

Full Length Research Paper

Two-dimensional based method in history matching problem

Faridoon Shabaninia

School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran. E-mail: shabani@shirazu.ac.ir.

Accepted 10 July, 2012

In this paper, we presented two recursive techniques for history matching of large-scale reservoir models. At first, singular value decomposition (SVD) is used to reduce the number of state vectors in order to truncate them to a few numbers of state variables corresponding to the largest singular values. The energy of the system is preserved by eliminating the state variables corresponding to the zero singular values or close to zero ones. The Ensemble Kalman filter (EnKF) and SVD are combined to decrease the estimation computational time of the unknown properties in large reservoirs. This approach is computationally efficient and scales well for Kalman filtering as it greatly reduces the size of the sample covariance matrices used to derive state estimates, where the values of the eliminated state variables can be obtained from the rest of state variables. On the other hand, using EnKF in history matching problem needs a huge number of reservoir simulations. In this paper, a new proxy based on fluid flow equations in subsurface, is introduced. This proxy is used instead of simulator during history matching process. Thus, the computational time is greatly reduced. It is shown that the equations in new coordinate axes are quite linear in comparison with highly nonlinear equations in Cartesian or Radial coordinate axes. Therefore, two-dimensional Kalman filter can be used instead of EnKF and so that computational time is reduced, significantly. Finally, the proposed method has been compared with previous methods that have used EnKF and results show the superior advantages in both accuracy and computational time.

Key words: History matching, two-dimensional Kalman filter, Ensemble Kalman filter, singular value decomposition.

INTRODUCTION

Accurate reservoir model plays an important role in reservoir management decisions. On the other hand, simulation model must have consistency with the reservoir production history, to be used for future prediction production. Tuning of model parameters with the aim of matching observation data and output simulation is typically done through history matching process. Therefore, in recent years, there has been an emerging interest to solve the problem through systematic approaches.

The use of Ensemble Kalman filter (EnKF) in history matching process has received significant attention in recent years (Nævdal et al., 2003; Jafarpour and McLaughlin, 2008; Nævdal et al., 2005; Wen and Chen, 2006; Liu and Dean, 2005; Liu, 2005; Gu and Oliver,

2005). It was first proposed by Nævdal et al. (2003). They applied EnKF technique for continuous model updating on two-dimensional reservoir models. In order to extend classical Kalman filter to nonlinear and large-scale problems, EnKF was proposed. This filter was originally developed to update the states, but history matching is often posed as a parameter estimation problem, rather than a state estimation problem (Jafarpour and McLaughlin, 2008). If the parameters of interest are simply added to the state vector, the problem can convert to the state estimation problem. The use of the EnKF for history matching is addressed in this paper.

Numerical reservoir simulators rely on discredited equations since the partial differential equations of fluid flow in porous medium can be solved analytically only in

very simple cases. In order to employ reservoir simulators, the space is subdivided into grid blocks with homogenous properties. Usually, the numbers of grid blocks are very large and can be up to 1 million blocks, especially in real cases. If the number of grid blocks in 2-phase model is N_c then number of states (that is, pressure and saturation of each grid block) is $2N_c$ and it will conclude to a huge state vector. Therefore, EnKF will pose a high computational load. However, a strong correlation exists between property values of neighbouring grid blocks, because geological media are connected layers of rock with similar physical properties (Jafarpour and McLaughlin, 2008). In this paper, we integrate singular value decomposition (SVD) with EnKF to obtain an efficient and accurate history matching algorithm. SVD is performed on the state vector and reduces the number of state variables. The state vector of the reservoir model is truncated to few numbers of states corresponding to the largest singular values. The energy of the system is preserved by eliminating the state variables corresponding to the zero singular values or close to zero ones. This approach reduces the size of the sample covariance matrices that are used to derive state estimates, significantly. This issue makes the method completely appropriate for Kalman filtering. The values of the eliminated state variables can be obtained from the rest of state variables. For example, if the pressure value in a certain grid block is eliminated, it means that the pressure depends on other state variables and there is no need to be calculated independently.

On the other hand, using EnKF in history matching problem needs a huge number of reservoir simulations. In this paper, a new proxy based on fluid flow equations in subsurface, is introduced. This proxy is used instead of simulator during history matching process. Therefore, the computational time is greatly reduced. It is shown that the equations in new coordinate axes are quite linear in comparison with highly nonlinear equations in Cartesian or Radial coordinate axes. Thus, two-dimensional Kalman filter can be used instead of EnKF and computational time is reduced significantly. Finally, the proposed method has been compared with previous methods that have used EnKF and results show the superior advantages of the proposed method in both accuracy and computational time.

