(1:M(WHERE)) .* (log(r_outTubing/r_inTubing)/Ktub +
log(r_outAdd/r_inAdd)/Kins)).^(-1
          Uwi((M(WHERE)+1):Num_z) = (rti((M(WHERE)+1):Num_z) .*
(log(r_outTubing/r_inTubing)/Ktub)).^(-1);
      else
          rto(1:Num_z) = r_outTubing;
          Uwi(1:Num_z) = (rti .* log(r_outTubing/r_inTubing)/Ktub).^(-1);
      end
   else
      if A(75) ~= 0
          rto(1:M(WHERE)) = r_outAdd;
          rto((M(WHERE)+1):Num_z) = r_outTubing;
          Uwi(1:M(WHERE)) = (rti(1:M(WHERE)) .* (log(r_inTubing./rti(1:M(WHERE)))/Kwax
+ log(r_outTubing/r_inTubing)/Ktub + log(r_outAdd/r_inAdd)/Kins)).^(-1);
          Uwi((M(WHERE)+1):Num_z)=(rti((M(WHERE)+1):Num_z).*(log(r_inTubing./rti((M(
WHERE)+1):Num_z))/Kwax + log(r_outTubing/r_inTubing)/Ktub)).^(-1);
      else
          rto(1:Num_z) = r_outTubing;
Uwi(1:Num_z)=(rti.*(log(r_inTubing./rti)/Kwax+log(r_outTubing/r_inTubing)/Ktub)).^(-1);
      end
   end
   if Welltype == 0
      if A(3) == 0
          [hti,hto,hci] = HTCF(Flowrate_mix,Flowrate_g,Dens_g,Visco_g,Kg,Cg,rti);
      else
          [hti,hto,hci] = HTCF(Flowrate_mix,Flowrate_g,A(3),(A(4)*2.4191),A(2),A(1),rti);
      end
   else
      [hti,hto,hci] = HTCF(Flowrate_mix,Flowrate_g,Dens_mix,Visco_mix,K_mix,C_mix,rti);
   end
   if n == 1
      [Toil,Tgas,Temp,T] = TempCal(MBLocation(:,1),Formation_temp,Welltype,rti,hti);
   else
      if A(92) == 1
          S1 = rh; S1(1:MPerm) = MB;
          while (norm(Check) ~= sqrt(MPerm))
             for i = 1:MPerm
                if Temp(i,1) <= 32
                   S1(i) = rh(i); Check(i) = 1; MDS(i) = 1;
                else
                   if MDS(i) == 1
                      S1(i) = MB(i); Check(i) = 0; MDS(i) = 0;
                   end
                end
```

```matlab
                    end
                    [Toil,Tgas,Temp,T] = TempCal(S1',Temp,Welltype,rti,hti);
                    for i = 1:MPerm
                        T1 = Temp(i,:);
                        if Check(i) == 0
                            V(i) = StefanCheck(S1(i),T1,i);
                            S = Timepace * (V(i) + V_old(i))/2 + MBLocation(i,n-1);
                            Diff = abs(S - S1(i));
                            if Diff >= 0.0001
                                Check(i) = 0;
                            else
                                Check(i) = 1; V_old(i) = V(i);
                            end
                            S1(i) = S;
                        end
                    end
                end
                Check(1:MPerm) = 0; MDS(1:MPerm) = 0;
                MBLocation(:,n) = S1; MB = S1(1:MPerm) + V_old * Timepace;
            else
                S1(1:Num_z)=zeros;
                [Toil,Tgas,Temp,T] = TempCal(MBLocation(:,1),Temp,Welltype,rti,hti);
            end
            if WoNWax ~= 0
                [W1, W2, W3, W4, W5, Block] = WaxCal2(Toil,T,Wax);
                if W4 == 1
                    disp('The tubing is blocked!'); break
                end
                WaxConcentration(:,n) = W1; WaxThickness(:,n) = W2; WaxFraction(:,n) = W3;
                Wax = [WaxConcentration(:,n),WaxThickness(:,n),WaxFraction(:,n)];
            end
        end
    end
    Fluid_Temp(:,n) = Toil; Gas_Temp(:,n) = Tgas;
end
Result(1:Num_z)= round(Depth(1:Num_z));Result((Num_z+1):2*Num_z)= Toil(1:Num_z);
Result((2*Num_z+1):3*Num_z)= Tgas(1:Num_z);
Result((3*Num_z+1):4*Num_z)= Pressure(:,n);
Result((4*Num_z+1):5*Num_z)= S1(1:Num_z);
Result((5*Num_z+1):6*Num_z)= WaxThickness(:,n);


=========================================================================
function [W1, W2, W3, W4] = WaxCal2(T1,T2,Wax)

global Num_z Dens_o Visco_o Ko Co Depth Delta_depth r_inTubing Cwbo Flowrate_mix
```

```
Kwax GelDens Cwbo Sa Sb Sc Dwo alpha WAT
Delt = 0.1; N = 432;
Vis_oil = Visco_o * (453.6 / 30.5 / 3600); OilDens = Dens_o * (453.6 / 30.5^3);
Koil = Ko * 0.01731; Coil = Co * 4.1868; Kw = Kwax / 57.79;
r(1:Num_z,1) = Wax(:,2); Xw(1:Num_z,1) = Wax(:,3);
L(1:Num_z) = (Depth(Num_z) - Depth(1:Num_z))*30.5;
Delta_L = Delta_depth * 30.5; Tb(1:Num_z) = (T1 - 32) * 5 / 9 + 273.15;
Ti(1:Num_z,1) = (T2(:,1) - 32) * 5 / 9 + 273.15;Tw = (T2(:,2) - 32) * 5 / 9 + 273.15;
Cwb(1:Num_z,1) = Cwbo; Cws = zeros(Num_z,1);R = r_inTubing * 30.5;
STOP = 0; BLOCK = 0;Kl(1:Num_z,1:N) = zeros; Ke(1:Num_z,1:N) = zeros;
Hi(1:Num_z,1:N) = zeros; y(1:Num_z,1) = 1-r(1:Num_z,1)/R;

for i = Num_z:-1:1
    if Ti(i,1) <= (WAT + 273.15)
        Dot = i; break
    end
end

for n = 2:N
    CountNum = 0; t(n) = t(n-1) + Delt;
    for i = Num_z:-1:1
        Count = 1;
        if Ti(i,n-1) >= (WAT + 273.15)
            Ti(i,n) = Ti(i,n-1); y(i,n) = y(i,n-1); r(i,n) = r(i,n-1); Xw(i,n) = Xw(i,n-1);
            dTdr(i,n) = 0; Cwb(i,n)= Cwbo; Count = 0; CountNum = CountNum+1;
            if CountNum == Num_z
                STOP = 1; break
            end
        end
        while Count == 1
            V(i) = (Flowrate_mix(i)*30.5^3/3600) / pi / (R*(1-y(i,n-1)))^2; % cm/s
            Re = 2 * V(i) * OilDens * (R*(1-y(i,n-1))) / Vis_oil;
            Pr = Vis_oil * Coil / Koil; Gzh = Re * Pr * (2 * (R*(1-y(i,n-1)))) / L(i);
            Le = 0.06 * Re;
            if L(i) >= Le
                Nuh = 4.36;
            else
                if Pr >= 5
                    Nuh = 3.66 + 0.0668*Gzh / (1 + 0.04*Gzh^(2/3));
                else
                    Nuh = 1.86 * Gzh^(1/3);
                end
            end
            Hi(i,n-1) = Nuh * Koil / (2 * (R*(1-y(i,n-1))));
```

