A general finite volume based numerical algorithm for hydrocarbon reservoir simulation using black oil model

Article *in* International Journal of Numerical Methods for Heat and Fluid Flow · October 2014

DOI: 10.1108/HFF-10-2013-0302

CITATIONS
5

READS
152

4 authors:

Mehdi Mosharaf-Dehkordi
University of Isfahan
19 PUBLICATIONS **44** CITATIONS
SEE PROFILE

Mehrdad Taghizadeh Manzari
Sharif University of Technology
62 PUBLICATIONS **738** CITATIONS
SEE PROFILE

Hamid Reza Ghafouri
Shahid Chamran University of Ahvaz
16 PUBLICATIONS **127** CITATIONS
SEE PROFILE

Rouhollah Fatehi
Persian Gulf University
36 PUBLICATIONS **277** CITATIONS
SEE PROFILE

A General Finite Volume Based Numerical Algorithm For Hydrocarbon Reservoir Simulation Using Blackoil Model

M. Mosharaf Dehkordi[+], M. T. Manzari[+], H. R. Ghafouri[††], R. Fatehi[*]

[+] Sharif University of Technology, Tehran, Iran Tehran, Iran
e-mail: mosharaf@mech.sharif.edu
e-mail: mtmanzari@sharif.edu
[††] Shahid Chamran University, Ahwaz, Iran
e-mail: ghafouri_h@scu.ac.ir
[*] Persian Gulf University, Bushehr, Iran.
e-mail: fatehy@gmail.com

Abstract

Purpose- This paper presents a detailed algorithm for simulating three-dimensional hydrocarbon reservoirs using the blackoil model.

Design/methodology/approach- The numerical algorithm uses a cell-centred structured grid finite volume method. The blackoil formulation is written in such a way that an Implicit Pressure Explicit Saturation (IMPES) approach can be used. The flow field is obtained by solving a general gas pressure equation derived by manipulating the governing equations. All possible variations of the pressure equation coefficients are given for different reservoir conditions. Key computational details including treatment of nonlinear terms, expansion of accumulation terms, transitions from under-saturated to saturated states and vice versa, high gas injection rates, evolution of gas in the oil production wells and adaptive time-stepping procedures are elaborated.

Findings- It was shown that using a proper linearisation method, less computational difficulties occur especially when free gas is released with high rates. The computational performance of the proposed algorithm is assessed by solving the first SPE comparative study problem with both constant and variable bubble point conditions.

Research limitations/implications- While discretization is performed and implemented for unstructured grids, as expected, the accuracy of numerical results are best for structured grids. Also, the reservoir is assumed to be non-fractured.

Practical implications- The proposed algorithm can be efficiently used for simulating a wide

range of practical problems wherever blackoil model is applicable.

Originality/value- A complete and detailed description of ingredients of an efficient finite volume based algorithm for simulating blackoil flows in hydrocarbon reservoirs is presented.

Keywords Blackoil Model, Hydrocarbon Reservoir, Finite Volume Method, IMPES

Paper type Research paper

Introduction

Hydrocarbon reservoir simulation has become an indispensable tool in oil and gas industry. The accuracy and computational efficiency of simulations have been constantly improved in past few decades and more and more complex physical models been incorporated in simulators. There are well established commercial softwares which are used worldwide and various scientific groups have successfully developed formulations and numerical approaches to deal with practical problems. Despite these achievements, it seems that there is a gap between the reservoir engineers and most of the computational engineers. This is mainly due to several less familiar issues involved in the field of reservoir engineering. This paper attempts to fill this gap by presenting a detailed formulation describing both physical and computational complexities.

In this work, a blackoil model (Aziz and Settari 1979) is used to describe the behaviour of reservoir fluids. This is a relatively simple physical model which can be efficiently used for many practical hydrocarbon reservoir applications. The blackoil model is a three-phase three-component model. Here, the term *phase* refers to a bulk of fluid with continuous physical properties, while the term *component* refers to each specific chemical pseudo-component. In the blackoil model, it is assumed that the reservoir fluid is distributed in water, liquid hydrocarbon and gaseous hydrocarbon phases. The water phase (aqua) consists of water component only. Also, the gaseous hydrocarbon phase (vapor phase) can only contain light hydrocarbon component (gas component). The liquid hydrocarbon phase (liquid phase) can contain both gas and oil components depending on reservoir conditions. When both liquid and vapor phases exist, the reservoir status is called *saturated*. In certain situations, the reservoir pressure is above its bubble point pressure and all of the gas component dissolves in the liquid phase. In such cases, the reservoir status is called *under-saturated*.

A crucial step in developing an efficient blackoil simulator is the choice of mathematical formulation. The blackoil model has been described using different mathematical formulations each has its own computational merits (Chen 2000, Trangenstein and Bell 1989, Almehaideb and Aziz 1989, Coats 1989, Coats et al. 1998). A critical feature of such formulations is their capability to handle physical nonlinearities and switching between saturated and under-saturated conditions. Such difficulties can lead to various numerical problems to produce a physically meaningful solution. A particularly convenient formulation was introduced by Coates and collaborators (Coats et al. 1998, Coats 1989) which has been extended to various applications. This formulation was originally devised for a compositional model, and a simplified form of it

was used for blackoil models. In the present paper, a variant of this formulation is used.

The numerical modelling of multi-phase flow in porous media has been extensively studied using different computational approaches such as Finite Difference, Finite Element and Finite Volume Element methods (Li et al. 2004, Trangenstein and Bell 1989, Qin et al. 2002, Chavent and Jaffré 1986, Bergamaschi et al. 1999, Pao et al. 2001, Pao and Lewis 2002). There are a number major research centres worldwide which specialize in reservoir simulations. Among these, the Center for Subsurface Modeling (CSM) of the University of Texas at Austin directed by M. Wheeler (Wheeler et al. 2000, Wheleer 2002, Thomas et al. 2011, Salama et al. 2014) and the SUPRI-B group at Stanford (Schiozer 1994, Verma 1996) have made great advances in this field. In recent years, Edwards and his collaborators have made significant developments on the use of Multi-point Flux Approximations methods for simulating flow in anisotropic porous media (Edwards 1998, Edwards 2002, Edwards 2008).

A particularly versatile method which has been widely accepted in the field of Computational Fluid Dynamics (CFD), is the finite volume method. The method enjoys being locally conservative, easily conceivable and can be implemented on both structured and unstructured grids. There is also a rich literature on this method in the context of CFD (see for example, (LeVeque 2002, Hirsch 2007)).

The temporal discretization of the governing equations can be achieved using different methods from explicit to fully implicit. Also, the order by which governing equations are solved can have a significant effect on the accuracy and stability of methods. Four major categories for this are the sequential, simultaneous solution, (variants of) the implicit pressure-explicit saturation (IMPES) and implicit methods. As the governing equations are highly coupled, one may prefer to use fully implicit methods as larger time steps can be used. This however can lead to prohibitively heavy computations in practical problems. A fairly convenient approach is the use of variants of the IMPES method. In this method a pressure equation is derived by manipulating the governing equations and is solved implicitly to obtain the pressure and flow fields consecutively. The computed flow field is then used to solve the transport equations representing the mass conservation of components. In this work, an iterative IMPES method is adopted.

This paper presents a numerical algorithm for simulating a three dimensional and three phase isothermal flow in heterogeneous reservoirs based on the blackoil model. The mathematical formulation is derived from the original formulation introduced by Coats et al. (Coats et al. 1998). However, the computational implementation has been revised and significant details have been added. In this paper, without loss of generality, the spatial discretizations are performed on orthogonal structured grids using a two-point cell-centred finite volume method.

In the following, first the blackoil formulation is presented and the process of derivation of the pressure equation, its linearisation and spatial diceretization are elaborated. Then, a number of special numerical treatments, well models and a complete structure of computer program

are described. Finally, to assess the computational performance of the proposed algorithm, the First SPE Comparative Problem is solved for both constant and variable bubble point cases and some conclusions are drawn.

Blackoil Formulation

The governing equations consist of the mass conservation equations for each component (Coats et al. 1998). For a full description of variables and their units, refer to Nomenclature. For the oil component, the conservation of mass reads

$$\frac{\partial}{\partial t}(\phi \rho_o S_o x_o) = -\nabla \cdot (\rho_o x_o \mathbf{u}_o) - \tilde{q}_o, \tag{1}$$

while for the gas component, one has

$$\frac{\partial}{\partial t}\left[\phi \left(\rho_o x_g S_o + \rho_g S_g\right)\right] = -\nabla \cdot (\rho_o x_g \mathbf{u}_o + \rho_g \mathbf{u}_g) - \tilde{q}_g, \tag{2}$$

and for the water component, the conservation of mass equation becomes

$$\frac{\partial}{\partial t}(\phi \rho_w S_w) = -\nabla \cdot (\rho_w \mathbf{u}_w) - \tilde{q}_w. \tag{3}$$

Here, $\tilde{q}_i = q_i/V$ refers to molar flux of component i per unit volume. In these equations, the phase velocities $(\mathbf{u_o}, \mathbf{u_g}, \mathbf{u_w})$ are found using the Darcy law as (Coats 1989)

$$\mathbf{u_o} = -\lambda_o \left(\nabla p_g - \nabla p_{cgo} - \gamma_o \nabla Z\right), \tag{4}$$

$$\mathbf{u_g} = -\lambda_g \left(\nabla p_g - \gamma_g \nabla Z\right), \tag{5}$$

$$\mathbf{u_w} = -\lambda_w \left(\nabla p_g - \nabla p_{cgo} - \nabla p_{cwo} - \gamma_w \nabla Z\right), \tag{6}$$

where mobility of phase α is given by

$$\lambda_\alpha = \mathbf{K}\frac{k_{r\alpha}}{\mu_\alpha}, \qquad \alpha = \{o, g, w\} \tag{7}$$

with $\mathbf{K}$, μ_α and $k_{r\alpha}$ being the rock absolute permeability tensor, and the viscosity and relative permeability of phase α, respectively. The vector of unknowns associated with these equations is $\mathbf{U} = \{x_o, x_g, S_o, S_g, S_w, p_g\}^T$. Moreover, the following constraints apply

$$x_o + x_g = 1, \tag{8}$$

$$S_o + S_g + S_w = 1. \tag{9}$$

Depending on whether free gas exists or not, a different sets of primary variables must be used. In numerical simulations, this is decided based on the status of each computational cell.

When a cell is saturated (contains free gas), the vector of unknowns becomes $\{S_o, S_g, p_g\}^T$, while x_g and x_o are functions of pressure (Coats et al. 1998). In a blackoil system these functions are normally given in a tabular format for the Pressure Volume Temperature (PVT) data. In the case of under-saturated cells (no free gas), the vector of unknowns becomes $\{S_o, x_g, p_g\}^T$ (Coats et al. 1998). It should be noted that as there is no gas in the system, the gas pressure p_g has no physical meaning but is merely a mathematical variable (Coats et al. 1998).

Derivation of Pressure Equation

As explained before, a particularly useful technique for solving the system of equations governing multiphase flows in porous media is the IMPES approach. In this approach, a pressure equation is derived by manipulating the conservation equations. This non-linear second-order equation is discretized on the computational domain and is solved implicitly to obtain the pressure distribution in the reservoir. Using the Darcy law, one can use this pressure distribution to obtain the velocity field in the domain. In the final stage of the IMPES approach, the mass conservation equations are solved explicitly to determine saturation of different phases in the reservoir.

In order to derive the pressure equation, it is necessary to expand the accumulation terms (left hand side of Eqs. (1) and (2)) according to the status of each computational cell and the set of primary variables used at the beginning and end of each time step (Farnstrom and Ertekin 1987). For a typical computational cell, in general, one of the following four conditions may occur during one time step (Farnstrom and Ertekin 1987):

- Free gas exists in the whole time step (Case 1)
- No free gas exists in the whole time step (Case 2)
- Free gas disappears during the time step (Case 3)
- Free gas appears during the time step (Case 4)

At the beginning of each time step, the status of each cell can be easily determined knowing its gas saturation value. During a time step, the estimated cell bubble point pressure is used to determine the state of the cell. If the cell pressure is below or fall under the bubble point pressure an amount of free gas will exist or be released. On the other hand, if the cell pressure is greater than the estimated cell bubble point pressure no free gas will exist. Finally, if the cell pressure increases from below to above the bubble point pressure all free gas will dissolve into the liquid phase. In the followings, mathematical formulations are given for each of these cases.

Expansion of accumulation terms

As mentioned before, a key part of the derivation of pressure equation is expansion of accumulation terms. This procedure is elaborated here. For all four cases mentioned above, the

expanded form of the left hand side of Eq. (3) (the water accumulation term) remains the same. Noting that $S_o + S_g + S_w = 1$, the following relation holds for the water component

$$\frac{\partial}{\partial t}(\phi \rho_w S_w) = -\phi \rho_w \frac{\partial S_o}{\partial t} - \phi \rho_w \frac{\partial S_g}{\partial t} + S_w \frac{\partial (\phi \rho_w)}{\partial p_g} \frac{\partial p_g}{\partial t}. \tag{10}$$

When no free gas exists during a time step, one has $\partial S_g / \partial t = 0$.