The highly nonlinear and complex fluid flow equations are transformed into linear state space model. This model has variable size state vectors. It seems the proxy can also be used in other part of reservoir engineering such as well control.

The outline of this paper is as follows: We first summarize the steps of the EnKF estimation and two-dimensional Kalman filter. Next, the implementation of EnKF in history matching problem is described, and the proposed approach using SVD. Proxy construction is described, and the implementation of two-dimensional Kalman filter in history matching problem is explained.

Simulation example is also described.

ENSEMBLE KALMAN FILTER

The EnKF belongs to particle filters, a broader category of filters. These filters choose a set of sample points. These samples capture the initial probability distribution of the state, and then propagated through the system. The ensemble of the estimate approximates the probability density function of the actual state (Gillijns et al., 2006). Consider a discrete-time nonlinear system as follows:

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) + w_k \\ y_k &= h(x_k) + v_k \end{aligned} \quad (1)$$

Where, $x_k, w_k \in \mathfrak{R}^m, u_k \in \mathfrak{R}^m, y_k$ and $v_k \in \mathfrak{R}^p$. w_k and v_k are stationary zero-mean white noise of process with covariance matrices Q_k and R_k respectively.

All kind of Kalman filters have two main steps, the analysis and the forecast one. These steps in EnKF are summarized as follows:

Analysis step

$$\begin{aligned} \hat{K}_k &= \hat{P}_{xy_k}^f (\hat{P}_{yy_k}^f)^{-1}, \\ x_k^{a_i} &= x_k^f + \hat{K}_k (y_k + v_k^i - h(x_k^f)), \\ \bar{x}_k^a &= 1/q \sum_{i=1}^q x_k^{a_i}. \end{aligned} \quad (2)$$

Forecast step

$$\begin{aligned} x_{k+1}^{f_i} &= f(x_k^{a_i}, u_k) + w_k^i, \\ \bar{x}_{k+1}^{f_i} &= 1/q \sum_{i=1}^q x_{k+1}^{f_i}, \\ E_k^f &= [x_{k+1}^{f_1} - \bar{x}_{k+1}^f \dots x_{k+1}^{f_q} - \bar{x}_{k+1}^f], \\ E_{y_k}^a &= [y_k^{f_1} - \bar{y}_k^f \dots y_k^{f_q} - \bar{y}_k^f], \\ \hat{P}_{xy_k}^f &= \frac{1}{q-1} E_k^f (E_{y_k}^f)^T, \\ \hat{P}_{yy_k}^f &= \frac{1}{q-1} E_{y_k}^f (E_{y_k}^f)^T. \end{aligned} \quad (3)$$

In fact, the forecast ensemble mean, \bar{x}_k^f , is interpreted as the best forecast estimate of the state, and the spread of the ensemble members around the mean between the best estimate and actual state (Gillijns et al., 2006).

In Kalman filtering process, the main source of the computational burden is determining covariance. EnKF approximates it, so the computational burden in EnKF is less than other Kalman filter, especially extended Kalman filter.

TWO-DIMENSIONAL KALMAN FILTER

Here, algorithm of two-dimensional Kalman filter for modified Fornasini and Marchesini (MFM) model is described. The model is transformed to wave advanced model (WAM). Two-dimensional Kalman filter is implemented to WAM model. Consider two-dimensional MFM model as follows:

$$\begin{aligned} x(m+1, n+1) &= J(m, n)x(m+1, n) \\ &+ K(m, n)x(m, n+1) + E(m, n)w(m, n+1) \\ &+ F(m, n)w(m+1, n) + u(m, n) \\ z(m, n) &= C(m, n)x(m, n) + v(m, n) \end{aligned} \quad (4)$$

Where, u, v, w are non-random input, measurement noise and input noise respectively. By defining vectors as follows:

$$\begin{aligned} X(n) &= \text{col}[x(0, n), x(1, n-1), \dots, x(n, 0)] \\ W(n) &= \text{col}[w(0, n), w(1, n-1), \dots, w(n, 0)] \\ V(n) &= \text{col}[v(0, n), v(1, n-1), \dots, v(n, 0)] \\ Y(n) &= \text{col}[z(0, n), z(1, n-1), \dots, z(n, 0)] \\ f(n) &= \text{col}[u(0, n), u(1, n-1), \dots, u(n, 0)] \end{aligned} \quad (5)$$