```
Gzm = Re * (Vis_oil / OilDens / Dwo) * (2 * (R*(1-y(i,n-1)))) / L(i);
if Gzm < 100
    Nui = 3.66 + 1.7813 * 10^(-3) * Gzm^(5/3) / (1 + 0.04 * Gzm^(2/3))^2;
else
    Nui = 1.24 * (Gzm)^(1/3);
end
Kl(i,n-1) = Nui * Dwo / (2 * (R*(1-y(i,n-1))));
Cws(i) = Sa * (Ti(i,n-1) -273.15 + Sb)^Sc;
dCwdT(i) = Sa * Sc * (Ti(i,n-1) -273.15 + Sb)^(Sc - 1);
dTdr(i,n-1) =(Ti(i,n-1)-Tw(i))/(R*(1-y(i,n-1))*log(1-y(i,n-1)));
AA1=(Kl(i,n-1)*(Cwb(i+1,n)-Cws(i)))/(R*Xw(i,n-1)*GelDens);
AB1=(Dwo/(1+alpha^2*Xw(i,n-1)^2/(1-Xw(i,n-1)))*dCwdT(i)*dTdr(i,n-1))/(R*Xw(i,n-1)*GelDens);
A1 = (AA1+AB1);
B1 =
-2*Dwo*dTdr(i,n-1)*(1-y(i,n-1))*dCwdT(i)/(R*GelDens*y(i,n-1)*(2-y(i,n-1))*(1+alpha^2*Xw(i,n-1)^2/(1-Xw(i,n-1))));
AA2=(Kl(i,n-1)*(Cwb(i+1,n)-Cws(i)))/(R*(Xw(i,n-1)+0.5*B1)*GelDens);
AB2=(Dwo/(1+alpha^2*(Xw(i,n-1)+0.5*B1)^2/(1-Xw(i,n-1)-0.5*B1))*dCwdT(i)*dTdr(i,n-1))/(R*(Xw(i,n-1)+0.5*B1)*GelDens);
A2 = (AA2+AB2);
B2 =
-2*Dwo*dTdr(i,n-1)*(1-y(i,n-1)-0.5*A1)*dCwdT(i)/(R*GelDens*(y(i,n-1)+0.5*A1)*(2-y(i,n-1)-0.5*A1)*(1+alpha^2*(Xw(i,n-1)+0.5*B1)^2/(1-Xw(i,n-1)-0.5*B1)));
AA3=(Kl(i,n-1)*(Cwb(i+1,n)-Cws(i)))/(R*(Xw(i,n-1)+0.5*B2)*GelDens);
AB3=(Dwo/(1+alpha^2*(Xw(i,n-1)+0.5*B2)^2/(1-Xw(i,n-1)-0.5*B2))*dCwdT(i)*dTdr(i,n-1))/(R*(Xw(i,n-1)+0.5*B2)*GelDens);
A3 = (AA3+AB3);
B3 =
-2*Dwo*dTdr(i,n-1)*(1-y(i,n-1)-0.5*A2)*dCwdT(i)/(R*GelDens*(y(i,n-1)+0.5*A2)*(2-y(i,n-1)-0.5*A2)*(1+alpha^2*(Xw(i,n-1)+0.5*B2)^2/(1-Xw(i,n-1)-0.5*B2)));
AA4=(Kl(i,n-1)*(Cwb(i+1,n)-Cws(i)))/(R*(Xw(i,n-1)+B3)*GelDens);
AB4=(Dwo/(1+alpha^2*(Xw(i,n-1)+B3)^2/(1-Xw(i,n-1)-B3))*dCwdT(i)*dTdr(i,n-1))/(R*(Xw(i,n-1)+B3)*GelDens);
A4 = (AA4+AB4);
B4 =
-2*Dwo*dTdr(i,n-1)*(1-y(i,n-1)-A3)*dCwdT(i)/(R*GelDens*(y(i,n-1)+A3)*(2-y(i,n-1)-A3)*(1+alpha^2*(Xw(i,n-1)+B3)^2/(1-Xw(i,n-1)-B3)));
y(i,n)= y(i,n-1) + Delt*(A1 +2*A2 +2*A3 +A4 )/6;
Xw(i,n)= Xw(i,n-1) + Delt*(B1 +2*B2 +2*B3 +B4 )/6;
Ke(i,n) = (2 * Kw + Koil + (Kw - Koil) * Xw(i,n)) * Koil / (2 * Kw + Koil - 2 * (Kw - Koil) * Xw(i,n));
Item1 = -Ke(i,n)/(R*(1-y(i,n))*log(1-y(i,n)));
Ti_new(i)=(Hi(i,n-1)*Tb(i)+Item1*Tw(i))/(Hi(i,n-1)+Item1);
```

```
         Diff = Ti_new(i) - Ti(i,n-1);
         if y(i,n) >= 1
            Count = 0; BLOCK = 1;
         end
         if abs(Diff) < 1e-5
            Count = 0; r(i,n) = R*(1-y(i,n)); Ti(i,n) = Ti_new(i); dXw(i,n) = Xw(i,n) - Xw(i,n-1);
            Item1 = pi*(R^2-r(i,n)^2)*dXw(i,n)*GelDens*Delta_L(i);
            Cwb(i,n) = Cwb(i+1,n) - Item1/(Flowrate_mix(i)*30.5^3/3600)/Delt;
         else
            Ti(i,n-1) = Ti_new(i);
         end
      end
      if BLOCK == 1
         STOP = 1; break
      end
   end
   if STOP == 1    break    end
   if BLOCK == 1    break    end
end
if STOP == 1
   W1 = Cwb(1:Num_z,1); W2 = Wax(:,2); W3 = Wax(:,3); W4 = BLOCK;
else
   W1 = Cwb(:,n); W2 = r(:,n); W3 = Xw(:,n); W4 = BLOCK;
end