Cases without status alteration

In this part, two cases in which no status alteration occurs are considered. This means that the cell pressure remains either below or above the bubble point pressure during the whole time step.

Free gas exists during the time step (Case 1)

In this case, the cell pressure remains below the calculated bubble point pressure of the cell and the cell conditions at the beginning and end of the time step are the same. Therefore, the primary variables during the time step are S_o, S_g, and p_g, and one can write

$$\frac{\partial}{\partial t}(\phi \rho_o S_o x_o) = \phi \rho_o x_o \frac{\partial S_o}{\partial t} + S_o \frac{\partial (\phi \rho_o x_o)}{\partial p_g} \frac{\partial p_g}{\partial t}, \tag{11}$$

$$\frac{\partial}{\partial t}\left[\phi \left(\rho_o x_g S_o + \rho_g S_g\right)\right] = \phi \rho_o x_g \frac{\partial S_o}{\partial t} + \phi \rho_g \frac{\partial S_g}{\partial t} + \left[S_o \frac{\partial (\phi \rho_o x_g)}{\partial p_g} + S_g \frac{\partial (\phi \rho_g)}{\partial p_g}\right] \frac{\partial p_g}{\partial t}. \tag{12}$$

Note that ϕ, ρ_o, and x_g are considered to be functions of the gas pressure, p_g.

No-free gas exists during the time step (Case 2)

In this case, the cell pressure is always above the bubble point pressure, no free gas is present, and there is no change in the state of the cell during the time step. Therefore, the primary variables are chosen as S_o, x_g, and p_g, and the accumulation terms are expanded as

$$\frac{\partial}{\partial t}(\phi \rho_o x_o S_o) = \phi \rho_o x_o \frac{\partial S_o}{\partial t} + \phi S_o \frac{\partial (\rho_o x_o)}{\partial x_g} \frac{\partial x_g}{\partial t} + S_o x_o \frac{\partial (\phi \rho_o)}{\partial p_g} \frac{\partial p_g}{\partial t}, \tag{13}$$

$$\frac{\partial}{\partial t}(\phi \rho_o x_g S_o) = \phi \rho_o x_g \frac{\partial S_o}{\partial t} + \phi S_o \frac{\partial (\rho_o x_g)}{\partial x_g} \frac{\partial x_g}{\partial t} + S_o x_g \frac{\partial (\phi \rho_o)}{\partial p_g} \frac{\partial p_g}{\partial t}. \tag{14}$$

Note that ρ_o is a function of both p_g and x_g, while ϕ is only a function of p_g.

Cases with status alteration

When the pressure of a cell crosses the bubble point pressure, strong non-linearity is introduced into the governing equations (Farnstrom and Ertekin 1987). This is because the variables like x_g, x_o and ρ_o, and their derivatives with respect to gas pressure need to be calculated at two points 'u' and 's' on opposite sides of the bubble point pressure 'b', as shown in Fig. 1. In such situations, the expansion terms for oil density can be modified to decrease the non-linearity of equations. This can be achieved by considering effect of the bubble point pressure in the

expansion of accumulation terms. In fact, the cell conditions at time t_n are first shifted to the corresponding bubble point and then the simulation is performed. For the case in which gas appears during a time step, the simulation starts from a typical point 'u' (Fig. 1). Then, during the time step, the cell condition approaches point 's' along chord 'u-s'. In such situations, one can shift the cell condition to its corresponding bubble point at the beginning of time step. Then, continue the simulation process along chord 'b-s'. Since both the start and end points of the time step are on one smooth chord, 'b-s', the non-linearity of equations is decreased (Thomas et al. 1976). To move the cell condition to the corresponding bubble point, one can modify the temporal variations of oil phase density. The temporal variation of ρ_o can be expressed by adding and subtracting the oil density associated with the bubble point pressure at time t_{n+1}, ρ_{ob}^{n+1}, as (Thomas et al. 1976)

$$\frac{\partial \rho_o}{\partial t} = \frac{\rho_o^{n+1} - \rho_{ob}^{n+1} + \rho_{ob}^{n+1} - \rho_o^n}{\Delta t} = \frac{\partial \rho_o}{\partial p_g}\Big|_{sat} \frac{p_g^{n+1} - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}, \tag{15}$$

where subscript 'sat' refers to the saturated state. In fact, the term $\partial \rho_o / \partial p_g\Big|_{sat}$ represents the slop of the saturated chord 'b-s', and it can be calculated as $(\rho_o^{n+1} - \rho_{ob}^{n+1})/(p_g^{n+1} - p_b^{n+1})$. By adding and subtracting p_g^n from $p_g^{n+1} - p_b^{n+1}$ in Eq. (15) and after some manipulations, one obtains

$$\frac{\partial \rho_o}{\partial t} = \frac{\partial \rho_o}{\partial p_g}\Big|_{sat} \frac{\partial p_g}{\partial t} + \frac{\partial \rho_o}{\partial p_g}\Big|_{sat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}. \tag{16}$$

Similarly, when all free gas in the cell dissolves in the oil phase during a time step, one can write

$$\frac{\partial \rho_o}{\partial t} = \frac{\partial \rho_o}{\partial p_g}\Big|_{Usat} \frac{\partial p_g}{\partial t} + \frac{\partial \rho_o}{\partial p_g}\Big|_{Usat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}, \tag{17}$$

where the subscript '$Usat$' refers to the under-saturated sate. Note that the derivative $\partial \rho_o / \partial p_g\Big|_{Usat}$ is calculated using the slop of chord 'b-u' as $(\rho_o^{n+1} - \rho_{ob}^{n+1})/(p_g^{n+1} - p_b^{n+1})$. Equations (16) and (17) are then used to expand the accumulation terms for cases 3 and 4.

Dissolution of all free gas (Case 3):

In this case, at the beginning of the time step, the cell pressure is below the estimated bubble point pressure (e.g. point 's' in Fig.1). During the time step, the cell pressure increases to reach a new value (point 'u' in Fig.1), which is greater than the corresponding bubble point value (point 'b'). In this case, all free gas dissolves into the liquid phase. As the states of the cell at the beginning and end of the time step are different, two different sets of variables need to be used. At the beginning of the time step, the primary variables are set to be the same as the saturated state (Case 1), while during the time step, the primary variables are switched to

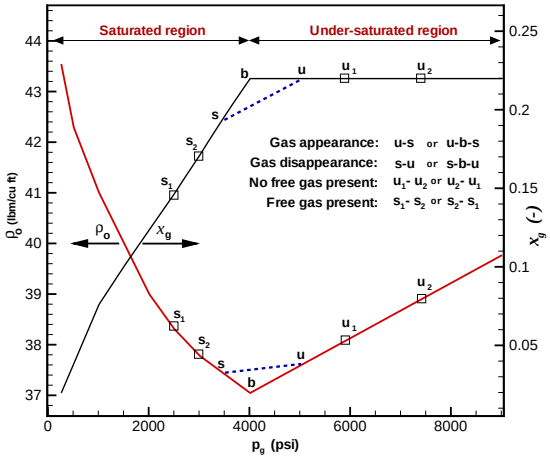


Figure 1: The variation of liquid phase density and gas mole fraction versus gas pressure with a constant bubble point pressure $p_b = 4014.7(psi)$. Three points 's' and 'b' and 'u' represent a typical saturated state, the bubble point and a typical under-saturated state for a computational cell, respectively.

S_o, x_g, and p_g. Therefore, one can write

$$\frac{\partial}{\partial t}(\phi\rho_o x_o S_o) = \phi\rho_o x_o \frac{\partial S_o}{\partial t} - \phi\rho_o S_o \frac{\partial x_g}{\partial t} + S_o x_o \left(\rho_o \frac{\partial \phi}{\partial p_g} + \phi \frac{\partial \rho_o}{\partial p_g}\Big|_{Usat}\right) \frac{\partial p_g}{\partial t} + R_o, \tag{18}$$

$$\frac{\partial}{\partial t}(\phi\rho_o x_g S_o) = \phi\rho_o x_g \frac{\partial S_o}{\partial t} + \phi\rho_o S_o \frac{\partial x_g}{\partial t} + S_o x_g \left(\rho_o \frac{\partial \phi}{\partial p_g} + \phi \frac{\partial \rho_o}{\partial p_g}\Big|_{Usat}\right) \frac{\partial p_g}{\partial t} + R_g, \tag{19}$$

where R_o and R_g are known residual terms and are defined as

$$R_o = \phi S_o x_o \left(\frac{\partial \rho_o}{\partial p_g}\Big|_{Usat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right),$$

$$R_g = \phi S_o x_g \left(\frac{\partial \rho_o}{\partial p_g}\Big|_{Usat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right).$$

Appearance of free gas (Case 4):

In this case, at the beginning of the time step, the cell pressure is above the corresponding bubble point pressure (point 'u' in Fig. 1), thus, no free gas is present. During the time step, the cell pressure decreases to reach a new value (point 's'), which is less than the cell bubble point pressure (point 'b'), and free gas is released from the liquid phase. As the state of the cell at the beginning and the end of the time step are different, two different sets of variables need to be used. At the beginning, the primary variable are set to S_o, x_g, and p_g (the expansion of accumulation terms is the same as Case 2), while when the free gas appears the variables are switched to S_o, S_g, and p_g. Thus, one can write

$$\frac{\partial}{\partial t}(\phi\rho_o x_o S_o) = \phi\rho_o x_o \frac{\partial S_o}{\partial t} + S_o \left(\rho_o x_o \frac{\partial \phi}{\partial p_g} - \phi\rho_o \frac{\partial x_g}{\partial p_g} + \phi x_o \frac{\partial \rho_o}{\partial p_g}\Big|_{sat}\right) \frac{\partial p_g}{\partial t} + R_o, \tag{20}$$

$$\frac{\partial}{\partial t}\left[\phi\left(\rho_o x_g S_o + \rho_g S_g\right)\right] = \phi\rho_o x_g \frac{\partial S_o}{\partial t} + \phi\rho_g \frac{\partial S_g}{\partial t} + \left[S_o\left(\rho_o x_g \frac{\partial \phi}{\partial p_g} + \phi\rho_o \frac{\partial x_g}{\partial p_g} + \phi x_g \frac{\partial \rho_o}{\partial p_g}\Big|_{sat}\right) + S_g \frac{\partial(\phi\rho_g)}{\partial p_g}\right]\frac{\partial p_g}{\partial t} + R_g, \tag{21}$$

where, R_o and R_g are known residual terms defined as

$$R_o = \phi S_o x_o \left(\frac{\partial \rho_o}{\partial p_g}\Big|_{sat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right), \tag{22}$$
$$R_g = \phi S_o x_g \left(\frac{\partial \rho_o}{\partial p_g}\Big|_{sat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right).$$

The pressure equations

The right hand sides of Eqs. (11) to (14) and (18) to (10) can be written in the matrix form $\mathbf{C}\frac{\partial \mathbf{U}}{\partial \mathbf{t}} + \mathbf{r}$ where $\mathbf{U}$ is the vector of primary variables, $\mathbf{r} = \{R_o, R_g, 0\}^T$ is the residual vector, and $\mathbf{C}$ is a square 3×3 matrix whose entities are given in Table 1. In addition, matrices $\mathbf{A}$ and $\mathbf{B}$ are defined as (Coats et al. 1998)

$$\mathbf{A} = \begin{bmatrix} C_{11} & C_{21} \\ C_{12} & C_{22} \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} C_{31} \\ C_{32} \end{bmatrix}, \tag{23}$$

and coefficients v_1 and v_2 are defined such that (Coats et al. 1998)

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \mathbf{A}^{-1}\mathbf{B}. \tag{24}$$

By choosing $v_3 = -1$, one can write

$$\begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix} \mathbf{C}\frac{\partial \mathbf{U}}{\partial t} = \left(v_1 C_{13} + v_2 C_{23} + v_3 C_{33}\right)\frac{\partial p_g}{\partial t}. \tag{25}$$

It is noticed that by applying this procedure the time derivatives of accumulation terms vanish from the equations and only time derivatives of gas pressure remain. Applying the same procedure to all other terms in the mass conservation equations (1)-(3), one obtains the following general pressure equation for each cell

$$c\frac{\partial p_g}{\partial t} = \sum_{j=1}^{3} v_j \nabla \cdot \left(a_j \nabla p_g\right) - R, \tag{26}$$

where $c = v_1 C_{13} + v_2 C_{23} + v_3 C_{33}$. In addition, v_1,v_2, and $a_j, j \in \{1, 2, 3\}$ are functions of the

Table 1: The coefficients of the matrix $\mathbf{C}$, the residual terms R_o, R_g and the other parameters in the pressure equation for different status of reservoir cells.