The MFM model is transformed to WAM model as follows:

$$\begin{aligned} X(k+1) &= A(k)X(k) + B(k)W(k) + E(k)f(k) \\ Y(k+1) &= H(k+1)X(k+1) + D(k+1)V(k+1) \end{aligned} \quad (6)$$

By defining covariance of $X_{k|k-1}, V_{k|k-1}$ as $\sum, R(k)$, respectively, the optimal state estimation of MFM model is obtained as follows:

$$\begin{aligned} \hat{X}_{k|k} &= \hat{X}_{k|k-1} + \\ &\sum_{k|k-1}^{k|k-1} H^T(k) [H(k) \sum_{k|k-1}^{k|k-1} H^T(k) + R^k]^{-1} \{Y(k) - H(k) \hat{X}_{k|k-1}\} \\ \sum_{k|k} &= \sum_{k|k-1}^{k|k-1} - K(k) H(k) \sum_{k|k-1}^{k|k-1} \\ \sum_{k+1|k} &= A(k) \sum_{k|k}^{k|k} . A^T(k) + B(k) Q(k) B^T(k) \\ K(k) &= \sum_{k|k-1}^{k|k-1} . H^T(k) [H(k) \sum_{k|k-1}^{k|k-1} H^T(k) + R^k]^{-1} \end{aligned} \quad (7)$$

The result (Equation 7) can be used for implementing filtering algorithm of two-dimensional systems such as model that is later described.

IMPLEMENTATION OF ENKF IN HISTORY MATCHING

Using EnKF for solving history matching problem has been considered in many articles, but the details of fully automated algorithm has not been described. In this paper, the algorithm is shown by a flow chart and the schemes which are used to run EnKF in MATLAB and link the data to Eclipse are explained.

At first the model with real data must be run by Eclipse. The desired outputs, later specified are written in RSM files after each time step of simulation. The state variables, pressure and saturation of grids are written in RST files. Then real outputs and initial values of state variables are read by MATLAB.

As mentioned before, parameters are added to state vector and dynamic variables and model parameters will be estimated simultaneously. Furthermore, the state vector is extended by the measurement variables or outputs. By this way, the relationship between state vector and outputs will be linear. If we have p outputs and the size of state vector is N , then:

$$y_k = Hx_k, \text{ where } H = [0_{p \times (N-p)} \quad I_{p \times p}] \quad (8)$$

And the Kalman gain can be computed as:

$$\hat{K}_k = \hat{P}_{xy_k}^f H^T (H \hat{P}_{xy_k}^f H^T + R_k)^{-1} \quad (9)$$

Where, R is covariance matrix of measurement noise. According to Equations 4 and 5, it can be seen the last p rows of matrix $\hat{P}_{xy_k}^f$ are needed for computation of Kalman gain, thus, the computational time is greatly reduced, so the state vector is defined as follows:

$$x = \begin{bmatrix} x_d \\ x_s \\ d \end{bmatrix} \quad (10)$$

Where, d is output vector and x_d, x_s are dynamic variables and static variables, respectively:

$$x_d = \begin{bmatrix} p_1 \\ \vdots \\ p_{N_c} \\ s_1 \\ \vdots \\ s_{N_c} \end{bmatrix}, x_s = \begin{bmatrix} k_1 \\ \vdots \\ k_{N_c} \end{bmatrix} \quad (11)$$

P_i, S_i and $K_i (i=1, \dots, N_c)$ are pressure, saturation and model parameter (for example, permeability) of i^{th} grid block respectively. N_c is number of grid blocks. Initial ensembles are described by Equation 12:

$$x_0^f = \bar{x}_0^f + e_0^m \quad (12)$$

Where, \bar{x}_0^f is mean of the initial ensemble and e_0^m ($i=1, \dots, q, q$ is the number of ensembles) are drawn from a mixture distribution with zero-mean. We assume the interest parameter in grid block (i_1, j_1, k_1) is correlated with that parameter in grid block (i_2, j_2, k_2) with correlation coefficient:

$$e = \frac{(i_1 - i_2)^2 - (j_1 - j_2)^2 - (k_1 - k_2)^2}{L} \quad (13)$$

L is correlation length. Equation 9 means the parameter is spatially correlated with Gaussian correlation model. A correlation matrix C is computed by using the correlation coefficients as mentioned above. The covariance of the model noise is computed as $\sigma^2 C$ where σ is the standard deviation.