=====================================================================
function [Tf_new,Tg_new,T,TT]=TempCal(S,temp,Welltype,rti,hti)

global Num Delta_depth Tf Tg Num_z Num_r Te Length Tw Q_inj Dens_g Cg Massrate_mix

C_mix GLV Num_GLV WoNGLV r_inTubing Kwax WoNWax Massrate_o Co Qg2
Clm_1(1:Num) = zeros; Clm_2(1:Num) = zeros; Clm_3(1:Num) = zeros;
Clm_4(1:Num) = zeros; Clm_5(1:Num) = zeros; Clm_6(1:Num) = zeros;
Clm_7(1:Num) = zeros; Clm_8(1:Num) = zeros; Clm_9(1:Num) = zeros;
Q(1:Num) = zeros;
[A,B,C,D,E,F,V1,V2,V3,V4,V5,V6,V7] = CoeffCal(S,WoNGLV,rti);
[H1,H2,H3,H4] = CoeffMatrix(S,temp);
switch WoNGLV
   case 0
      if Welltype    == 1
         Q(Num_z-1) = (1/Delta_depth(Num_z-1) - A(Num_z)) * Tf;
         Q(2*Num_z - 1) = - V1(Num_z) * Tf;
         for i = 1:Num_z
            if (S(i) > Length(i,1)) && (S(i) <= Length(i,2))
```

```matlab
        if (i == Num_z)
            Q(i + (Num_z-1)) = -V1(i) * Tf - V3(i) * Tw;
        else
            Q(i + (Num_z-1)) = - V3(i) * Tw;
        end
        V3(i) = 0;
    end
end
Q((2*Num_z):Num) = H4;
Clm_1(Num_z:(Num - Num_z)) = H1; Clm_2(1:(Num_z-1)) = V1(1:(Num_z-1));
Clm_3(1:Num_z-1) = 1 ./ Delta_depth(1:Num_z-1);
Clm_3(Num_z:2*Num_z-1) = V2(1:Num_z); Clm_3(2*Num_z:Num) = H2;
for i = 2:(Num_z-1)
    Clm_4(i) = A(i) - 1 / Delta_depth(i-1);
end
Clm_5((Num_z+1):2*Num_z-1) = B(2:Num_z);
Clm_5(2*Num_z:(3*Num_z-1)) = V3;
Clm_5(3*Num_z:Num) = H3(1:Num-(3*Num_z-1));
Tsolution(1:(Num_z-1)) = Tf;
for j = 1:Num_r-1
    for i = 1:Num_z
        index = j * Num_z - 1 + i; Tsolution(index) = Te(i);
    end
end
Bmax = [Clm_1',Clm_2',Clm_3',Clm_4',Clm_5'];
d = [-Num_z, -(Num_z-1), 0, 1, Num_z];
Coeff = spdiags(Bmax, d, Num, Num); Tsolution = Coeff \ Q';
Tf_new(1:Num_z - 1) = Tsolution(1:Num_z - 1);
Tf_new(Num_z) = Tf;
for j = 1:Num_r-1
    for i = 1:Num_z
        Index = (j-1)*Num_z + i;
        T(i,j) = Tsolution((Num_z-1)+Index);
        Tg_new(i) = V5(i) * Tf_new(i) + V6(i) * T(i,1);
    end
end
T(:,Num_r) = Te; Twi = V7(4,:) .* Tf_new + V7(1,:) .* V7(3,:) .* V7(4,:) .* T(:,1)';
if WoNWax == 0
    Tti = Twi;
else
    Tti = Twi - rti .* hti .* (Tf_new - Twi) .* log(r_inTubing ./ rti) / Kwax;
end
TT = [Twi',Tti'];
else
```

```matlab
Q(1) = - Tf / Delta_depth(1); Q(Num_z) = - V1(1) * Tf;
for i = 1:Num_z
   if (S(i) > Length(i,1)) && (S(i) <= Length(i,2))
      if (i == 1)
         Q(i + (Num_z-1)) = -V1(i) * Tf - V3(i) * Tw;
      else
         Q(i + (Num_z-1)) = - V3(i) * Tw;
      end
      V3(i) = 0;
   end
end
Q((2*Num_z):Num) = H4;
Clm_1(1:(Num_z - 1)) = V1(2:Num_z); Clm_1(Num_z :(Num - Num_z)) = H1;
Clm_2(1:(Num_z-2)) = 1 / Delta_depth(2:(Num_z-1));
for i = 1:Num
   if i < Num_z
      Clm_3(i) = A(i+1) - 1 / Delta_depth(i);
   elseif (i >= Num_z) && (i <= (2*Num_z-1))
      Clm_3(i) = V2(i-(Num_z-1));
   else
      Clm_3(i) = H2(i-(2*Num_z-1));
   end
end
for i = (Num_z+1):Num
   if (i < 2*Num_z)
      Clm_4(i) = B(i-Num_z+1);
   elseif (i >= 2*Num_z) && (i <= (3*Num_z-1))
      Clm_4(i) = V3(i-(2*Num_z-1));
   else
      Clm_4(i) = H3(i-(3*Num_z-1));
   end
end
Tsolution(1:(Num_z-1)) = Tf;
for j = 1:Num_r-1
   for i = 1:Num_z
      index = j * Num_z - 1 + i; Tsolution(index) = Te(i);
   end
end
Bmax = [Clm_1',Clm_2',Clm_3',Clm_4']; d = [-Num_z, -1, 0, Num_z];
Coeff = spdiags(Bmax, d, Num, Num); Tsolution = Coeff \ Q';
for i = 1:Num_z
   if i == 1
      Tf_new(1) = Tf;
   else
```