Parameter	Case1 (Saturated)	Case2 (Under-saturated)	Case3 (Gas disappearance)	Case4 (Gas appearance)
C_{11}	$\phi\rho_o x_o$	$\phi\rho_o x_o$	$\phi\rho_o x_o$	$\phi\rho_o x_o$
C_{12}	0	$\phi S_o \frac{(\partial\rho_o x_o)}{\partial x_g}$	$-\phi\rho_o S_o$	0
C_{13}	$S_o \frac{\partial(\phi\rho_o x_o)}{\partial p_g}$	$S_o x_o \frac{\partial(\phi\rho_o)}{\partial p_g}$	$S_o x_o \left[\rho_o \frac{\partial\phi}{\partial p_g} + \phi \frac{\partial\rho_o}{\partial p_g}\Big\vert_{Usat}\right]$	$S_o \left[\rho_o x_o \frac{\partial\phi}{\partial p_g} - \phi\rho_o \frac{\partial x_g}{\partial p_g} + \phi x_o \frac{\partial\rho_o}{\partial p_g}\Big\vert_{sat}\right]$
C_{21}	$\phi\rho_o x_g$	$\phi\rho_o x_g$	$\phi\rho_o x_g$	$\phi\rho_o x_g$
C_{22}	$\phi\rho_g$	$\phi S_o \frac{(\partial\rho_o x_g)}{\partial x_g}$	$\phi\rho_o S_o$	$\phi\rho_g$
C_{23}	$S_o \frac{\partial(\phi\rho_o x_g)}{\partial p_g} + S_g \frac{\partial(\phi\rho_g)}{\partial p_g}$	$S_o x_g \frac{\partial(\phi\rho_o)}{\partial p_g}$	$S_o x_g \left[\rho_o \frac{\partial\phi}{\partial p_g} + \phi \frac{\partial\rho_o}{\partial p_g}\Big\vert_{Usat}\right] - S_g \frac{\partial(\phi\rho_g)}{\partial p_g}$	$S_o \left[\rho_o x_g \frac{\partial\phi}{\partial p_g} + \phi x_g \frac{\partial\rho_o}{\partial p_g}\Big\vert_{sat} + \phi\rho_o \frac{\partial x_g}{\partial p_g}\right] + S_g \frac{\partial(\phi\rho_g)}{\partial p_g}$
C_{31}	$-\phi\rho_w$	$-\phi\rho_w$	$-\phi\rho_w$	$-\phi\rho_w$
C_{32}	$-\phi\rho_w$	0	0	$-\phi\rho_w$
C_{33}	$S_w \frac{\partial(\phi\rho_w)}{\partial p_g}$	$S_w \frac{\partial(\phi\rho_w)}{\partial p_g}$	$S_w \frac{\partial(\phi\rho_w)}{\partial p_g}$	$S_w \frac{\partial(\phi\rho_w)}{\partial p_g}$
R_o	0	0	$\phi S_o x_o \left[\frac{\partial\rho_o}{\partial p_g}\Big\vert_{Usat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right]$	$\phi S_o x_o \left[\frac{\partial\rho_o}{\partial p_g}\Big\vert_{sat} \frac{(p_g^n - p_b^{n+1})}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right]$
R_g	0	0	$\phi S_o x_g \left[\frac{\partial\rho_o}{\partial p_g}\Big\vert_{Usat} \frac{p_g^n - p_b^{n+1}}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right]$	$\phi S_o x_g \left[\frac{\partial\rho_o}{\partial p_g}\Big\vert_{sat} \frac{(p_g^n - p_b^{n+1})}{\Delta t} + \frac{\rho_{ob}^{n+1} - \rho_o^n}{\Delta t}\right]$
a_1	$\rho_o x_o \lambda_o$	$\rho_o x_o \lambda_o$	$\rho_o x_o \lambda_o$	$\rho_o x_o \lambda_o$
a_2	$\rho_o x_g \lambda_o + \rho_g \lambda_g$	$\rho_o x_g \lambda_o + \rho_g \lambda_g$	$\rho_o x_g \lambda_o + \rho_g \lambda_g$	$\rho_o x_g \lambda_o + \rho_g \lambda_g$
a_3	$\rho_w \lambda_w$	$\rho_w \lambda_w$	$\rho_w \lambda_w$	$\rho_w \lambda_w$

gas pressure, p_g, and given in Table 1. The total residual term R is given by

$$\begin{aligned} R =& v_1 \nabla \cdot [a_1 (\nabla P_{cgo} + \gamma_o \nabla Z)] \\ &+ v_2 \nabla \cdot [\rho_o x_g \lambda_o \nabla p_{cgo} + \rho_o x_g \lambda_o \gamma_o \nabla Z + \rho_g \lambda_g \gamma_g \nabla Z] \\ &+ v_3 \nabla \cdot [a_3 (\nabla p_{cgo} + \nabla p_{cwo} + \gamma_w \nabla Z)] \\ &+ v_1 \tilde{q}_o + v_2 \tilde{q}_g + v_3 \tilde{q}_w - v_1 R_o - v_2 R_g, \end{aligned} \tag{27}$$

which contains both the capillary pressure and gravity effects, along with mass fluxes due to wells.

Linearisation

The pressure equation (26) is nonlinear and needs proper linearisation before discretization process. For a typical nonlinear term X, variation in X during one time step is given by $\overline{\delta} X = X^{n+1} - X^n$ and during one iteration is shown by $\delta X = X^{l+1} - X^l$, where superscript n refers to time t_n while superscript l refers to an iteration level. Therefore, one can write (Coats et al. 1998)

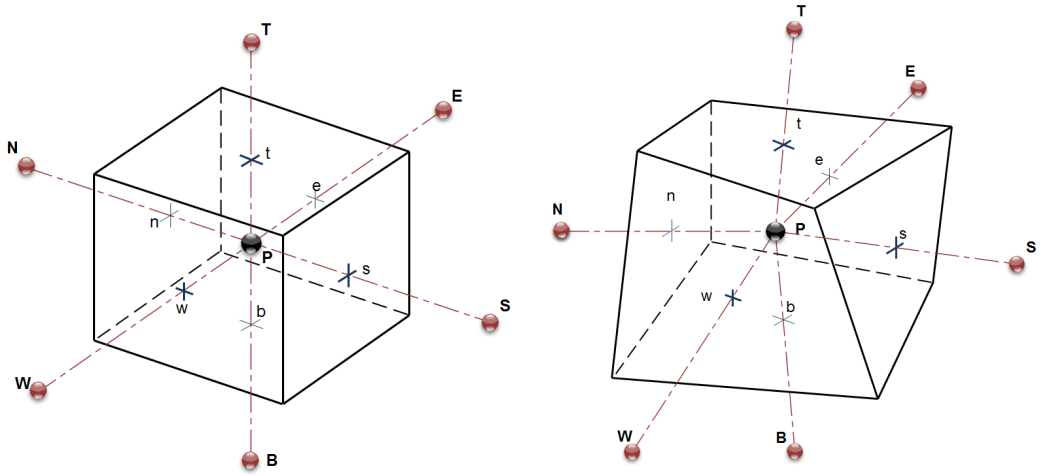
$$\overline{\delta} X \cong \delta X + X^l - X^n. \tag{28}$$

In limit, $X^{l+1} \to X^{n+1}$ and the above equation becomes an equality. Also, one has

$$X^{n+1} \cong X^{l+1} = X^l + \delta X. \tag{29}$$

Using this approximation, Eq. (26) becomes

$$c^l \frac{p_g^{l+1} - p_g^n}{\Delta t} = v_1^l \nabla \cdot \left(a_1^l \nabla p_g^{l+1}\right) + v_2^l \nabla \cdot \left(a_2^l \nabla p_g^{l+1}\right) + v_3^l \nabla \cdot \left(a_3^l \nabla p_g^{l+1}\right) - R^l. \tag{30}$$



(a) Orthogonal (b) Non-orthogonal

Figure 2: Typical internal cells in three-dimensional orthogonal and non-orthogonal grids.

The iterative algorithm used in this work to solve the above nonlinear equations is shown in Fig. 3a.

Discretization of Equations

In this work two kinds of time dependent equations need to be discretized; a parabolic pressure equation and a set of hyperbolic transport equations. As mentioned before, the pressure equation is solved implicitly in time while the transport equations are solved using the explicit Euler time integration scheme. Both pressure and transport equations are discretized in space using a Finite Volume method implemented on orthogonal structured grids. The computational domain is discretized using a set of non-overlapping computational cells forming the computational grid. Figure 2 shows two typical three-dimensional cells along with their neighbouring cell centres for both orthogonal and non-orthogonal grids. Here, a cell centred approach is used with centre of the cell being positioned at the geometrical centre of the cell. Primitive variables are assumed to be uniform within each computational cell.

From a discretization point of view in the context of a finite volume method, the terms in the pressure and transport equations, can be generally divided into four categories; accumulation, source, convection and diffusion terms. While the discretization of accumulation and source terms is straightforward, the discretization of convection and diffusion terms needs special considerations. There is an extensive literature on this subject (LeVeque 2002, Eymard et al. 2000, Versteeg and Malalasekera 2007, Lee et al. 2008) hence only some general implementation issues are highlighted here.

Spatial Discretization of Pressure Equation

In order to discretize the pressure equation, one can integrate Eq. (26) over a typical computational cell, which also plays the role of a control volume, CV (Versteeg and Malalasekera 2007). This leads to

$$\int_{CV} c\frac{\partial p_g}{\partial t}\,dV = \int_{CV} \sum_{j=1}^{3} v_j \nabla \cdot (a_j \nabla p_g)\; dV - \int_{CV} R\,dV, \tag{31}$$

where V refers to the volume of the control volume. Since primitive variables are assumed to be uniform within each computational cell, the left hand side of Eq. (31) becomes $c \times V \times \partial p_g/\partial t$. The partial derivative in time is then approximated by Euler's forward difference scheme.

All integrals on left hand side of Eq. (31) have integrands of a general form $\nabla \cdot (\Gamma \nabla \varphi)$ in which Γ is a diffusivity-type coefficient and φ belongs to $\{p_g, p_{cgo}, p_{cwo}, Z\}$. In the context of the finite volume method, such terms are converted into surface integrals using Gauss divergence theorem (Versteeg and Malalasekera 2007). Each surface integral is, in turn, replaced by a summation over all boundary faces of the control volume. For a typical diffusion term this leads to

$$D(\Gamma, \varphi) \equiv \int_{CV} \nabla \cdot (\Gamma \nabla \varphi)\, dV = \int_{A} \Gamma \nabla \varphi \cdot \mathbf{n}\, dA = \sum_{f} \Gamma_f A_f \left(\nabla \varphi \cdot \mathbf{n}\right)_f, \tag{32}$$

where the summation is over all cell faces, e.g. $f \in \{e, w, n, s, b, t\}$ for the cell shown in Fig. 2a. In addition, Γ_f is the diffusivity coefficient computed on face f, A_f is surface area of the face, and $\mathbf{n}$ is an outward pointing unit vector normal to face f. Different averaging methods are used to estimate the fluid and rock properties on cells faces. While properties such as the gas mole fraction, and phase density and viscosity, are calculated by the weighted arithmetic averaging method, the absolute permeability is estimated by a weighted harmonic mean on the cell faces (Chen et al. 2006). In addition, a first order upwind method is used to calculate the phase relative permeabilities on faces of cells (Chen et al. 2006).

To approximate the diffusion fluxes $\Gamma_f A_f \left(\nabla \varphi \cdot \mathbf{n}\right)_f$ crossing face f of the computational cell, one needs to compute $(\nabla \varphi)_f$. Here, an iterative scheme is used to compute gradients at the centre of faces (Hadzic 2005). In this approach, first, for each face shared between two computational cells, a new point j is found. The point j is the normal projection of the point F (center of the face) to the line d connecting centres of two neighbouring cells. Then, φ_j is estimated using linear interpolation between the associated cell-centre values. As the first estimation, the face-centre value φ_f is set equal to φ_j. Then, the estimated values of face centres (φ_f) are used to compute the gradients $(\nabla \varphi)$ at the centres of computational cells. This can

be achieved by computing the following integral using finite volume method

$$\nabla\varphi \equiv \frac{1}{V}\int \nabla\varphi \, dV = \frac{1}{V}\sum_f A_f \varphi_f . \mathbf{n}_f. \tag{33}$$

Then, the values of $\nabla\varphi$ are interpolated between two neighbouring cells to calculate $\nabla\varphi_j$. Finally, these gradients are utilized along with φ_j values to update values of φ_f using $\varphi_f = \varphi_j + \nabla\varphi_j.\mathbf{d}$, where $\mathbf{d}$ is the vector connecting point j to the centre of the face f. This procedure is repeated until 2-norm of gradient vector converges within a predefined tolerance (here, $\epsilon = 10^{-6}$). This method converges within a few number of iterations (typically less than 5) for nearly orthogonal grids (refer to (Hadzic 2005) for more details). Using the converged cell centre gradients, all variables at face centres can be evaluated with second-order accuracy (Peric, M. 2004). To do this, for each face shared between two computational cells, two new points are introduced. Each of these new points is the normal projection of a cell centre to the face normal line. Using $\nabla\varphi$ and φ at the centres of the cells, the value of the variable at each new point (φ_{np}) is evaluated according to $\varphi_{np} = \varphi + \nabla\varphi.\mathbf{r}$. Here, $\mathbf{r}$ is the vector connecting the new point to its corresponding cell centre. Finally, the face values are computed by interpolating between the values associated with these two new points.

