Abstract - Most of the partial differential equations involved in reservoir simulation cannot be solved analytically because of their nonlinear nature. Numerical techniques must be used to solve the flow equations. The most popular numerical method in use in the oil industry is the finite difference method. The finite difference method is implemented by superimposing a finite-difference grid over the reservoir to be modeled. The chosen grid system is then used to approximate the spatial derivatives in the continuous equations. These approximations are obtained by truncating the Taylor’s series expansion of the unknown variables as a function of the grid locations. This paper proposes the use of finite difference equations that incorporate the singularities in pressure at the wells. The new finite difference equations represent the actual pressures at the wellbore and elsewhere in the well cells. No well equations are required. The traditional finite difference equations are unable to predict wellbore pressures because they are based on Taylor series, which are polynomial in form. Polynomials are continuous functions and are unable to represent singularities. Instead of polynomials, finite equations are derived using logarithmic (ln(r)) function, it is singular as r approaches zero. In this paper, main emphasis is laid on incorporating the idea of singularity in the finite difference equations. Calculation part is merely stated. The idea mentioned in the paper is used by many authors for various purposes.

1. INTRODUCTION

The equation which govern the flow of fluids in porous media are non-linear partial differential equations which relate the pressure and saturation changes with time throughout the medium. These equations are extremely complex, and their applications are complicated by the presence of specialized boundary conditions. The solutions of these equations by analytical means is generally impossible except for the trivial cases. The solutions, when they do exist, give a continuous distribution of the dependent parameters (pressure or saturation).

The numerical solution of these equations is generally the only way that a solution can be obtained in most applications. The numerical solution produces answers at discrete points within the system. The transformation of the continuous differential equation to a discrete form is made by the use of finite differences. In this process both space and time are discretized.

The solution to the systems of flow equations commonly encountered in reservoir engineering work involves the determination of some dependent parameters in space and time. The solution is obtained at discrete points in space and time. The spatial domain is broken up into a number of cells, grids, or blocks by superimposing some type of grid. This grid is usually rectangular in form but not necessarily so. The time domain is also discretized into a number of time steps, during each of which the problem is solved to obtain new values of the dependent parameter. The size of these steps depends on the particular problem being solved, and generally the smaller the time step the more accurate is the solution.

The numerical solution of partial differential equations by finite differences refers to the process of replacing the partial derivatives by finite difference quotients and then obtaining solutions of the resulting system of algebraic equations.

The partial differential equation is replaced by its finite difference equivalent. The finite difference equations can be derived by making a Taylor series expansion of the function at a given point and then solving for the required derivative.

Consider the following Taylor series expansions:
\[ P(x + \Delta x) = P(x) + \Delta x P'(x) + \frac{1}{2} \Delta x^2 P''(x) + \frac{1}{6} \Delta x^3 P''' + \ldots \]
\[ \ldots \]
\[ P(x - \Delta x) = P(x) - \Delta x P'(x) + \frac{1}{2} \Delta x^2 P''(x) - \frac{1}{6} \Delta x^3 P''' + \ldots \]
\[ \text{where } P' = \frac{\partial P}{\partial x} \text{ and } P'' = \frac{\partial^2 P}{\partial x^2}, \text{ etc.} \]

These equations could be solved for the first or second derivatives as required,
\[ P'(x) = \frac{P(x + \Delta x) - P(x)}{\Delta x} + O(\Delta x) \]
\[ P'(x) = \frac{P(x) - P(x - \Delta x)}{\Delta x} + O(\Delta x). \]
These are the forward and backward differences respectively for the first derivative. A central difference can also be obtained as follows:

\[ \frac{p(x + \Delta x) - p(x - \Delta x)}{2\Delta x} + O(\Delta x^2) \]

The errors associated with these approximations are different, the forward and backward differences have errors of the order of \( \Delta x \), while the error in the central form is of the order of \( \Delta x^2 \). These error associated with the finite difference form of the partial differential equation is called the truncation error.

For second derivative,

\[ P'(x) = \frac{p(x + \Delta x) - 2p(x) + p(x - \Delta x)}{\Delta x^2} + O(\Delta x^2) \]

Solving for \( P''(x) \):

\[ \frac{p(x + \Delta x) - 2p(x) + p(x - \Delta x)}{\Delta x^2} + O(\Delta x^2) \]

Therefore the error associated with the second derivative is of the order of \( \Delta x^2 \).

II. WELL MODELING

Accurate well modeling is very important for flow simulations in reservoir engineering. The key point of well modeling is to perform accurate fluid flow simulations in the near-well region. The computational accuracy of well parameters such as the well flow rate or the wellbore pressure depends greatly on the near-well flow modeling.