```matlab
                Tf_new(i) = Tsolution(i-1);
            end
        end
        for j = 1:Num_r-1
            for i = 1:Num_z
                Index = (j-1)*Num_z + i;
                T(i,j) = Tsolution((Num_z-1)+Index); Tg_new(i) =V5(i)*Tf_new(i)+V6(i) * T(i,1);
            end
        end
        T(:,Num_r) = Te; Twi = V7(4,:) .* Tf_new + V7(1,:) .* V7(3,:) .* V7(4,:) .* T(:,1)';
        if WoNWax == 0
            Tti = Twi;
        else
            Tti = Twi - rti .* hti .* (Tf_new - Twi) .* log(r_inTubing ./ rti) / Kwax;
        end
        TT = [Twi',Tti'];
    end
    case 1
        for i = 1:2*(Num_z-1)
            if (i < (Num_z-1))
                Q(i) = 0;
            elseif (i == (Num_z-1))
                if GLV(Num_GLV) == Num_z
                    Q(i) = (1 / Delta_depth(i) - A(i+1)) * Tf * (-Massrate_o *Co - Qg2 * Dens_g * Cg) /
(Massrate_mix(Num_z) * C_mix(Num_z));
                else
                    Q(i) = (1 / Delta_depth(i) - A(i+1)) * Tf;
                end
            elseif (i == Num_z)
                Q(i) = - Tg / Delta_depth(1);
            elseif (i > Num_z) && (i < 2*(Num_z-1))
                Q(i) = 0;
            else
                if GLV(Num_GLV) == Num_z
                    Q(i) = -D(Num_z) * Tf * (-Massrate_o *Co - Qg2 * Dens_g * Cg) /
(Massrate_mix(Num_z) * C_mix(Num_z));
                else
                    Q(i) = 0;
                end
            end
        end
        for i = 1:Num_z
            if (S(i) > Length(i,1)) && (S(i) <= Length(i,2))
                if (i == 1)
```

```matlab
            Q(i + 2*(Num_z-1)) = -V2(i) * Tg - V4(i) * Tw;
        elseif (i == Num_z)
            if GLV(Num_GLV) == Num_z
                Q(i + 2*(Num_z-1)) = -V1(i) * Tf * (-Massrate_o *Co - Qg2 * Dens_g * Cg) /
(Massrate_mix(Num_z) * C_mix(Num_z)) - V4(i) * Tw;
            else
                Q(i + 2*(Num_z-1)) = -V1(i) * Tf - V4(i) * Tw;
            end
        else
            Q(i + 2*(Num_z-1)) = - V4(i) * Tw;
        end
        V4(i) = 0;
    else
        if (i == 1)
            Q(i + 2*(Num_z-1)) = -V2(i) * Tg;
        elseif (i == Num_z)
            if GLV(Num_GLV) == Num_z
                Q(i + 2*(Num_z-1)) = -V1(i) * Tf * (-Massrate_o *Co - Qg2 * Dens_g * Cg) /
(Massrate_mix(Num_z) * C_mix(Num_z));
            else
                Q(i + 2*(Num_z-1)) = -V1(i) * Tf;
            end
        else
            Q(i + 2*(Num_z-1)) = 0;
        end
    end
end
Q((3*Num_z-1):Num) = H4(1: Num -(3*Num_z-2));
for i = 1:(Num_z-1)
    Clm_1(i) = V1(i);
    for j = 1: Num_GLV
        if i == GLV(j)
            Clm_1(i)=V1(i)*Massrate_mix(i+1)*C_mix(i+1)/(Massrate_mix(i) * C_mix(i));
            break;
        end
    end
end
for i = 2:(Num_z-1)
    Clm_3(i) = D(i); Clm_6(i) = (A(i) - 1 / Delta_depth(i-1));
    for j = 1: Num_GLV
        if i == GLV(j)
            Clm_3(i) = D(i)*Massrate_mix(i+1) * C_mix(i+1) / (Massrate_mix(i) * C_mix(i));
            Clm_6(i) = (A(i) - 1 / Delta_depth(i-1)) * Massrate_mix(i+1) * C_mix(i+1) /
(Massrate_mix(i) * C_mix(i)); break;
```

```
            end
          end
        end
      for i = Num_z:(Num - Num_z)
        if i <= 2*(Num_z-1)
          Clm_2(i) = V2(i-(Num_z-1)+1);
          for j = 1: Num_GLV
            if i == GLV(j) + (Num_z - 2)
              Clm_2(i) = V2(GLV(j)) + V1(GLV(j)) * (-Q_inj(j)) * Dens_g * Cg /
(Massrate_mix(GLV(j)) * C_mix(GLV(j)));
            end
          end
        else
          Clm_2(i) = H1(i-2*(Num_z-1));
        end
      end
      for i = Num_z:(Num_z-1 + GLV(Num_GLV)-2)
        Clm_4(i) = 1 / Delta_depth(i-(Num_z-1)+1);
      end
      for i = 1:Num
        if i <= (Num_z-1)
          Clm_5(i) = 1 / Delta_depth(i);
        elseif (i > (Num_z-1)) && (i <= 2*(Num_z-1))
          Clm_5(i) = E(i-(Num_z-1)+1) - 1 / Delta_depth(i-(Num_z-1));
          for j = 1: Num_GLV
            if i == GLV(j) + (Num_z - 2)
              Clm_5(i) = (E(i-(Num_z-1)+1) - 1 / Delta_depth(i-(Num_z-1))) +
D(i-(Num_z-1)+1) * (-Q_inj(j)) * Dens_g * Cg / (Massrate_mix(GLV(j)) * C_mix(GLV(j)));
            end
          end
        elseif (i > 2*(Num_z-1)) && (i <= 3*Num_z-2)
          Clm_5(i) = V3(i-2*(Num_z-1));
        else
          Clm_5(i) = H2(i-(3*Num_z-2));
        end
      end
      for i = Num_z:(2*(Num_z-1))
        Clm_7(i) = B(i-(Num_z-1)+1);
        for j = 1: Num_GLV
          if i == GLV(j) + (Num_z - 2)
            Clm_7(i) = B(i-(Num_z-1)+1) + (A(i-(Num_z-1)+1) - 1 / Delta_depth(i-(Num_z-1)))
* (-Q_inj(j)) * Dens_g * Cg / (Massrate_mix(GLV(j)) * C_mix(GLV(j)));
          end
        end
```