While all terms associated with the first integral in Eq. (31) are expressed in an implicit form, the second integral on the right hand side of this equation is written in an explicit form. Note that since coefficient Γ is a function of gas pressure, it is computed at iteration level l. For a six-side cell, this leads to a 7-point stencil formulation. After assembling all discretized equations in a matrix form, one obtains a linear system of equations which needs to be solved for p_g. For more details on the solution algorithm of the pressure equation refer to Fig. 3a and 3b.

Spatial Discretization of Transport Equations

Each transport equation can be written in the general form

$$\frac{\partial(\beta U)}{\partial t} = -\nabla.(F\mathbf{u}) - \tilde{q} - R_s, \tag{34}$$

where β and F coefficients consist of a number of known parameters given on cell centres and $R_s \in \{R_o, R_g, 0\}$ represents the residual term. Term $\mathbf{u}$ is the known velocity vector, $\tilde{q} = q/V$ is the molar flux per unit volume due to the source terms, and U is the main variable. Integrating this general form over each computational cell, one obtains

$$V\frac{\partial(\beta U)}{\partial t} = -\int_{\Omega_i} \nabla\cdot(F\mathbf{u})dV - q - \int_{\Omega_i} R_s dV. \tag{35}$$

The first integral on the right hand side can be calculated by (Greenshields et al. 2010)

$$\mathrm{J}(\mathbf{u}, F) \equiv \int_{\Omega_i} \nabla \cdot (F\mathbf{u}) dV = \sum_f F_f A_f \mathbf{u}_f . \mathbf{n}_f, \tag{36}$$

where $\sum$ indicates summation over all cell faces $f \in \{e, w, n, s, b, t\}$. In addition, F_f represents the value of coefficient F calculated on face f. Note that the face centre values are computed using the cell centre values and their cell centre gradients, as described above. A_f is the surface area, and $\mathbf{n}_f$ is a unit vector pointing out of the face f. Therefore, the discretized form of Eq. (34) is written as

$$V \frac{(\beta U)^{n+1} - (\beta U)^n}{\Delta t} = -\mathrm{J}(\mathbf{u}, F) - q - \frac{V R_s}{\Delta t}. \tag{37}$$

This general discretized equation is used for solving the transport equations.

Special Treatments

In this section, the numerical treatments applied for two special cases, i.e. negative gas saturation and excessive dissolved gas in the liquid phase, are given. Before this, calculations of the gas mole fraction and the bubble point pressure, are described.

Calculation of gas mole fraction using the saturation pressure: The gas mole fraction in the liquid phase is calculated as a function of the saturation pressure, $x_g = x_g(p_{sat})$, using the given PVT data. The saturation pressure, p_{sat}, is the cell pressure for cells containing free gas (saturated cells), and is the bubble point pressure for under-saturated cells (Aziz and Settari 1979).

Estimation of bubble point pressure: At the beginning of each time step, one needs to estimate the bubble point pressure to predict the status of a computational cell. For reservoir cells containing free gas, the bubble point pressure is estimated by placing all available free and injected gas into the liquid phase (Thomas et al. 1976). Therefore, the gas mole fraction at the bubble point pressure, x_{gb}, can be calculated by

$$x_{gb} = x_g^n + \frac{\rho_g^n s_g^n}{\rho_o^n s_o^n} - \frac{q_g \Delta t}{\rho_o^n S_o^n \phi^n V}. \tag{38}$$

For cells without free and injected gas, one can simply set $x_{gb} = x_g^n$. The estimated values of x_{gb} are then used to compute the bubble point pressure from the given PVT data as $p_b^n = p_b^n(x_{gb})$.

There are two cases for which the solution obtained by solving the transport equations should be modified to achieve correct mass balance and physically meaningful results. These two cases are described in the following.

Negative gas saturation: This case occurs when the simulator adds more free gas to the liquid phase than what is available. In this case, the summation of liquid and aqua phase saturations is greater than one, and $s_g^{n+1} < 0$ (Farnstrom and Ertekin 1987). Therefore, the calculated oil and gas saturations should be modified. To do this, the water saturation and volume balance approach is used to modify the liquid phase saturation as

$$S_o^{n+1} = 1 - S_w^{n+1}. \tag{39}$$

Once the liquid phase saturation is modified, one can improve the gas mole fraction in the liquid phase, using the following relation

$$x_g^{n+1}\Big|_{improved} = x_g^{n+1} + \frac{\rho_g^{n+1} s_g^{n+1}}{\rho_o^{n+1} s_o^{n+1}}. \tag{40}$$

Finally, the new gas saturation is set to zero.

Excessive dissolved gas in the liquid phase: This case can occur when the calculated x_g^{n+1} for a computational cell exceeds the gas mole fraction in the liquid phase obtained using the gas pressure and the given PVT data $x_g(p_g^{n+1})$. In fact, in such a case, free gas should be produced. Thus, one needs to correct the gas mole fraction and recalculate the liquid and gas saturations for the cell. To do this, one can set $x_g^{n+1} = x_g(p_g^{n+1})$ and solve equations (49) and (50) to obtain modified values for S_o^{n+1} and S_g^{n+1}, respectively.

Well model

In reservoir simulation, a well model acts as either a boundary condition for the problem or a source term in the governing equations. In general, wells can be of different types; vertical, horizontal, multi-segment, etc. (Chen and Zhang 2009, Wolfsteiner et al. 2003). Traditionally, well boundary condition is applied using a well model. It is therefore extremely important to use an accurate model for the wells and also apply a careful numerical procedure for them. In this paper, only a vertical well model is considered. Using a well model, one can correlate the rate of fluid flow to the difference between the well cell pressure and the bottom hole pressure of the well (Coats et al. 1998). In a three-phase problem, this reads

$$q_o = W_I\, \rho_o x_o \lambda_o \left[p_{bh} - p_o - \rho_o g(z_{bh} - z)\right], \tag{41}$$

$$q_g = W_I\, \left(\rho_o x_g \lambda_o + \rho_g \lambda_g\right) \left[p_{bh} - p_g - \rho_g g(z_{bh} - z)\right], \tag{42}$$

$$q_w = W_I\, \rho_w \lambda_w \left[p_{bh} - p_w - \rho_w g(z_{bh} - z)\right], \tag{43}$$

where W_I represents the so-called well index. For a well placed in a cell with anisotropic permeability field $\mathbf{K} = diag(K_{xx}, K_{yy}, K_{zz})$, one can write (Chen and Zhang 2009)

$$W_I = \frac{2\pi\Delta z\sqrt{K_{xx}K_{yy}}}{\ln(\frac{r_e}{r_w}) + s}, \tag{44}$$

in which r_w and s are the well-bore radius and the well skin factor, respectively. In addition, r_e represents an equivalent radius given by (Wolfsteiner et al. 2003)

$$r_e = \frac{0.28\sqrt{(\Delta x)^2(K_{yy}/K_{xx})^{0.5} + (\Delta y)^2(K_{xx}/K_{yy})^{0.5}}}{(K_{yy}/K_{xx})^{0.25} + (K_{xx}/K_{yy})^{0.25}}.$$

In general, two types of injection wells, are normally used in simulations; Gas injection or Water injection. In both cases, the rate of injections are known functions of time. Here, the injection wells operate with constant flow rates. Thus, either q_g or q_w is used to determine the bottom hole pressure of the well.

In this work, a production well can operate under two different scenarios; constant oil production rate and constant bottom hole pressure. For a well operating under given oil production rate q_o, one can employ Eq. (41) and use the current cell pressure to determine p_{bh}. The calculated value for the bottom pressure is then used to calculate the gas and water flow rates, using Eqs. (42) and (43). In the case of constant bottom hole pressure, the difference between the cell pressure and the bottom hole pressure is used to determine the production rates. Note that each well adds one extra unknown (flow rate of a phase or well bottom hole pressure) to the system of equations (Wolfsteiner et al. 2003). Thus, unknowns in the well model equations should be solved simultaneously with other unknowns in the pressure equation (Chen et al. 2006). This needs to be taken care of by the computer program.

Adaptive Time Stepping and Linear Solver Options

Adaptive time stepping is a key technique to increase the speed of simulations and can improve the convergence rate of the nonlinear iteration method. An appropriate time step can be estimated using the following criteria:

1. The time step should be between the given minimum and maximum time steps, i.e. $\Delta t_{min} \leq \Delta t \leq \Delta t_{max}$ (Li et al. 2004). The values of Δt_{min} and Δt_{max} are usually set by practical considerations.

2. The time step should be selected in such a way that both the gas pressure and phase saturation increments, in a single time step, be less than given maximum allowable values (Li et al. 2004). At the end of each time step one needs to check the results to ensure this. When this condition is violated, a smaller time step has to be used and the computations must be repeated starting from the previous time level (the recalculation loop in Fig. 3a).

3. If the iterative pressure solver converged in the previous time step within a few iterations and without decreasing time step, the current time step can be increased using a predefined increasing factor. In this paper, the range of increasing factor is chosen between 1.005 and 1.01. Note that the new time step should also satisfy the other controlling conditions such as the maximum allowable phase saturation change in a single time step and the stability criterion of the IMPES method.

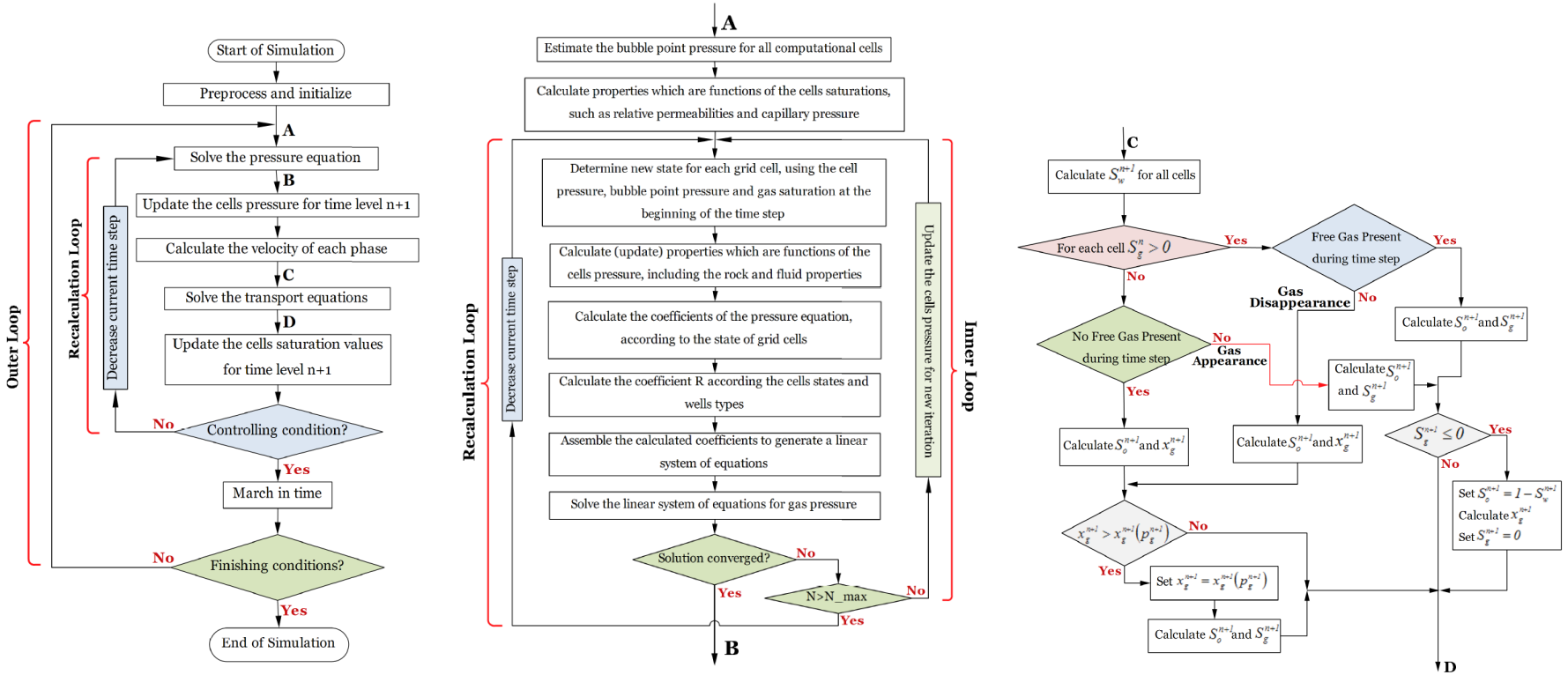
4. If the pressure solver iterations does not converge for the time level $n+1$, the current time step is decreased by a factor of 0.9. Then, the pressure equations are solved again, using information at time level n. This time step correction procedure needs to be continued until the pressure solver converges. This procedure, in fact, is the recalculation loop in the pressure solver, as shown in Fig. 3b). For the results given in this paper, it is shown that number of correction (recalculation) steps is relatively low (less than 5). This technique fairly improves the speed of simulations due to using the maximum possible time step value in each time level.