All of ensemble variables are defined except, y_k^f :

$$y_k^f = \bar{y}_k^f + e_k^i \quad (14)$$

Where, e_k is white noise with zero-mean and covariance R_k .

After the determination of initial values, EnKF applies to them. In Analysis step, the approximation of covariance matrix is computed. This step is the main source of computational burden. SVD is further used to reduce the size of state vector, consequently reduces the size of the covariance matrix.

In the Forecast step, the simulator must be run q times, where q is the number of ensembles. In this step, the following steps must be done:

- 1) Obtained values in Analysis step (pressure, saturation and interest model parameter) are saved in separate vectors.
- 2) These values written include files instead of initial values of the model by MATLAB.
- 3) Eclipse must be run for a time step.
- 4) Outputs and values of pressure and saturation are read from RSM and RST files, respectively. In Forecast step, the values of interest parameter are fixed.

After the aforementioned steps, get back to Analysis step.

SVD ASSISTED ENKF FOR HISTORY MATCHING

Here, SVD concept and the way to use it in history matching problem are described.

At first, the data matrix contains vectors of pressures and saturations of all grid blocks in all time steps, as defined as follows:

$$X = [x_d(t_1) \dots x_d(t_n)]_{2N_c \times n} \quad (15)$$

Where $t_i (i=1, \dots, n)$ is i^{th} time step. Matrix X can be transformed into the form of Equation 12 by SVD:

$$X = USV^T \quad (16)$$

With, $U^T U = U U^T = I_{2N_c}$, $V^T V = V V^T = I_n$ and S being $2N_c \times n$ with singular values of X on the diagonal. Singular values of X are positive square roots of eigen values of XX^T . The columns of V are orthonormalized eigen vectors of $X^T X$ and the columns of U are orthonormalized eigen vectors of XX^T . The rank of X equals the number of non-zero singular values. If the rank of X is r , the first r columns of S are orthonormal basis of the range space of X . Singular values of stable system indicate the respective state energy of the system. Therefore, reduced order can be directly determined by examining the system singular values. If s_1, \dots, s_n are singular values of X in decreasing order, singular values with small amount can be removed. The remaining energy to total energy is calculated as follows:

$$\frac{\sum_{i=1}^l s_i^2}{\sum_{i=1}^n s_i^2} \quad (17)$$

Transfer matrix ϕ will contain only first l columns of U . Now the reduced state vector, designated z_d , can be calculated as:

$$z_d = \phi^T x_d \quad (18)$$

As previously discussed, U is orthonormal, thus:

$$\phi \phi^T = I \quad (19)$$

$$x_d = \phi z_d \quad (20)$$

The values of the eliminated state variables can be obtained from the rest state variables by using Equation

20. For example, if the pressure value in a certain grid block is eliminated, it means that the pressure depends on other state variables and there is no need to be calculated independently. After reducing dynamic state vector, the reduced state vector can be defined as:

$$z = \begin{bmatrix} z_d \\ x_s \\ d \end{bmatrix}. \quad (21)$$

The static vector and the output vector are kept completely in z . If the total transfer matrix is defined as Equation 18, the state vector and its reduced have relationship as follows:

$$\phi = \begin{bmatrix} \phi_{2N \times d} & 0_{2N \times N_c} & 0_{2N \times p} \\ 0_{N \times d} & I_{N_c} & 0_{N \times d} \\ 0_{p \times d} & 0_{p \times N_c} & I_p \end{bmatrix} \quad (22)$$

$$x = \phi z. \quad (23)$$

$$z = \phi^T x. \quad (24)$$

Note that after transformations, all the elements in static vector will remain, constantly and therefore no error will emerge while transferring the state vector to the reduced state vector and vice versa. In addition, the output vector is remained in z to preserve linear relationship between output and state variables. The predicted measurements are related nonlinearly to the model state and their inclusion in the state vector simplifies comparison with measure data in the Forecast step of EnKF.

After calculating the reduced state vector, Analysis step of EnKF can be done on z . The computational time is greatly reduced by this method. Then, the main state vector is obtained by Equation 24 and initialized the simulator to run for one step time and obtaining the state vector in Forecast step. This loop is repeated until the end of time steps. Flow chart of the method is shown in Figure 1.