```matlab
        end
    Clm_8((2*Num_z) : (3*Num_z-2)) = F(2: Num_z);
    Clm_8((3*Num_z-1): (4*Num_z-2)) = V4(1: Num_z);
    Clm_8(4*Num_z-1: Num) = H3(1:Num-(4*Num_z-2));
    Clm_9((2*Num_z):(3*Num_z-2)) = C(2: Num_z);
    Tsolution(1:(Num_z-1)) = Tf; Tsolution(Num_z:2*(Num_z-1)) = Tg;
    for j = 1:Num_r-1
        for i = 1:Num_z
            index = (j+1)*Num_z - 2 + i; Tsolution(index) = Te(i);
        end
    end
    Bmax = [Clm_1',Clm_2',Clm_3',Clm_4',Clm_5',Clm_6',Clm_7',Clm_8',Clm_9'];
    d = [-2*(Num_z-1), -Num_z, -(Num_z-2), -1, 0, 1, (Num_z-1), Num_z, (2*Num_z-1)];
    Coeff = spdiags(Bmax, d, Num, Num); Tsolution = Coeff \ Q';
    Tg_new(1) = Tg; Tg_new(2:Num_z) = Tsolution(Num_z: 2* Num_z - 2);
    Tf_new(1:Num_z-1) = Tsolution(1:Num_z-1);
    if GLV(Num_GLV) == Num_z
        Tf_new(i) = (Tg_new(Num_z) * Q_inj(Num_GLV) * Dens_g * Cg - Tf * (-Massrate_o
*Co - Qg2 * Dens_g * Cg))/(-Massrate_mix(Num_z) * C_mix(Num_z));
    else
        Tf_new(i) = Tf;
    end

    for j = 1:Num_r-1
        for i = 1:Num_z
            Index = (j-1)*Num_z + i; T(i,j) = Tsolution(2*(Num_z-1)+Index);
        end
    end
    T(:,Num_r) = Te;
    for i = 1:Num_z
        if GLV(Num_GLV) == Num_z
            Tto(i) = Tf_new(i) / V7(5,i) + Tg_new(i) * V7(6,i) + T(i,1)' * V7(7,i);
            Twi(i) = (Tf_new(i) + V7(3,i) * Tto(i)) / (1 + V7(3,i));
        else
            if i <= GLV(Num_GLV)
                Tto(i) = Tf_new(i) / V7(5,i) + Tg_new(i) * V7(6,i) + T(i,1)' * V7(7,i);
                Twi(i) = (Tf_new(i) + V7(3,i) * Tto(i)) / (1 + V7(3,i));
            else
                Twi(i) = V7(4,i) * Tf_new(i) + V7(1,i) * V7(3,i) * V7(4,i) * T(i,1)';
                Tto(i) = V7(2,i) * Twi(i) + V7(3,i) * T(i,1)';
            end
        end
    end
    if WoNWax == 0
```

```matlab
            Tti = Twi;
        else
            Tti = Twi - rti .* hti .* (Tf_new - Twi) .* log(r_inTubing ./ rti) / Kwax;
        end
        TT = [Twi',Tti'];
end


%=========================================================================
function [A,B,C,D] = CoeffMatrix(S, temperature)

global Length Lenpace Tw Timepace Ks Kl Cs Cl Form_Dens Num_z Num_r Te

L = Length(:,2:Num_r); Pace = Lenpace(:,2:Num_r);
for i = 1:Num_z
    for j = 1:Num_r-1
        if S(i) <= L(i,j)
            Dot = j; break;
        end
    end
    for n = 1:(Num_r-2)
        index = (n - 1) * Num_z + i;
        variable1 = 2 / (Pace(i,n) + Pace(i,n+1)); variable2 = variable1 / (2 * L(i,n));
        if Dot == 1
            constant = Form_Dens * Cs / (Ks * Timepace);
            if n == 1
                if S(i) == Length(i,1)
                    A(index) = variable1 / Pace(i,n) - variable2;
                    B(index) = -variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant;
                    C(index) = variable1 / Pace(i,n+1) + variable2;
                    D(index) = -temperature(i,n+1) * constant;
                else
                    Delta_r = L(i,n) - S(i); I1 = 2 / (Delta_r + Pace(i,n+1)); I2 = I1 / (2 * L(i,n));
                    A(index) = 0;
                    B(index) = -I1 * (1 / Delta_r + 1 / Pace(i,n+1)) - constant;
                    C(index) = I1 / Pace(i,n+1) + I2;
                    D(index) = -temperature(i,n+1) * constant - (I1 / Delta_r - I2) * Tw;
                end
            elseif n == (Num_r-2)
                A(index) = variable1 / Pace(i,n) - variable2;
                B(index) = -variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant;
                C(index) = 0;
                D(index) = -temperature(i,n+1)*constant-(variable1/Pace(i,n+1)+variable2)*Te(i);
            else
                A(index) = variable1 / Pace(i,n) - variable2;
```

```
        B(index) = -variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant;
        C(index) = variable1 / Pace(i,n+1) + variable2;
        D(index) = -temperature(i,n+1) * constant;
    end
elseif Dot == (Num_r-2)
    constant1 = Form_Dens * Cl / (Kl * Timepace);
    constant2 = Form_Dens * Cs / (Ks * Timepace);
    if n < Dot
        if n == (Dot-1)
            Delta_r = S(i) - L(i,n); I1 = 2 / (Pace(i,n) + Delta_r); I2 = I1 / (2 * L(i,n));
            A(index) = I1 / Pace(i,n) - I2;
            B(index) = - I1 * (1 / Pace(i,n) + 1 / Delta_r) - constant1;
            C(index) = 0;
            D(index) = -temperature(i,n+1) * constant1 - (I1 / Delta_r + I2) * Tw;
        else
            A(index) = variable1 / Pace(i,n) - variable2;
            B(index) = - variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant1;
            C(index) = variable1 / Pace(i,n+1) + variable2;
            D(index) = -temperature(i,n+1) * constant1;
        end
    else
        Delta_r = L(i,n) - S(i); I1 = 2 / (Delta_r + Pace(i,n+1)); I2 = I1 / (2 * L(i,n));
        A(index) = 0;
        B(index) = - I1 * (1 / Delta_r + 1 / Pace(i,n+1)) - constant2;
        C(index) = 0;
       D(index)=-temperature(i,n+1)*constant2-(I1/Delta_r-I2)*Tw-(I1/Pace(i,n+1)+I2)*Te(i);
    end
else
    constant1 = Form_Dens * Cl / (Kl * Timepace);
    constant2 = Form_Dens * Cs / (Ks * Timepace);
    if n < Dot
        if n == (Dot-1)
            Delta_r = S(i) - L(i,n); I1 = 2 / (Pace(i,n) + Delta_r); I2 = I1 / (2 * L(i,n));
            A(index) = I1 / Pace(i,n) - I2;
            B(index) = - I1 * (1 / Pace(i,n) + 1 / Delta_r) - constant1;
            C(index) = 0;
            D(index) = -temperature(i,n+1) * constant1 - (I1 / Delta_r + I2) * Tw;
        else
            A(index) = variable1 / Pace(i,n) - variable2;
            B(index) = - variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant1;
            C(index) = variable1 / Pace(i,n+1) + variable2;
            D(index) = -temperature(i,n+1) * constant1;
        end
    else
```