The iterative procedure used for solving pressure equations is continued until either the maximum number of iterations is reached or the maximum increment in gas pressure and gas mole fraction becomes less than predefined values (here, $1psi$ for pressure and 0.005 for gas mole fraction). For the iterative linear solver used in this work, the iterations are terminated when either the desired residual is achieved or the maximum number of iterations is reached.

Structure of Computer Program

An Object Oriented C++ computer software was developed based on the aforementioned algorithm. The computational engine of this software has two modules; a preprocessor and a simulator. The preprocessor module reads all required data including computational grid, rock and fluid properties, well data, and initial and boundary conditions, and then generates a data structure based on *CELLS* and *FACES* as its primary elements.

At each time step, the gas pressure is first computed implicitly followed by computing velocity distribution in the flow field. The following iterative procedure is used for this purpose (Fig. 3):



(a) General algorithm. (b) The pressure solver algorithm. (c) The saturation solver algorithm.

Figure 3: Computational algorithm for the iterative IMPES method.

1. Loop over all computational cells using current values of variables, estimate bubble point pressure for each cell, using Eq. (38) and the given PVT data.

2. For all computational cells, update the physical properties, such as capillary pressure and relative permeabilities, which are functions of saturations.

3. Determine the state of each computational cell, using the following pseudocode:

   if $S_g^n > 0$ **then**
   Cell state = FREE GAS PRESENT.
   if $p_g^l >= p_b^n$ **then**
   Cell state = GAS DISAPPEARANCE.
   end if
   else
   Cell state = NO FREE GAS PRESENT.
   if $p_g^l < p_b^n$ **then**
   Cell state = GAS APPEARANCE.
   end if
   end if

4. Update the rock and fluid properties, which are functions of gas pressure, using the given PVT data and available physical models.

5. If necessary, injection and production well conditions are updated.

6. Compute parameters v_1, v_2, C_{13}, C_{23}, C_{33}, a_1, a_2, a_3, R_o, R_g, and R according to Table 1 and Eq. (27), respectively.

7. Generate a linear system of equations from the linearised pressure equation, using Eqs. (30).

8. Solve the linear system of equations for p_g to reach a prescribed accuracy. An efficient sparse linear solver based on the Bi-conjugate Gradient Stabilized method is used for this purpose. This solver employs ILU_0 as a pre-conditioner.

9. Using the newly computed pressures, p_g, return to the step 3 and repeat this process until δp_g becomes smaller than a given tolerance, say, 1 (psi).

10. Using the converged gas pressure field, p_g, and Eqs. (4) to (7), compute phase velocities on faces of all computational cells.

Steps 3 to 9 describe the pressure solver module of the computational algorithm. After the velocity field was determined, the mass conservation equations can be solved to update phase saturations. This is achieved by the following procedure, which is explicit in time:

1. Using the computed aqua phase velocities in Eq. (6), determine saturations of water for all computational cells by

$$\frac{\overline{\delta}[V\phi\rho_w S_w]}{\Delta t} = -\,\mathrm{J}\left(\mathbf{u}_w, \rho_w\right) - q_w, \tag{45}$$

This leads to

$$S_w^{n+1} = \frac{1}{(\phi\rho_w)^{n+1}}\left[(\phi\rho_w S_w)^n - \frac{\Delta t}{V}\left(\mathrm{J}(\mathbf{u}_w, \rho_w) + q_w\right)\right]. \tag{46}$$

2. Using the computed phase velocities from Eqs. (4) and (5), determine oil saturation along with either gas saturation (saturated reservoir) or dissolved gas (under-saturated reservoir) using the following equations

$$\frac{\overline{\delta}[V\phi\rho_o S_o x_o]}{\Delta t} = -\,\mathrm{J}(\mathbf{u}_o, \rho_o x_o) - q_o - R_o, \tag{47}$$

$$\frac{\overline{\delta}[V\phi(\rho_o S_o x_g + \rho_g S_g)]}{\Delta t} = -\,\mathrm{J}(\mathbf{u}_o, \rho_o x_g) - \mathrm{J}(\mathbf{u}_g, \rho_g) - q_g - R_g. \tag{48}$$

When free gas exists or appears during a time step, since concentration of gas in the liquid phase x_g is a function of pressure, computation of the saturations of water and oil phases is direct. In such cases, saturation of gas phase can be explicitly computed using saturation of the oil phase. Using Eqs. (47) and (48), one can write

$$S_o^{n+1} = \frac{1}{(\phi\rho_o x_o)^{n+1}}\left[(\phi\rho_o S_o x_o)^n - \frac{\Delta t}{V}\Big(\mathrm{J}(\mathbf{u}_o, \rho_o x_o) + q_o\Big) - R_o\right], \tag{49}$$

$$S_g^{n+1} = \frac{1}{(\phi\rho_g)^{n+1}}\left[(\phi\rho_g S_g)^n - \overline{\delta}[\phi\rho_o x_g S_o] - \frac{\Delta t}{V}\left(\mathrm{J}(\mathbf{u}_o, \rho_o x_g) + \mathrm{J}(\mathbf{u}_g, \rho_g) + q_g\right) - R_g\right]. \tag{50}$$

For cells without free gas or cells in which free gas disappears during a time step, however, equations describing oil saturation and x_g are coupled and require special treatment. It is not generally recommended to use constraint $S_o + S_g + S_w = 1$. Instead, saturation of each phase is computed from its associated equation. In this way, although conservation of phase volumes might not be exact but conservation of mass for each component is fully satisfied (Coats et al. 1998). In the case of under-saturated cells the gas saturation is zero and x_g is an unknown. Noting that $x_o + x_g = 1$, one can remove x_o^{n+1} and x_g^{n+1} from Eqs. (47) and (48) by adding these two equations and determining S_o^{n+1}. This gives

$$S_o^{n+1} = \frac{1}{(\phi\rho_o)^{n+1}}\left[(\phi\rho_o S_o)^n - \frac{\Delta t}{V}\left(\mathrm{J}(\mathbf{u}_o, \rho_o) + \mathrm{J}(\mathbf{u}_g, \rho_g) + q_o + q_g\right) - R_o - R_g\right], \tag{51}$$

$$x_g^{n+1} = \frac{1}{(\phi\rho s_o)^{n+1}}\left[(\phi\rho_o S_o x_g)^n - \frac{\Delta t}{V}\left(\mathrm{J}(\mathbf{u}_o, \rho_o x_g) + \mathrm{J}(\mathbf{u}_g, \rho_g) + q_g\right) - R_g\right]. \tag{52}$$

In the above equations, the convective flux J is computed for all cells using a first-order upwind approximation. Also, the phase densities, and viscosities are computed based on the cell condition.

At the end of each time step, all computed variables are saved in files. The above procedure is repeated until either a certain time is passed or a finishing condition (e.g. minimum oil production rate or maximum gas-oil ratio) is satisfied (Fig. 3a).

Benchmark Problems

In this section, the first (Odeh 1981) and the eighth (Quandalle 1983) SPE comparative study problems are solved to assess the accuracy and computational performance of the proposed simulation algorithm. These problems are solved both on orthogonal and non-orthogonal grids. As a final test case, water flooding in a two-phase system with a fairly general 3D geometry is studied.

First SPE comparative study problem

This is a challenging problem as there is a high rate of gas injection along with a varying Gas Oil Ratio (GOR) in the production well. The geometry of the reservoir and the grid properties are shown in Fig. 4. Top views of grids used for this study are shown in Fig. 5. These include one orthogonal and two non-orthogonal grids. The reservoir consists of three layers with the same porosity ($\phi = 0.3$ at 14.7 psi with rock compressibility 3×10^{-6}) but different thicknesses showing vertical heterogeneity. The reservoir initially contains oil $S_o = 0.88$ and water $S_w = 0.12$. The reservoir pressure is $p_g = 4800psi$, with an initial bubble point pressure $p_b = 4014.7psi$, therefore the reservoir is initially under-saturated. While the gravity effect is considered in this problem, the effect of capillary pressure is neglected. Gas is injected into the reservoir through a well positioned at the upper corner cell $(1, 1, 1)$ and oil is produced from a well placed at the lower corner cell $(10, 10, 3)$, as shown in Fig. 4.

As long as the bottom hole pressure for the production well is above $1000psi$, the oil production rate is kept constant, $q_o = 20000\frac{STB}{Day}$. When the well bottom hole pressure drops below $1000psi$, its value is set to $1000psi$. The physical properties of phases for both saturated and under-saturated states and the relative permeability data are given in Tables 2 and 3, respectively. This problem is solved under two different conditions:

1. The bubble point pressure remains constant equal to its initial value. In this case, no more free gas dissolves in the solution above the initial bubble point (Case A).

2. The bubble point pressure varies with gas saturation. Above the calculated bubble point pressures, the PVT lines are parallel to the original line. In this case, depending on the cell pressure value, some or all of free gas can dissolve in the liquid phase (Case B).

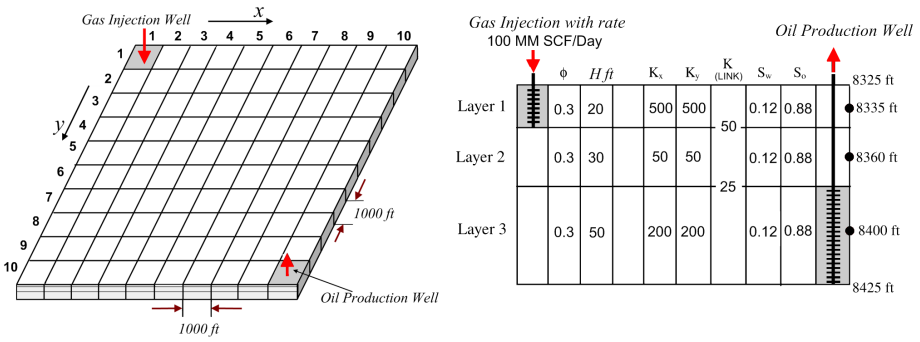


Figure 4: The reservoir and grid system (left) together with its diagonal cross section (right).

Table 2: The physical properties of different phases for both saturated and under-saturated states.

State	$p_g\,(psia)$	$\mu_o\,(cp)$	$\rho_o\,(\frac{lbm}{ft^3})$	$Rs_o\,(\frac{SCF}{STB})$	$\mu_g\,(cp)$	$\rho_o\,(\frac{lbm}{ft^3})$	$\mu_w\,(cp)$	$\rho_w\,(\frac{lbm}{ft^3})$
Saturated	14.7	1.0400	46.244	1.0	0.008	0.0647	0.31	62.238
	264.7	0.9750	43.544	90.5	0.0096	0.8916	0.31	62.283
	514.7	0.9100	42.287	180.0	0.0112	1.7185	0.31	62.328
	1014.7	0.8300	41.004	371.0	0.014	3.3727	0.31	62.418
	2014.7	0.6950	38.995	636.0	0.0189	6.6806	0.31	62.599
	2514.7	0.6410	38.304	775.0	0.0208	8.3326	0.31	62.690
	3014.7	0.5940	37.781	930.0	0.0228	9.9837	0.31	62.781
	4014.7	0.5100	37.046	1270.0	0.0268	13.2952	0.31	62.964
	5014.7	0.4490	36.424	1618.0	0.0309	16.6139	0.31	63.160
	9014.7	0.2030	34.482	2984.0	0.047	27.9483	0.31	63.959
Under-saturated	4014.7	0.5100	37.046	1270.0	–	–	0.31	62.964
	9014.7	0.7400	39.768	1270.0	–	–	0.31	63.959

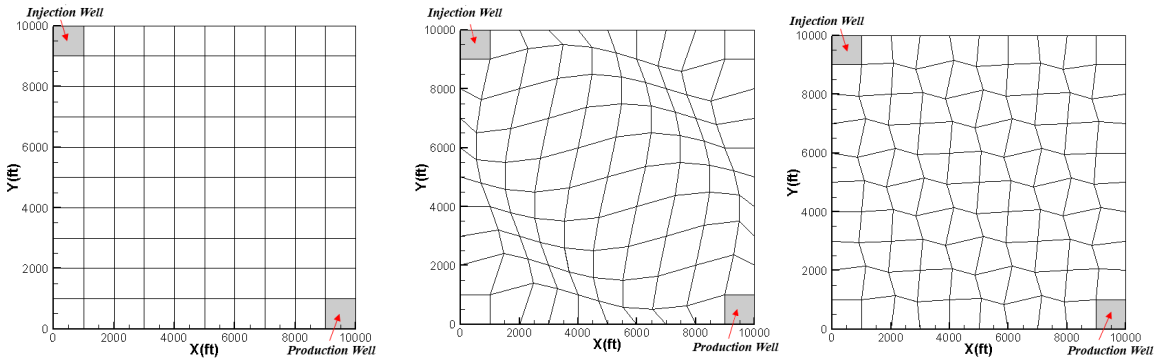
From Fig. 4, it can be seen that the gas injection well is located on the top layer of the reservoir with the smallest thickness (cell volume) and the highest absolute permeability. Therefore, one can expect that a large amount of free gas will transfer toward the production well through this layer. In addition, the buoyancy effects force the free gas (if available) to move from the lower layers towards the top layer.