The main difficulty of well modeling is the problem of singularity because of the difference in scale between the small wellbore diameter (less than 0.3m) and the large wellblock grid dimensions used in the simulation (from tens to hundreds of meters), and to the radial nature of the flow around the well (i.e. nonlinear but logarithmic variations of the pressure away from the well). Thus the wellblock pressure calculated by standard finite-difference methods is not the wellbore pressure.

The first theoretical study of well equations was given by Peaceman [1] for block centred finite difference methods on square grids for single phase flow. Peaceman’s study gave a proper interpretation of a well-block pressure and indicated how it relates to the flowing bottom hole pressure. Peaceman first demonstrated that a wellblock pressure calculated by finite difference in a uniform grid corresponds to the pressure at an equivalent wellblock radius \( r_0 \) related to gridblock dimensions. Assuming a radial flow around the well, he demonstrated that this radius could be used to relate the wellblock pressure to the wellbore pressure.

The importance of his study is that the computed block pressure is associated with the steady-state pressure for the actual well at an equivalent radius \( r_0 \). For a square grid with grid size \( h \), Peaceman derived a formula for \( r_0 \) by three different approaches: (1) analytically by assuming that the pressure in the blocks adjacent to the well block is computed exactly by the radial flow model, obtaining \( r_0 = 0.208h \), (2) numerically by solving the pressure equation on a sequence of grids, deriving \( r_0 = 0.2h \), and (3) by solving exactly the system of difference equations and using the equation for the pressure drop between the injector and producer in a repeated five-spot pattern problem, finding \( r_0=0.1987h \). From these approaches, he concluded that \( r_0 \approx 0.2h \).

Peaceman’s finite difference well models on square grids have been extended in various directions, including to rectangular grids, anisotropic reservoirs, horizontal wells, and multiphase flows and to incorporating gravity force, skin, and non-Darcy effects. Peaceman himself extended his classical well model to more general scenarios where rectangular grids and anisotropic permeabilities are allowed.

III. WELL-BLOCK EQUATIONS

Peaceman has defined an equivalent well-block radius, \( r_e \), as the radius at which the steady state flowing pressure in the reservoir is equal to the numerically calculated pressure, \( p_o \), of the block containing the well. This definition of \( r_e \) can be used to relate the well pressure, \( p_w \), to the flow rate, \( q \), through \( p_o \):

\[ q = \frac{2\pi kh}{\mu} \left[ p_o - p_w \right] \ln \left( \frac{r_e}{r_w} \right) \]  \hspace{1cm} (1)

Peaceman has obtained an approximate value of \( r_0 \) for an interior well in a uniform square grid by assuming radial steady-state flow between the well block and the block adjacent to this block:

\[ p_i = p_0 + \frac{q}{2\pi kh} \ln \left( \frac{\Delta x}{r_0^2} \right) \]  \hspace{1cm} (2)

where \( i=1, \ 2, \ 3, \ 4 \) for the four surrounding blocks in the five-point finite difference scheme. Combining this equation with the steady-state difference equation for the well block,

\[ \frac{kh}{\mu} (p_1 + p_2 + p_3 + p_4 - 4p_0) = q \]  \hspace{1cm} (3)

Peaceman obtained the value

\[ \frac{r_0}{\Delta x} = \exp \left( -\frac{3}{2} \right) = 0.208 \]  \hspace{1cm} (4)
which is close to the more precise numerically computed value of 0.1982 ($\approx 0.2$). Peaceman obtained this more precise value by the use of the difference in pressure between injection and production wells in repeated five-spot as derived by Muskat, who used potential theory. In a second paper, Peaceman considered the problem of rectangular or nonsquare blocks, and by following the approach used in deriving equation (*), he obtained

\[ \frac{r_0}{\Delta x} = \exp \left( \ln \frac{n - a}{1 + a^2} \right) \]  

(5)

Using the numerical approach along with the analytical solution for the repeated five-spot, Peaceman showed that the radial flow assumption implicit in equation (6) is not valid for $a$ outside the range 0.5 to 2.0. A more general numerical result obtained by Peaceman for a uniform rectangular grid is

\[ \frac{r_0}{\Delta x} = 0.14(1 + a^2)\frac{r}{\Delta x} \]  

(6)

Peaceman also acquired the result in equation (6) by obtaining an analytical solution of the difference equations and derived the following value for the coefficient in eq.(6): 0.1403 = $[\exp(-\gamma)]/4$, where $\gamma = 0.5772157$ is Euler’s constant.