PROXY CONSTRUCTION

The general partial differential equation (PDE) fluid flow equations can be approximated for the grid block ijk as:

$$V_{ijk} W_{ijk} \frac{dx_{ijk}}{dt} = T_{ijk} \begin{bmatrix} x_{i,j,k-1} \\ x_{i,j-1,k} \\ x_{i-1,j,k} \\ x_{i,j,k} \\ x_{i+1,j,k} \\ x_{i,j+1,k} \\ x_{i,j,k+1} \end{bmatrix} + V_{ijk} \tilde{q}_{ijk} \quad (25)$$

Where:

$$x_{ijk} = \begin{bmatrix} P_{ijk} \\ S_{ijk} \end{bmatrix} \quad (26)$$

$$\tilde{q}_{ijk} = \begin{bmatrix} \frac{q_o^{ijk}}{\rho_o} \\ \frac{q_w^{ijk}}{\rho_w} \end{bmatrix} \quad (27)$$

$$V_{ijk} = \begin{bmatrix} \Delta V \rho_o & 0 \\ 0 & \Delta V \rho_w \end{bmatrix} \quad (28)$$

$$W_{ijk} = \begin{bmatrix} \phi(1-S)(c_o + c\phi) & -\phi \\ \phi S(c_w + c\phi) & \phi[(1-S)(c_w + c\phi)(\frac{dP_c}{dS})] \end{bmatrix}_{ijk} \quad (29)$$

$$T_{ijk} = \begin{bmatrix} \tilde{T}_{i,j,k-\frac{1}{2}}^o & \tilde{T}_{i,j-\frac{1}{2},k}^o & \tilde{T}_{i-\frac{1}{2},j,k}^o \\ \tilde{T}_{i,j,k-\frac{1}{2}}^w & \tilde{T}_{i,j-\frac{1}{2},k}^w & \tilde{T}_{i-\frac{1}{2},j,k}^w \\ -\tilde{T}_{i,j,k}^o & \tilde{T}_{i+\frac{1}{2},j,k}^o & \tilde{T}_{i,j+\frac{1}{2},k}^o & \tilde{T}_{i,j,k+\frac{1}{2}}^o \\ -\tilde{T}_{i,j,k}^w & \tilde{T}_{i+\frac{1}{2},j,k}^w & \tilde{T}_{i,j+\frac{1}{2},k}^w & \tilde{T}_{i,j,k+\frac{1}{2}}^w \end{bmatrix} \quad (30)$$

By breaking T_{ijk} into T_{ijk}^1 and T_{ijk}^2 , the state space equation for grid block ijk can be written as follows:

$$V_{ijk} W_{ijk} \frac{dx_{ijk}}{dt} = T_{ijk}^1 \begin{bmatrix} x_{i,j,k-1} \\ x_{i,j-1,k} \\ x_{i-1,j,k} \\ x_{i,j,k} \end{bmatrix} + T_{ijk}^2 \begin{bmatrix} x_{i+1,j,k} \\ x_{i,j+1,k} \\ x_{i,j,k+1} \end{bmatrix} + V_{ijk} \tilde{q}_{ijk} \quad (31)$$

Equation 31 can be written as follows:

$$T_{ijk}^2 \begin{bmatrix} x_{i+1,j,k} \\ x_{i,j+1,k} \\ x_{i,j,k+1} \end{bmatrix} = V_{ijk} W_{ijk} \frac{dx_{ijk}}{dt} - T_{ijk}^1 \begin{bmatrix} x_{i,j,k-1} \\ x_{i,j-1,k} \\ x_{i-1,j,k} \\ x_{i,j,k} \end{bmatrix} - V_{ijk} \tilde{q}_{ijk} \quad (32)$$