In case A, as long as the cell pressure is above the initial bubble point, all of the injected gas goes toward the production well. This process continues until the reservoir pressure drops below the bubble point pressure. At this time, gas can be released from the solution. In case B, depending on the pressure of the well cell, some of the injected gas can dissolve into the solution. Compared with case A, in general, it is expected that lower amounts of free gas goes towards the production well.

Case A: For this case, the initial time step is set to 1.0 *Day* and the time step increasing and decreasing factors are set to 1.01 and 0.8, respectively. The variations of the gas saturation and the gas pressure versus time for cells containing wells are plotted in Figs. 6a to 6d. In addition, for the cell containing the production well, the GOR and oil production rate are shown in Figs. 6e and 6f, respectively. These curves are plotted using symbols to show the variations in the time step size during the simulation. The results are in good agreement with those shown in (Odeh 1981). Considering Figs. 6b and 6d, it can be seen the gas pressure at the production

Table 3: The relative permeability data for the first SPE comparative study problem.

s_g	k_{rg}	k_{ro}	k_{rw}
0	0.0	1.0	0.0
0	0.0	1.0	0.0
0.02	0.0	0.997	0.0
0.05	0.005	0.980	0.0
0.12	0.025	0.700	0.0
0.2	0.075	0.350	0.0
0.25	0.125	0.200	0.0
0.3	0.19	0.090	0.0
0.4	0.41	0.021	0.0
0.45	0.60	0.010	0.0
0.5	0.72	0.001	0.0
0.6	0.87	0.0001	0.0
0.7	0.94	0.000	0.0
0.85	0.98	0.000	0.0
1.0	1.0	0.000	0.0



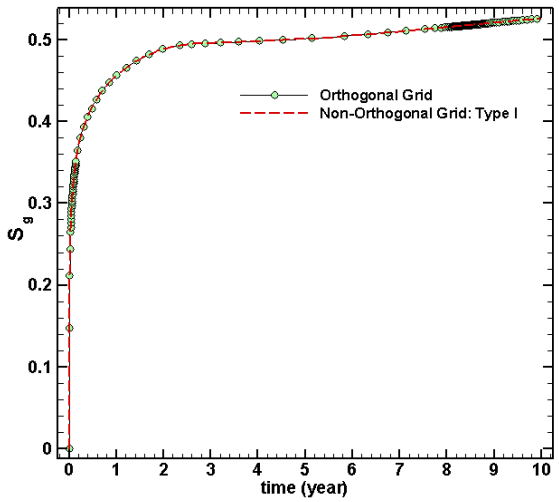
(a) Original orthogonal grid. (b) Non-orthogonal grid: Type I (c) Non-orthogonal grid: Type II

Figure 5: Top view of different grids used for solving the first SPE comparative study problem.

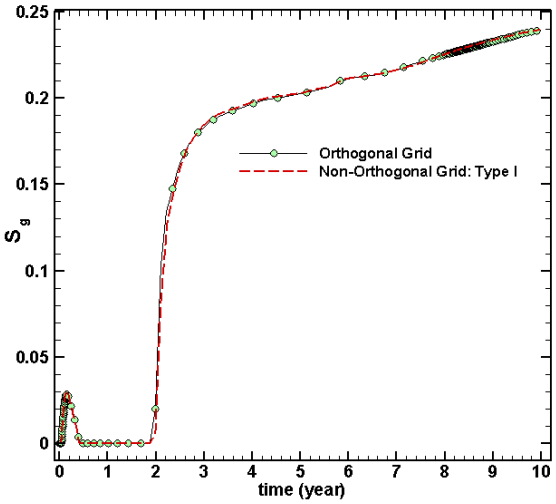
well cell within the first year of production drops below the initial bubble point and the cell status changes to the saturated state. In such a condition, the solution liberates some free gas and the saturation pressure is equal to the gas pressure. In addition, it is clear that the front of gas saturation reaches the production well cell at about 2 years. At $t = 3$ years, the production scenario changes to a constant bottom hole pressure condition and the production rate is decreased to values below $q_o = 6000\text{STB/Day}$ at the end of simulation time.

Figure 7 shows the gas saturation contour plots at time $t = 1$ year for two different grids. Although the gas saturation distributions for these grids are somehow different, the results shown in Fig. 6 indicate that different parameters associated with wells are nearly identical. This means that the mass balance is satisfied for both of the computational grids.

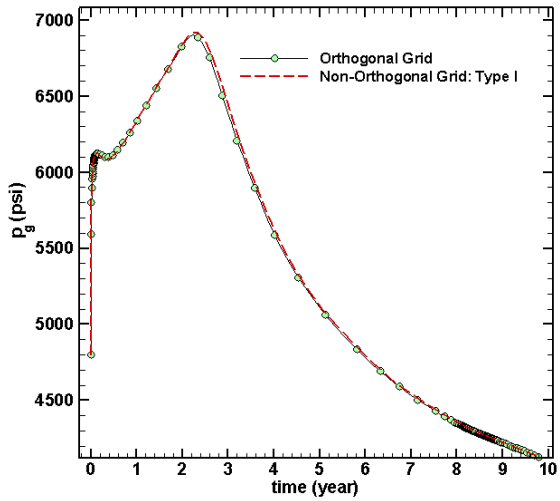
Case B: For this case, the initial time step is set to 1.1 *Day* and the time step increasing and decreasing factors are set to 1.008 and 0.80, respectively. The variations of the gas saturation, cell pressure, oil production rate and the GOR versus time for well cells are shown in Figs. 8a to 8f. Again, the results are shown by symbols in order to show the variations in the time step size during the simulation. The numerical results are in good agreements with those reported



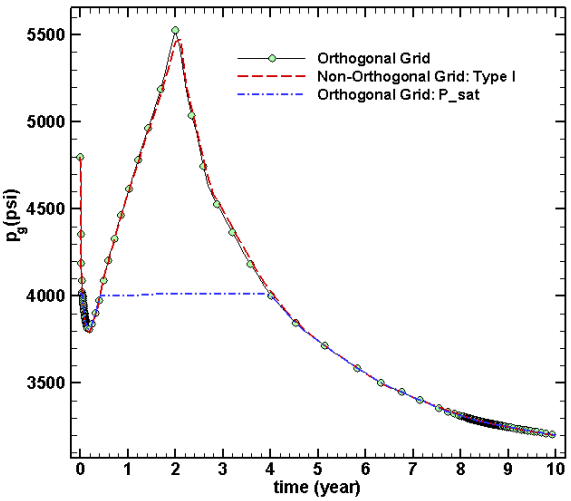
(a) Gas saturation at cell (1,1,1).



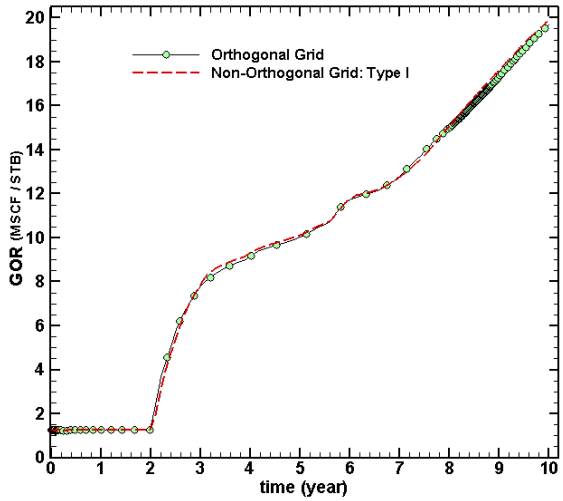
(b) Gas saturation at cell (10,10,1).



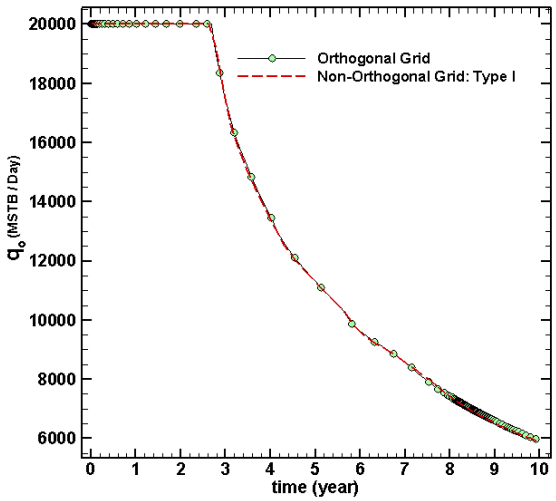
(c) Gas pressure and the saturation pressure at cell (1,1,1).



(d) Gas pressure and the saturation pressure at cell (10,10,1).

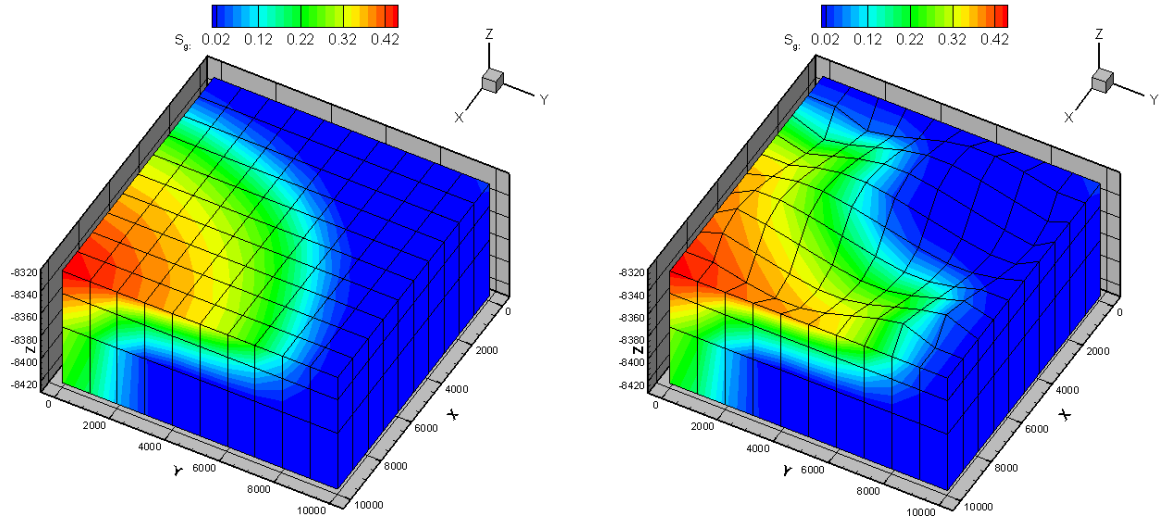


(e) GOR.



(f) Oil production rate.

Figure 6: Variation of different parameters versus time in case A of the first comparative SPE problem.



(a) Original orthogonal grid. (b) Non-orthogonal grid: Type I

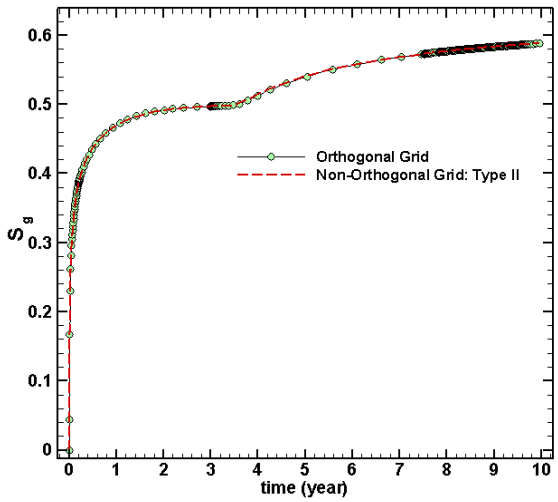
Figure 7: Gas saturation contour plots for case A of SPE 1 comparative study problem for different grids at $t = 1$ (year).

in (Odeh 1981, Coats et al. 1998, Li et al. 2004).