```matlab
            if n == Dot
                Delta_r = L(i,n) - S(i); I1 = 2 / (Delta_r + Pace(i,n+1)); I2 = I1 / (2 * L(i,n));
                A(index) = 0;
                B(index) = - I1 * (1 / Delta_r + 1 / Pace(i,n+1)) - constant2;
                C(index) = I1 / Pace(i,n+1) + I2;
                D(index) = - temperature(i,n+1) * constant2 - (I1 / Delta_r - I2) * Tw;
            elseif n == (Num_r-2)
                A(index) = variable1 / Pace(i,n) - variable2;
                B(index) = - variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant2;
                C(index) = 0;
                D(index) = -temperature(i,n+1)*constant2-(variable1/Pace(i,n+1)+variable2)*Te(i);
            else
                A(index) = variable1 / Pace(i,n) - variable2;
                B(index) = - variable1 * (1 / Pace(i,n) + 1 / Pace(i,n+1)) - constant2;
                C(index) = variable1 / Pace(i,n+1) + variable2;
                D(index) = - temperature(i,n+1) * constant2;
            end
        end
    end
  end
end


=========================================================================
function [A,B,C,D,E,F,V1,V2,V3,V4,V5,V6,V7] = CoeffCal(S,WoNGLV,rti)

global Cg C_mix Massrate_g Massrate_mix Num_z Length Lenpace Ks Kl GLV Num_GLV hci
hr hti hto Kcas r_inProdcasing r_outProdcasing rh rto Uwi Uco

for i = 1:Num_z
  if S(i) == Length(i,1)
    Alfa(i) = Ks * rh(i) /(r_outProdcasing * Uco(i) * Lenpace(i,2));
  else
    if S(i) <= Length(i,2)
      Delta_r = S(i) - Length(i,1);
    else
      Delta_r = Lenpace(i,2);
    end
    Alfa(i) = Kl * rh(i) /(r_outProdcasing * Uco(i) * Delta_r);
  end
end

switch WoNGLV
  case 0
    Alfa_1 = r_outProdcasing * Uco * log(r_outProdcasing / r_inProdcasing) / Kcas;
```

```
Alfa_2 = Kcas ./ (rto .* (hr + hci) * log(r_outProdcasing / r_inProdcasing));
Alfa_3 = rto .* (hr + hci) ./ (rti .* Uwi); Alfa_4 = Uwi ./ hti;
Alfa_5 = (1 + Alfa_1) ./ (1 + Alfa_1 + Alfa_1 .* Alfa_2);
Alfa_6 = Alfa_1 .* Alfa_2 ./ (1 + Alfa_1 + Alfa_1 .* Alfa_2);
Alfa_7 = 1 ./ (1 + Alfa_3 - Alfa_3 .* Alfa_5);
Alfa_8 = Alfa_3 .* Alfa_6 ./ (1 + Alfa_3 - Alfa_3 .* Alfa_5);
Alfa_9 = 1 ./ (1 + Alfa_4 - Alfa_4 .* Alfa_7);
Variable_1 = -2 * pi * rti .* hti ./ (Massrate_mix .* C_mix);
A = Variable_1 .* (1 - Alfa_9); B = - Variable_1 .* Alfa_4 .* Alfa_8 .* Alfa_9;
V1 = Alfa_5 .* Alfa_7 .* Alfa_9;
V2 = Alfa_5.*Alfa_8.*(1+Alfa_4.*Alfa_7.*Alfa_9)+Alfa_6-1-(1+Alfa_1).*Alfa;
V3 = (1 + Alfa_1) .* Alfa;
C(1:Num_z) = zeros; D(1:Num_z) = zeros;
E(1:Num_z) = zeros; F(1:Num_z) = zeros;
V4(1:Num_z) = zeros; V5 = (Alfa_5 + 1) .* Alfa_7 .* Alfa_9 / 2;
V6 = ((Alfa_5 + 1) .* Alfa_8 .* (Alfa_4 .* Alfa_7 .* Alfa_9 + 1) + Alfa_6) / 2;
V7 = [Alfa_4; Alfa_7; Alfa_8; Alfa_9; Alfa_1; Alfa_5; Alfa_6];
case 1
  for i = 1:Num_z
    if i <= GLV(Num_GLV)
      Alfa_2 = r_outProdcasing * Uco(i) * log(r_outProdcasing / r_inProdcasing) / Kcas;
      Alfa_3 = r_inProdcasing * hci(i) * log(r_outProdcasing / r_inProdcasing) / Kcas;
      Alfa_4 = rto(i) * hr * log(r_outProdcasing / r_inProdcasing) / Kcas;
      Variable_1 = Alfa_2 / (1 + Alfa_2);
      Alfa_5 = Alfa_3 / (Variable_1 + Alfa_3 + Alfa_4);
      Alfa_6 = Alfa_4 / (Variable_1 + Alfa_3 + Alfa_4);
      Alfa_7 = Variable_1 / (Variable_1 + Alfa_3 + Alfa_4);
      Alfa_8 = Uwi(i) / hti(i); Alfa_9 = rto(i) * hto(i) / (rti(i) * Uwi(i));
      Alfa_10 = rto(i) * hr / (rti(i) * Uwi(i));
      Alfa_11 = 1 + (1 + Alfa_8) * (Alfa_9 + Alfa_10 * (1 - Alfa_6));
      Alfa_12 = (Alfa_9 + Alfa_5 * Alfa_10) * (1 + Alfa_8) / Alfa_11;
      Alfa_13 = Alfa_7 * Alfa_10 * (1 + Alfa_8) / Alfa_11;
      Alfa_14 = (Alfa_11 - 1) / Alfa_11;
      Alfa_15 = 1 - Alfa_5 - Alfa_6 * Alfa_12;
      Alfa_16 = Alfa_6 * Alfa_13 + Alfa_7;
      V1(i) = Alfa_6 / Alfa_11; V2(i)= Alfa_5 + Alfa_6 * Alfa_12;
      V3(i) = Alfa_7 + Alfa_6 * Alfa_13 - Alfa(i) * (1 + Alfa_2) - 1;
      V4(i) = Alfa(i) * (1 + Alfa_2);
      Variable_2 = -2 * pi * rti(i) * hti(i) / (Massrate_mix(i) * C_mix(i));
      Variable_3 = 2 * pi * rto(i) * hto(i) / (Massrate_g(i) * Cg);
      Variable_4 = 2 * pi * r_inProdcasing * hci(i) / (Massrate_g(i) * Cg);
      Variable_5 = Variable_2 * Alfa_8 / (1 + Alfa_8);
      A(i) = Variable_5 * Alfa_14; B(i) = -Variable_5 * Alfa_12;
      C(i) = -Variable_5 * Alfa_13; D(i) = (Variable_3 + Alfa_6 * Variable_4) / Alfa_11;
```