The traditional Taylor-series based finite difference equations are inaccurate in representing reservoir pressures near the wells in petroleum reservoirs. This is a well known fact. Most simulators do not simulate wellbore pressures directly with finite difference equations, but instead correct simulated well cell (i.e. wellblock) pressures to obtain the actual wellbore pressures with a “well equation”. Many use an empirical productivity index, $PI$:

\[ Q = PI \cdot (p_{well} - p_{cell}) \]  

(7)

In 1978, Peaceman1 was perhaps the first to suggest a method of calculating the $PI$, or the difference in the well bore and well cell pressures:

\[ PI = 2\pi \frac{kh}{\mu} \left[ \ln \left( \frac{0.5\Delta x}{r_{m}} \right) \right]^{-1} \]  

or

\[ p_{well} - p_{cell} = \frac{1}{2} \frac{Q\mu}{2\pi kh} \ln \left( \frac{0.5\Delta x}{r_{m}} \right) \]  

(8)

This expression is based on the pressures in a 2-D, homogeneous, isotropic, reservoir with vertical, fully penetrating wells arranged in a five-spot pattern. The finite difference grid consists of square cells. This expression, “Peaceman’s correction”, is still widely used despite errors that occur when the geometry and reservoir properties differ substantially those of his study. However, since that pioneering work, Peaceman [2] & [3] and others [4] to [7] have proposed alternative well equations to accommodate non-square well cells, anisotropic permeabilities, and off-center and multiple wells within a grid cell. Ding et al. [8] proposed altered transmissibilities between the well cell and neighboring cells, as a companion to the well equation. However, none of the proposed well equations are adequate for all wells, and the growing complexity of well geometries, including horizontal wells, slant wells, and multilateral completions, makes it difficult to know which if any of the well equation is adequate.

Finite Difference Equations Based on Logarithmic Functions:

In an infinite, homogeneous reservoir with steady-state flow, the flow velocity, $q$, from an infinite straight line source in the reservoir, dissipates inversely as the cylindrical area around the line:

\[ q = Q/2 \pi r \]  

where $Q$ is the total flow rate per unit length. If Darcy’s Law, $q = -\frac{k\mu}{\mu} \frac{\partial p}{\partial r}$ is incorporated, the equation can be integrated to obtain:

\[ p = \frac{Q}{2\pi k} \ln(r) + c \]  

(9)

For many, simultaneous line sources, superposition requires that the steady state pressure in an infinite, homogeneous, isotropic reservoir, is given by

\[ p = \frac{\mu}{2\pi k} \Sigma_{all\ wells} Q_n \ln(r_n) + c \]  

(10)

where $r_n$ is the least distance to well $n$, i.e. the perpendicular distance. This fundamental expression suggests that finite difference equations which are based on expressions incorporating a $\Sigma Q\ln(r)$ – term may result in solutions which accurately incorporate the singularity in pressures around the wells. There are many such expressions that might be used. Most, however, do not result in pressure equations which conserve mass.

Perhaps the simplest conservative expression is

\[ p = a \Sigma_{all\ wells} Q_n \ln(r_n) + c \]  

(11)

If we use this equation to find $p$ at grid points $i$ and $i + 1$, the two equations can then be combined to eliminate $c$, and solved for $a$:

\[ a = \frac{p_{i+1} - p_i}{\Sigma_{n=1}^{\Sigma_{all\ wells}} \ln(r_{n,i+1}) - \Sigma_{n=1}^{\Sigma_{all\ wells}} \ln(r_{n,i})} \]  

(12)

Differentiating equation (4) and substituting (5) for $a$, results in a finite difference expression for the derivative:

\[ \frac{\partial p}{\partial x} = -\frac{\Sigma_{n=1}^{\Sigma_{all\ wells}} \frac{\partial p}{\partial x} \ln(r_{n,i+1}) - \Sigma_{n=1}^{\Sigma_{all\ wells}} \frac{\partial p}{\partial x} \ln(r_{n,i})}{\Sigma_{n=1}^{\Sigma_{all\ wells}} \ln(r_{n,i+1}) - \Sigma_{n=1}^{\Sigma_{all\ wells}} \ln(r_{n,i})} \]  

(13)

where $\Sigma$ still represents the sum of all the wells.
Traditional, Taylor-series based, finite difference equations for reservoir simulation approximate this derivative with

$$\frac{\partial P}{\partial x} = \frac{P(i+1)-P(i)}{x_{i+1}-x_{i}} \quad \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \lo