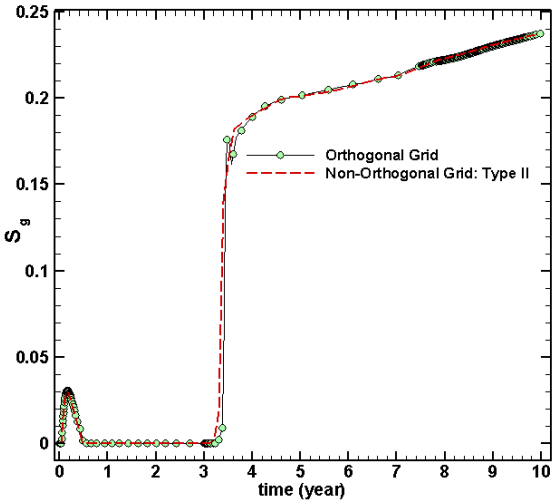
During the first year of production, Fig. 8d shows that the gas pressure at cell (10,10,3) drops below the initial bubble point and the solution liberates some gas, as shown in Fig. 8b. In this situation, the production cell is saturated and the saturation pressure is equal to the gas pressure (Fig. 8d). After this period, the production cell pressure increases and crosses the bubble point pressure. Therefore, the production cell status changes to an under-saturated state. The production cell pressure continues to increase to reach a peak about $5900psi$. For times about third year, the front of injected gas reaches the production wells and the state of the well cell changes to saturated. At this time, the pressure equations becomes highly nonlinear due to entrance of the injected gas into under-saturated cells. Therefore, the simulator decreases the time steps to reach a converged solution. Due to the existence of free gas in cell (10,10,3), its saturation pressure should be equal to the cell pressure; Therefore, the saturation pressure experiences a big jump (about 1800 psi) to reach the cell pressure. For the rest of simulation, the saturation pressure remains equal to the gas pressure, as shown in Fig. 8d. Within the first four years of the simulation, the production scenario is a constant oil production rate and after that the production well operates with constant bottom hole pressure.

Eighth SPE comparative study problem

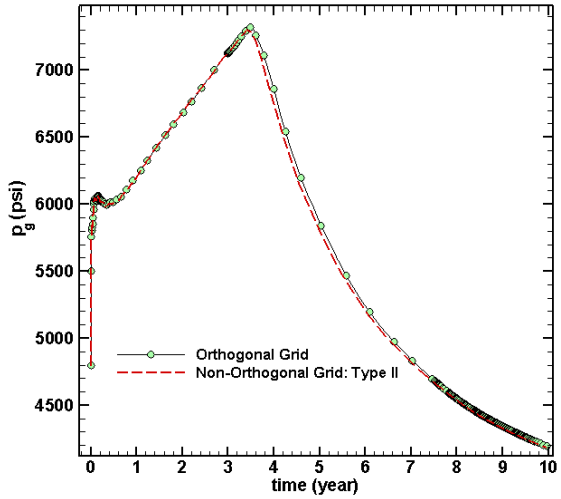
As the second benchmark problem, the eighth SPE comparative study problem (Quandalle 1983) is investigated. While the geometry and original grid of this problem is similar to those of the first SPE problem (Fig. 4), the reservoir introduced in the SPE8 problem has 4 layers. All PVT and rock data of this problem are given in (Quandalle 1983). In this problem, gas is



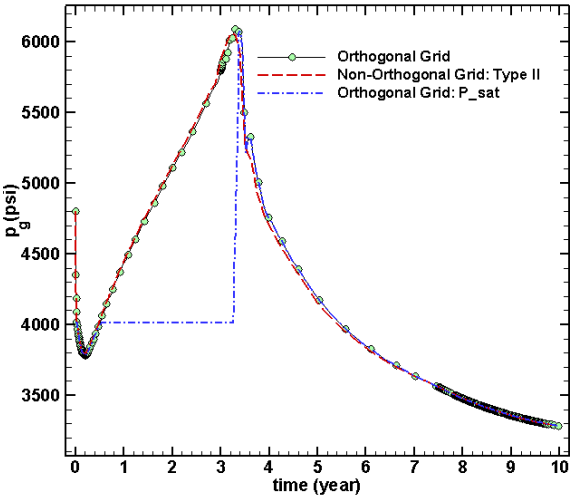
(a) Gas saturation at cell (1,1,1).



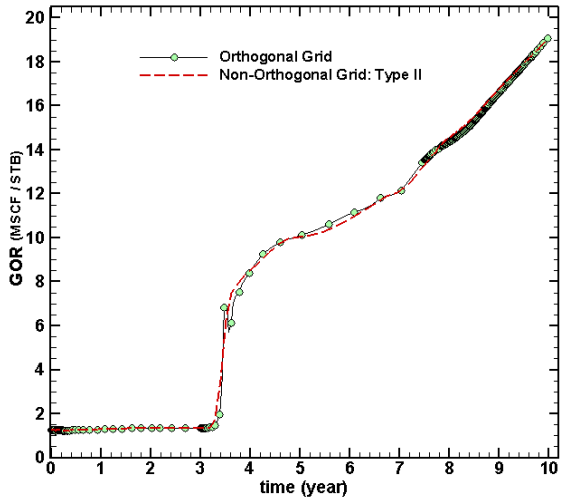
(b) Gas saturation at cell (10,10,1).



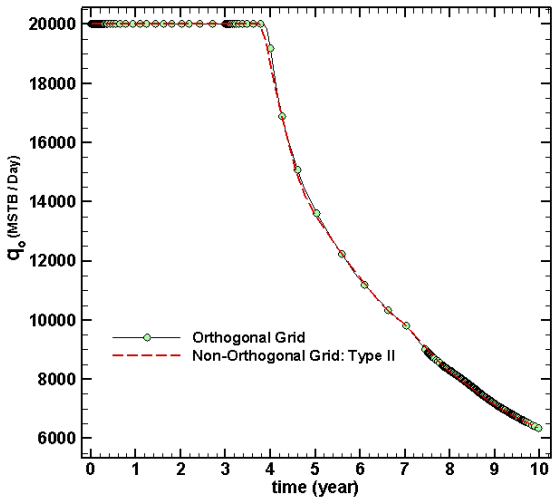
(c) Gas pressure and saturation pressure at cell (1,1,1).



(d) Gas pressure and the saturation pressure at cell (10,10,1).



(e) GOR.

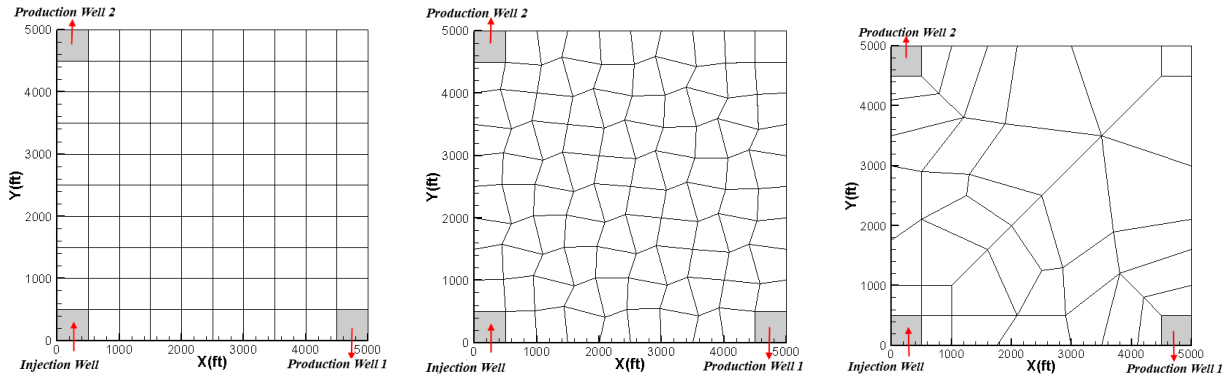


(f) Oil production rate.

Figure 8: Variation of different parameters versus time in case B of the first comparative SPE problem.

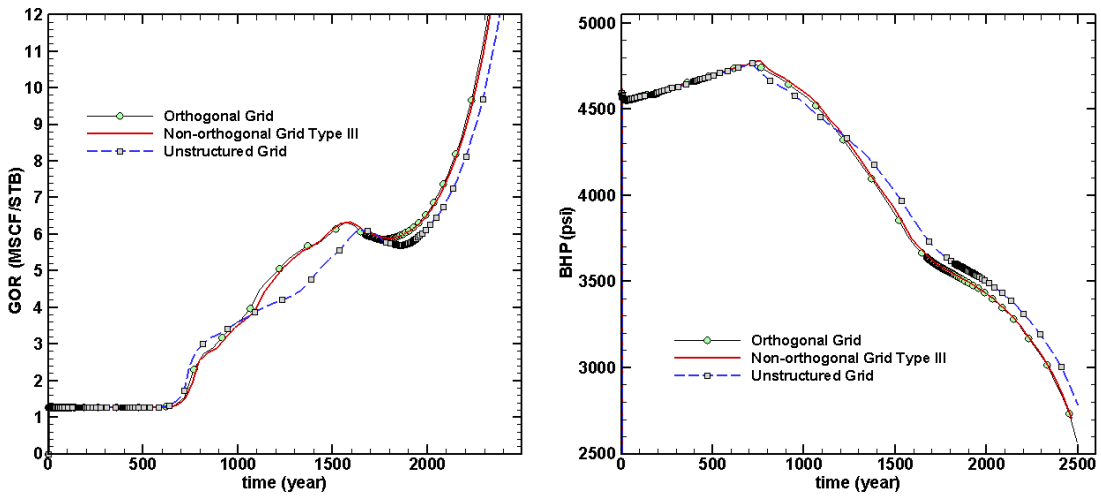
injected into the reservoir from a well and oil is produced by two production wells. Here, three different computational grids including orthogonal, non-orthogonal and unstructured grids are used. These computational grids along with injection/production well positions are shown in Fig. 9.

In (Quandalle 1983), it was shown that using different grids, in general, leads to relatively different results. Here, the variation of GOR and also bottom hole pressure of the producer 1 are shown in Figs. 10a and 10b for three different grids. The results for the orthogonal grid explicitly show that the proposed numerical algorithm has acceptable accuracy when compared with results given in (Quandalle 1983). While the results associated with the non-orthogonal grid have a good agreement with those of the orthogonal grid, the results for unstructured grid are slightly different (Fig. 10).



(a) Orthogonal grid (b) Non-orthogonal grid: Type III (c) Unstructured Grid

Figure 9: Different grids used for solving the eighth SPE comparative study problem.



(a) GOR. (b) Bottom hole pressure

Figure 10: Variation of GOR and bottom hole pressure of production well 1 versus time for the eighth SPE comparative study problem.

Note that the total number of computational cells in the unstructured gird ($N = 156$) is less than that of the orthogonal (and also non-orthogonal) grids ($N = 400$). Since the number of computational cells is less, it is expected that the computational cost of the simulation with unstructured grid is less than that of the orthogonal grid. To clarify this point, the number of time steps N_{ts}, iterations N_i and the CPU times used to solve the first and eighth SPE comparative study problems are given in Table 4. Here, the total number of iterations, N_{ts}, is calculated by including all recalculation steps due to time step changes. Note that there are a few parameters (e.g. maximum acceptable time step and time step increasing and decreasing factors) that affect the total number of time steps and also the CPU time of the simulation. Obviously, for each benchmark problem, certain choices of these parameters will lead to an optimum CPU times. Table 4 shows that the average number of iterations per time step is less than six. In addition, it can be concluded that the computational cost of each time step for unstructured and also non-orthogonal grids is generally higher than that of orthogonal grid. Comparing the total CPU time for the SPE8 problem, it is clear that the total CPU time for the unstructured grid is less than that of the original orthogonal grid. This is because the total number of cells in unstructured grid is much less than of number of cells in the orthogonal one.

Table 4: The CPU times and number of time steps used to solved the benchmark problems.

Problem		Grid size	Grid type	Iterations and time steps		Total CPU time (s)
				Total N_{ts}	Total N_i	
SPE 1	Case A	$10 \times 10 \times 3$	Orthogonal	540	3198	12.0
			Non-orthogonal:Type I	670	3658	62.0
			Non-orthogonal:Type II	595	3326	41.0
	Case B	$10 \times 10 \times 3$	Orthogonal	1308	6341	29.0
			Non-orthogonal:Type I	25007	6702	342.0
			Non-orthogonal:Type II	3897	15004	193.0
SPE 8		$10 \times 10 \times 4$	Orthogonal	3711	14400	98.0
			Non-orthogonal:Type III	2390	9718	164.0
		39×4	Unstructured	1407	5672	68.0

A two phase problem with complex geometry

In this section, a two phase (water-oil) problem is solved to show the ability of the finite volume method to handle problems with complex geometry and irregular cells. The geometry of the reservoir together with the grid used for simulation are shown in Fig. 11. All PVT and rock data except porosity and absolute permeability are the same as the SPE1 problem. Here, the rock porosity and absolute permeability are chosen as $\phi = 0.15$ (at $14.7psi$) and $K = 200md$ in all x, y, and z directions. It is assumed that the relative permeability values are given by $k_{rw} = S_w$ and $k_{ro} = 1.0 - S_w$. While the gravity effect is considered in this problem, the effect of capillary pressure is neglected.

The reservoir is initially filled with dead oil (without dissolved gas) and its initial pressure is $p_g = 4800psi$. Water is injected into the reservoir through a well positioned at a corner of the reservoir and dead oil is produced from a well placed at the opposite corner, as shown in Fig. 11.

While the injection well injects water with a constant rate of $q_w = 1000 \frac{STB}{Day}$, the production well operates with a constant bottom hole pressure of $4800 psi$.

Figure 12 shows the water saturation contour plot at time $t = 80\, years$ together with oil and water production rates. As can be seen, the oil production rate at initial time is equal to zero and increases sharply with time to reach a constant rate. When water front reaches the cell containing the production well, the oil production rate decreases and the water production rate increases.