```matlab
            E(i) = Variable_3 * (Alfa_12 - 1) - Variable_4 * Alfa_15;
            F(i) = Variable_3 * Alfa_13 + Variable_4 * Alfa_16;
            V5(i) = zeros; V6(i) = zeros; Alfa_1 = Alfa(i);
        else
            Alfa_1 = r_outProdcasing * Uco(i) * log(r_outProdcasing / r_inProdcasing) / Kcas;
            Alfa_2 = Kcas / (rto(i) * (hr + hci(i)) * log(r_outProdcasing / r_inProdcasing));
            Alfa_3 = rto(i) * (hr + hci(i)) / (rti(i) * Uwi(i));
            Alfa_4 = Uwi(i) / hti(i); Alfa_5 = (1 + Alfa_1) / (1 + Alfa_1 + Alfa_1 * Alfa_2);
            Alfa_6 = Alfa_1 * Alfa_2 / (1 + Alfa_1 + Alfa_1 * Alfa_2);
            Alfa_7 = 1 / (1 + Alfa_3 - Alfa_3 * Alfa_5);
            Alfa_8 = Alfa_3 * Alfa_6 / (1 + Alfa_3 - Alfa_3 * Alfa_5);
            Alfa_9 = 1 / (1 + Alfa_4 - Alfa_4 * Alfa_7);
            Variable_1 = -2 * pi * rti(i) * hti(i) / (Massrate_mix(i) * C_mix(i));
            A(i) = Variable_1 * (1 - Alfa_9); C(i) = - Variable_1 * Alfa_4 * Alfa_8 * Alfa_9;
            V1(i) = Alfa_5 * Alfa_7 * Alfa_9; V2(i) = zeros;
            V3(i) = Alfa_5*Alfa_8*(1+Alfa_4*Alfa_7*Alfa_9)+Alfa_6-1-(1+Alfa_1)*Alfa(i);
            V4(i) = (1 + Alfa_1) * Alfa(i);
            B(i) = zeros; D(i) = zeros; E(i) = zeros; F(i) = zeros;
            V5(i) = (Alfa_5 + 1) * Alfa_7 * Alfa_9 / 2;
            V6(i) = ((Alfa_5 + 1) * Alfa_8 * (Alfa_4 * Alfa_7 * Alfa_9 + 1) + Alfa_6) / 2;
        end
        V7(:,i) = [Alfa_4, Alfa_7, Alfa_8, Alfa_9, Alfa_11, Alfa_12, Alfa_13];
    end
end


%=======================================================================
function V = StefanCheck(X,Temp,Nlm)

global Length Ks Kl Form_Dens Latent_H Num_r
for j = 1:Num_r
   if X <= Length(Nlm,j)
      Dot = j; break;
   end
end
n1 = Dot; n2 = Num_r - n1 + 1;
V_l = Liquidregion_velocity(Temp,n1,X,Nlm); V_s = Solidregion_velocity(Temp,n2,X,Nlm);
V = (Ks * V_s - Kl * V_l) / (Latent_H * Form_Dens);


%=======================================================================
function V_l = Liquidregion_velocity(T,n1,S,I)

global Length Tw

if n1 <= 2
```

```matlab
      V_l = (T(1) - Tw) / (Length(I,1) - S);
   else
      if (n1 > 2) && (n1 <= 4)
         x = [Length(I,n1-2) Length(I,n1-1) S];
         y = [T(n1-2) T(n1-1) Tw];
      else
         x = [Length(I,n1-4) Length(I,n1-3) Length(I,n1-2) Length(I,n1-1) S];
         y = [T(n1-4) T(n1-3) T(n1-2) T(n1-1) Tw];
      end
      a = S - (S - Length(I,n1-1))/10; T_diff = interp1(x,y,a); V_l = (T_diff - Tw) / (a - S);
   end


%======================================================================
function V_s = Solidregion_velocity(T,N3,S,I)

global Length Tw Num_r

if N3 == 1
   V_s = (Tw - T(Num_r)) / (S - Length(I,Num_r));
else
   if (N3 > 1) && (N3 <= 3)
      x = [S Length(I,Num_r-N3+1) Length(I,Num_r-N3+2)];
      y = [Tw T(Num_r-N3+1) T(Num_r-N3+2)];
   else
      x=[S Length(I,Num_r-N3+1) Length(I,Num_r-N3+2) Length(I,Num_r-N3+3)
Length(I,Num_r-N3+4)];
      y = [Tw T(Num_r-N3+1) T(Num_r-N3+2) T(Num_r-N3+3) T(Num_r-N3+4)];
   end
   a = (Length(I,Num_r-N3+1) - S)/10 + S; T_diff = interp1(x,y,a); V_s = (T_diff - Tw) / (a - S);
end


%======================================================================
function [h1,h2,h3] = HTCF(Flowrate,Flowrate_g,Dens,Visco,K,C,rti)

global r_inProdcasing Dens_g Visco_g Kg Num_z rto Welltype Depth Cg

if Welltype == 0
   Re1(1: Num_z) = 2 * Flowrate * Dens / pi ./ rti / Visco; Pr1(1: Num_z) = Visco * C / K;
   for i = 2:Num_z
      if Re1(i) < 2300
         Gzh(i) = Re1(i-1) * Pr1(i-1) * (2 * rti(i-1)) / Depth(i); Le = 0.06 * Re1(i-1);
         if Depth(i) >= Le
            Nuh(i) = 4.36;
         else
```

```matlab
            if Pr1(i) >= 5
                Nuh(i) = 3.66 + 0.0668 * Gzh(i) / (1 + 0.04 * Gzh(i)^(2/3));
            else
                Nuh(i) = 1.86 * (Gzh(i))^(1/3);
            end
          end
        h1(i)= Nuh(i) * K /(2 * rti(i-1));
      else
        h1(i) = 0.023 * (K /(rti(i-1) * 2)) * (Re1(i-1)^0.8) * (Pr1(i-1)^(1/3));
      end
    end
  h1(1) = h1(2);
else
  Re1(1:Num_z) =2*Flowrate(1:Num_z).*Dens(1:Num_z)/pi./rti(1:Num_z)./Visco(1:Num_z);
  Pr1(1:Num_z) = Visco(1:Num_z) .* C(1:Num_z) ./ K(1:Num_z);
  L(1:Num_z) = (Depth(Num_z) - Depth(1:Num_z));
  for i = 1:Num_z-1
    if Re1(i) < 2300
      Gzh(i) = Re1(i) * Pr1(i) * (2 * rti(i)) / L(i); Le = 0.06 * Re1(i);
      if L(i) >= Le
        Nuh(i) = 4.36;
      else
        if Pr1 >= 5
          Nuh(i) = 3.66 + 0.0668 * Gzh(i) / (1 + 0.04 * Gzh(i)^(2/3));
        else
          Nuh(i) = 1.86 * (Gzh(i))^(1/3);
        end
      end
      h1(i)= Nuh(i) * K(i)/(2 * rti(i));
    else
      h1(i) = 0.023 * (K(i) /(rti(i) * 2)) * (Re1(i).^0.8) * (Pr1(i)^(1/3));
    end
  end
  h1(Num_z)=h1(Num_z-1);
end