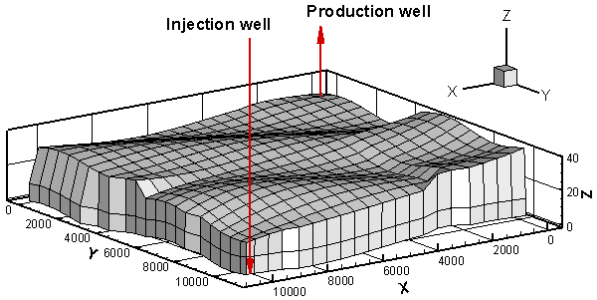
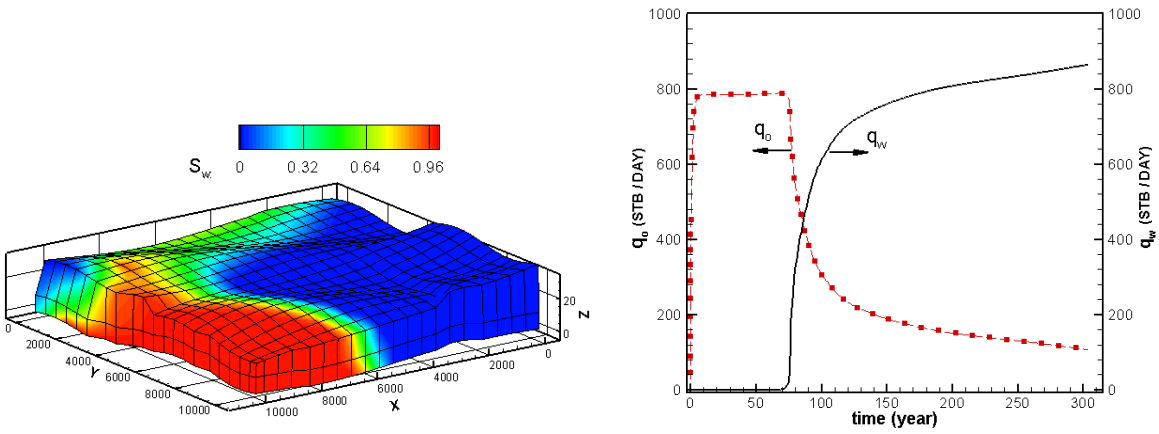


Figure 11: A reservoir with complex geometry and a computational grid with irregular cells.



(a) Water saturation contour plot at $t = 80$(years).

(b) Oil and water production rates.

Figure 12: Numerical results for the problem with complex geometry.

Conclusions

In this paper, a black oil model was used along with a general finite-volume based IMPES algorithm for the simulation of multiphase flows in hydrocarbon reservoirs and tested against a benchmark 3-D problem. This work showed that there are a number of key issues which need to be taken care of properly. Any simulation procedure without such considerations may fail as underlying physical phenomena might be overseen. In particular, the appropriate treatment of accumulation terms with regards to the cell status plays a crucial role and can significantly

decrease the non-linearity of equations. In addition, special treatments were used for cases with negative gas saturation and excessive gas mole fraction in liquid phase to achieve physically meaningful results. It was observed that these numerical treatments can improve the convergence rate of the proposed method.

This study also showed that a major challenge in solving practical reservoir engineering problems originates from the way injection and production wells are treated. It was shown that the proposed algorithm works well for complex situations such as high gas injection rates, and sudden free gas appearance and disappearance in a cell.

The use of an adaptive time integration scheme also improved the computational performance of the proposed algorithm. Here, a time step control factor was used during all simulations which proved to be useful to reach an stable and physically meaningful solution. When the pressure equation did not converge or converged slowly, the time step was modified using a control factor smaller than one (typically between 0.8 and 0.9). In some situations, this modification was repeated at most five times to reach a converged solution. To improve the computational cost of the solution algorithm, a time step control factor larger than one (typically between 1.001 and 1.1) was used whenever the pressure equation converged within a few time steps.

Finally, the numerical results obtained in this work showed that the use of unstructured grids in a finite volume framework greatly enhances the computational capabilities of the reservoir simulator, by allowing more flexibility for complex geometries and requiring much less computational cells, while acceptable accuracy can be achieved.

Nomenclature

References

Aziz, K. and Settari, A., (1979), "Petroleum Reservoir Simulation", *Applied Science*, London.

Almehaideb, R., Aziz, K., (1989). "A reservoir/wellbore model for multiphase injection and pressure transient analysis". *Middle East Oil Show.*

Bergamaschi, L., Mantica, S., Manzini, G., (1999). "A mixed finite element-finite volume formulation of the black-oil model". *SIAM Journal on Scientific Computing* vol. 20 no. 3, pp. 970–997.

Chavent, G., Jaffré, J., (1986). "Mathematical models and finite elements for reservoir simulation: single phase, multiphase, and multicomponent flows through porous media". *Elsevier*, Holland.

Chen, Z., (2000). "Formulations and numerical methods of the black oil model in porous media". *SIAM Journal on Numerical Analysis*, vol. 38 no 2, pp. 489–514.

Chen, Z., Huan, G., Ma, Y., (2006). "Computational methods for multiphase flows in porous media". *Society for Industrial and Applied Mathematics.*

Table 5: Nomenclature of the formulation

Variable	Description	Unit
A	Cross Sectional Area	ft^2
$\mathbf{d}$	Vector connecting two points	–
K	Absolute Permeability	Mili-Darcy (md)
k_{r_α}	Relative Permeability	–
N	Number of computational cells	–
N_i	Number of iterations	–
N_{ts}	Number of time steps	–
p_{bh}	Bottom Hole Pressure	psi
p_{cgo}	Oil-Gas Capillary Pressure	psi
p_{cow}	Oil-Water Capillary Pressure	psi
p_g	Gas Phase Pressure	psi
q_o	Production Rate of Oil Component	$mole/Day$
q_g	Production Rate of Gas Component	$mole/Day$
q_w	Production Rate of Water Component	$mole/Day$
$\mathbf{r}$	Vector connecting two points	–
R	Total residual term	$mole/Day$
R_o	Oil Residual term	$mole/Day$
R_g	Gas Residual term	$mole/Day$
S_α	Saturation of Phase α	–
t	Time	Day
$\mathbf{u}_\alpha$	phase velocity	$mD.psi/cp$
V	Volume of Computational Cell	Reservoir Barrel (RB)
W_I	Well Index	$\frac{RB.cp}{Psi.Day.md}$
x_g	Gas Mole Fraction in Liquid Phase	–
x_o	Oil Mole Fraction in Liquid Phase	–
Greeks		
Δt	Time-step size	Day
∇Z	Depth Variation	ft
ρ_α	Molar Density of Phase α	$moles/RB$
γ	Specific Density of Fluid	psi/ft
μ_α	Viscosity of Phase α	$Centi - Poise(cp)$
$\lambda_\alpha = \mathbf{K}k_{r_\alpha}/\mu_\alpha$	Mobility of Phase α	md/cp
Subscripts		
b	Bubble point	
f	Face of computational cell	
g	Gaseous phase	
o	Oil (liquid) phase	
w	Water (aqua) phase	
Superscripts		
l	Iteration level	
n	Time level	

Chen, Z., Zhang, Y., (2009). "Well flow models for various numerical methods". *International Journal of Numerical Analysis & Modeling.* vol. 6 no. 3, pp. 375–388.

Coats, K., (1989). "Implicit compositional simulation of single-porosity and dual-porosity reservoirs". In: *SPE Symposium on Reservoir Simulation.*

Coats, K., Thomas, L., Pierson, R., (1998). "Compositional and black oil reservoir simulation". *SPE Reservoir Evaluation & Engineering.* vol. 1 no. 4, pp. 372–379.

Edwards, M. G., (2002). "Unstructured, Control-Volume Distributed, Full-Tensor Finite Volume Schemes With Flow Based Grids",*Computational Geosciences.* vol. 6, no. 3-4 , pp. 433–452.

Edwards, M. G., (2008). "A quasi-positive family of continuous Darcy-flux finite-volume schemes with full pressure support", *Journal of Computational Physics.* vol. 227, no. 22 , pp. 9333–9364.

Edwards, M. G., Rogers C. F., (1998). "Finite volume discretization with imposed flux continuity for the general tensor pressure equation", *Computational Geosciences*. vol. 2, no. 1 , pp. 259–290.

Eymard, R., Gallouët, T., Herbin, R., (2000). "Finite volume methods". *Handbook of numerical analysis*. vol. 7, pp. 713–1018.

Farnstrom, K., Ertekin, T., (1987). "A versatile, fully implicit, black oil simulator with variable bubble-point option". In: *SPE California Regional Meeting*.

Greenshields, C. J., Weller, H. G., Gasparini, L., Reese, J. M., (2010). "Implementation of semi-discrete, non-staggered central schemes in a collocated, polyhedral, finite volume framework, for high-speed viscous flows".*International Journal for Numerical Methods in Fluids*. vol. 63 no. 1 , pp. 1–21.

Hirsch, C., (2005). "Development and Application of a Finite Volume Method for the Computation of Flows around Moving Bodies on Unstructured, Overlapping Grids". ,*Technische Universitat Hamburg-Harburg*,

Hirsch, C., (2007). "Numerical computation of internal and external flows: fundamentals of computational fluid dynamics". vol. 1,*Butterworth-Heinemann*,

Lee, S., Wolfsteiner, C., Tchelepi, H., (2008). "Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity".*Computational Geosciences*. vol. 12 no. 3 , pp. 351–366.

LeVeque, R., (2002). "Finite volume methods for hyperbolic problems". vol. 31. *Cambridge Univ. Press*.

Li, B., Chen, Z., Huan, G., (2004). "Control volume function approximation methods and their applications to modeling porous media flow ii: the black oil model." *Advances in water resources*. vol. 27 no. 2, pp. 99–120.

Odeh, A., (1981). "Comparison of solutions to a three-dimensional black-oil reservoir simulation problem". *Journal of Petroleum Technology*. vol. 33 no. 1, pp. 13–25.

Pao, W., Lewis, R., (2002). "Three-dimensional finite element simulation of three-phase flow in a deforming fissured reservoir." *Computer methods in applied mechanics and engineering*. vol. 191 no. 23-24, pp. 2631–2659.

Pao, W., Lewis, R., Masters, I., (2001). "A fully coupled hydrothermo-poro-mechanical model for black oil reservoir simulation." *International journal for numerical and analytical methods in geomechanics*. vol. 25, no. 12, pp. 1229–1256.

Peric, M., (2004). "Numerical methods for computing turbulent flows, in: Introduction to turbulence modelling V." Technical report.

Qin, L., Malgorzata, P., Wheeler, M. F., (2002). "A parallel multiblock black-oil model in multimodel implementation". *SPE Journal*. vol 7 no. 3, pp. 278–287.

Quandalle, P., (1983). "Eighth SPE comparative solution project: Gridding techniques in reservoir simulation." *Journal of Petroleum Technology*. vol. 7 no. 1, pp. 343–357.

Salama, A., Sun, S., and Wheeler, M. f., (2014). "Solving global problem by considering multitude if local problems: application to fluid flow in anisotropic porous media using the multipoint flux approximation." *Journal of computational and applied mathematics*. vol 267 no. 1, pp. 117–130.

Schiozer, D. J., (1994). "Simultaneous simulation of reservoir and surface facilities." PhD thesis, Standford University, Department of petroleum engineering.

Thomas, S. G., Wheeler, M. F., (2011). "Enhanced velocity mixed finite element methods for modeling coupled flow and transport on non-matching multiblock grids." *Computational Geosciences*. vol. 15 no.4 , pp. 605–625.

Thomas, L., Lumpkin, W., Reheis, G., (1976). "Reservoir simulation of variable bubble-point problems". *Old SPE Journal*. vol. 16 no. 1, pp. 10–16.

Trangenstein, J., Bell, J., (1989). "Mathematical structure of the black-oil model for petroleum reservoir simulation". *SIAM Journal on Applied Mathematics*, vol. 49 no. 3, pp. 749–783.

Verma, S. K., (1996). "Flexible grids for reservoir simulation". PhD thesis, Standford University.

Versteeg, H., Malalasekera, W., (2007). "An introduction to computational fluid dynamics: the finite volume method". *Prentice Hall*.

Wolfsteiner, C., Durlofsky, L., Aziz, K., (2003). "Calculation of well index for nonconventional wells on arbitrary grids". *Computational Geosciences*. vol. 7 no. 1, pp. 61–82.

Wheeler Mary F, Wheeler j. A., Peszynska M., (2000). "A distributed computing portal for coupling multi-physics and multiple domains in porous media". *computational methods in water resources*. vol. 12. pp. 167–174.

Wheleer, M. F., (2002). "Advanced techniques and algorithms for reservoir simulation. II: the multiblock approach in the integrated prarallel accurate reservoir simulator (IPARS)." *Resource recovery, confinement, and remediation of environmental Hazards, Springer*.