Dh(2: Num_z) = 2*(r_inProdcasing - rto(2: Num_z));
a(2: Num_z) = r_inProdcasing ./ rto(2: Num_z);
for i = 2:Num_z
  if Flowrate_g(i) == 0
    h3(i) = 0.4;
  else
    Re(i) = 2 * Flowrate_g(i) * Dens_g / pi / (r_inProdcasing + rto(i)) / Visco_g;
    if Re(i) >= 10000
```

```
        if a(i) <= 5
            Thi(i) = 1;
        else
            Thi(i) = 1 + 7.5 * ((a(i) - 5) / ((a(i) + 1) * Re(i))) ^ 0.6;
        end
        h3(i) = (0.06759 * (a(i) ^ 0.16) * (Re(i) ^ 0.8) * Thi(i) / (a(i) + 1) ^ 0.2) * Kg / Dh(i);
    else
        P = 1.013 * exp(-0.067 * a(i));
        C0 = 0.003 * a(i)^1.86 / (0.063 * a(i)^3 - 0.674 * a(i)^2 + 2.225 * a(i) - 1.157);
        h3(i) = C0 * Re(i)^P * (Visco_g * Cg / Kg)^0.3333 * Kg / Dh(i);
    end
  end
end
h3(1) = h3(2); h2 = h3;


=====================================================================
function [Pressure,Q_new] = FdMinQ(Vara, rti, Dens, Vis, Time, a)

global   VDepth Num_z Delta_depth ResPVara_1 ResPVara_2

d = 2 * rti; H(1:Num_z-1) = VDepth(2:Num_z) - VDepth(1:(Num_z-1));
L(1:Num_z-1) = Delta_depth(1:Num_z-1); Visco = Vis / 2.4191;
if a == 1
  Ptf = Vara; Q = 0; Q_new = 300;
  while abs(Q - Q_new) > 0.1
    Q = Q_new;
    V(1:Num_z-1) = Q * 5.615 / 86400 / pi ./ rti(1:Num_z-1).^2;
    Re(1:Num_z-1) = 1.48 * Q * 12 * Dens ./ d(1:Num_z-1) ./ Visco;
    for i = 1:(Num_z-1)
      if Re(i) <= 2100
        fm(i) = 16 / Re(i);
      else
        Lambda = (0.001^(1.1098)) / 2.8257 + (7.149 / Re(i))^(0.8981);
        fm(i) = (4 * log10(0.001 / 3.7065 - 5.0452 / Re(i) * log10(Lambda)))^(-2);
      end
    end
    Item1(1:Num_z-1) =2*fm(1:Num_z-1)*Dens.*L(1:Num_z-1).*V(1:Num_z-1).^2/32.17./
d(1:Num_z-1) / 144;
    Item2(1:Num_z-1) = Dens * H(1:Num_z-1) / 144;
Item3(1:Num_z-1)=1.53e-8*Q^2*Dens*((rti(1:Num_z-1)*24).^(-4)-(rti(2:Num_z)*24).^(-4));
    P = sum(Item2) + Ptf + sum(Item1) + 14.7 + sum(Item3);
    Q_new = (5651 - P) / (ResPVara_1 * (log10(Time) + ResPVara_2));
  end
  Pressure(Num_z) = 5651 - ResPVara_1 * Q * (log10(Time) + ResPVara_2);
```

```matlab
    for i = (Num_z-1):-1:1
        Pressure(i) = Pressure(i+1) - Item1(i)    - Item2(i) - Item3(i);
    end
elseif a == 0
    Density(1:Num_z) = Dens; Viscosity(1:Num_z) = Visco; Q = Vara;
    Pressure(Num_z) = 5651 - ResPVara_1 * Q * (log10(Time) + ResPVara_2) ;
    for i = (Num_z-1):-1:1
        V(i) = Q * 5.615 / 86400 / pi / rti(i)^2;
        Re(i) = 1.48 * Q * Density(i) / (d(i) * 12) / Viscosity(i);
        if Re(i) <= 2100
            fm(i) = 16 / Re(i);
        else
            Lambda = (0.001^(1.1098)) / 2.8257 + (7.149 / Re(i))^(0.8981);
            fm(i) = (4 * log10(0.001 / 3.7065 - 5.0452 / Re(i) * log10(Lambda)))^(-2);
        end
        Item1(i) = 2 * fm(i) * Density(i) * L(i) * V(i)^2 / 32.17 / d(i) / 144;
        Item2(i) = Density(i) * H(i) / 144;
        Item3(i) = 1.53e-8 * Q ^2 * Density(i) * ((rti(i)*24)^(-4) - (rti(i+1)*24)^(-4));
        Pressure(i) = Pressure(i+1) - Item1(i)    - Item2(i) - Item3(i);
    end
    Q_new = Q;
else
    Pressure(Num_z) = Vara + 14.7;
    Q_new = (5651 - Pressure(Num_z)) / (ResPVara_1 * (log10(Time) + ResPVara_2));
    V(1:Num_z) = Q_new * 5.615 / 86400 / pi ./ rti(1:Num_z).^2;
    Re(1:Num_z) = 1.48 * Q_new * Dens ./ (d(1:Num_z)*12) ./ Visco;
    for i = (Num_z-1):-1:1
        if Re(i) <= 2100
            fm(i) = 16 / Re(i);
        else
            Lambda = (0.001^(1.1098)) / 2.8257 + (7.149 / Re(i))^(0.8981);
            fm(i) = (4 * log10(0.001 / 3.7065 - 5.0452 / Re(i) * log10(Lambda)))^(-2);
        end
        Item1(i) = 2 * fm(i) * Dens * L(i) * V(i)^2 / 32.17 / d(i) / 144;
        Item2(i) = Dens * H(i) / 144;
        Item3(i) = 1.53e-8 * Q_new^2 * Dens* ((rti(i)*24)^(-4) - (rti(i+1)*24)^(-4));
        Pressure(i) = Pressure(i+1) - Item1(i)    - Item2(i) - Item3(i);
    end
end
```