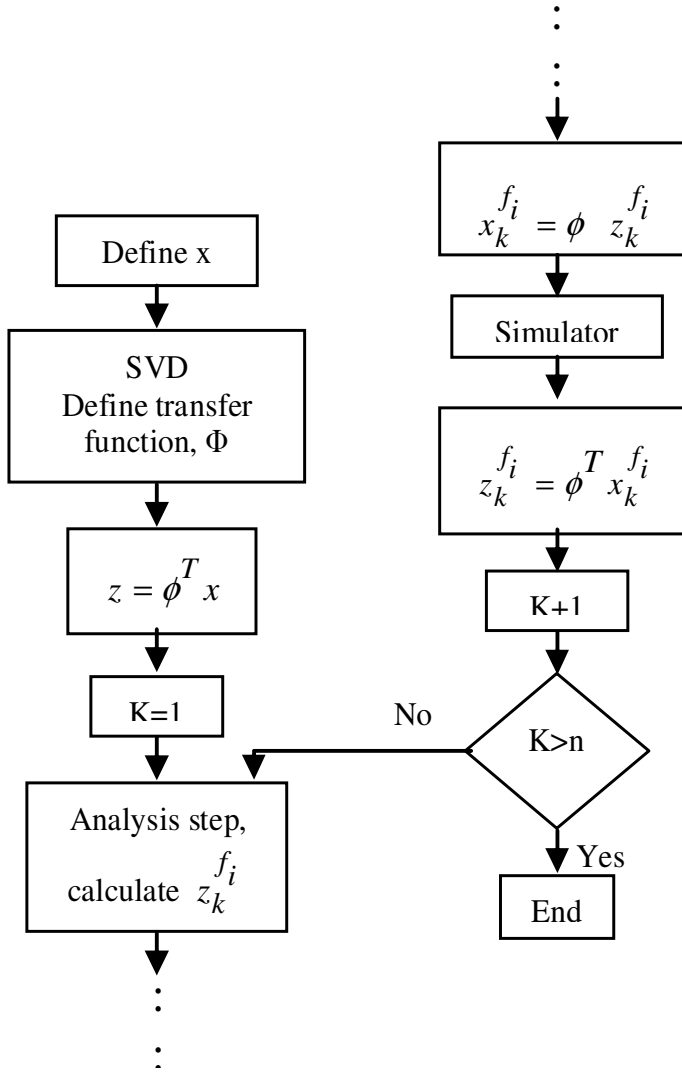


Figure 1. Flow chart of the proposed method.

It can be deduced that the dynamic values of state variables in each grid can be obtained from its' neighbours. In this paper, we claim the relationships between neighbours are linear and it is shown by simulation results. In MFM model Equation 4, the state variables are pressure and saturation of each grid.

$$x(m,n) = \begin{bmatrix} p(m,n) \\ S(m,n) \end{bmatrix} \quad (33)$$

Thus, $J(m,n), K(m,n)$ and $C(m,n)$ are unknown for all grid blocks. In history matching problem, we have access to data over time. Therefore, least square method (Lsq) can be implemented to the problem for each grid block. The dimension of matrix are small, thus, this idea has not any computational burden.

$$y = \begin{bmatrix} P(m+1,n+1) \\ S(m+1,n+1) \end{bmatrix} \quad (34)$$

$$u = \begin{bmatrix} P(m+1,n) \\ S(m+1,n) \\ P(m,n+1) \\ S(m,n+1) \end{bmatrix} \quad (35)$$

$$y = u^T \theta, \theta = \begin{bmatrix} J_{(1,1)} & J_{(2,1)} \\ J_{(1,2)} & J_{(2,2)} \\ K_{(1,1)} & K_{(2,1)} \\ K_{(1,2)} & K_{(2,2)} \end{bmatrix} \quad (36)$$

By Lsq method, we have:

$$Y = \begin{bmatrix} y_1^T \\ y_2^T \\ \vdots \\ y_n^T \end{bmatrix}_{n \times 2}, U^T = \begin{bmatrix} u_1^T \\ u_2^T \\ \vdots \\ u_n^T \end{bmatrix}_{n \times 4}, Y = U^T \theta \quad (37)$$

$$\hat{\theta} = (U^T U)^{-1} U^T Y \quad (38)$$

Streamline simulation

The breakthrough work for tracing streamlines efficiently in three-dimensional was that of Pollock (1988). Pollock's method is simple, analytical, and is formulated in terms of a time-of-flight (TOF) coordinate (Figure 2). To apply Pollock's tracing method to any cell, the total flux in and out of each boundary is calculated using Darcy's Law. With the flux known, the algorithm centres on determining the exit point of a streamline and the time to exit given any entry point assuming a piece-wise linear approximation of the velocity field in each coordinate direction. The equations are simple: if v is the interstitial velocity ($v=u/\phi$), then a linear velocity description in the x -direction gives:

$$v_x = v_{x0} + g_x(x - x_0); \quad g_x = \frac{v_{x,x} - v_{x_0}}{\Delta x} \quad (39)$$

Where v_{x0} is the x -velocity at $x=x_0$, and g_x is the velocity gradient in the x -direction. Since $v_x = dx/dt$, we can integrate the expression of the x -velocity (and in analogous fashion in the y - and z -direction) to get the exit times out of each face given an arbitrary entry point (x_i, y_i, z_i) and exit coordinates x_e, y_e , and z_e .

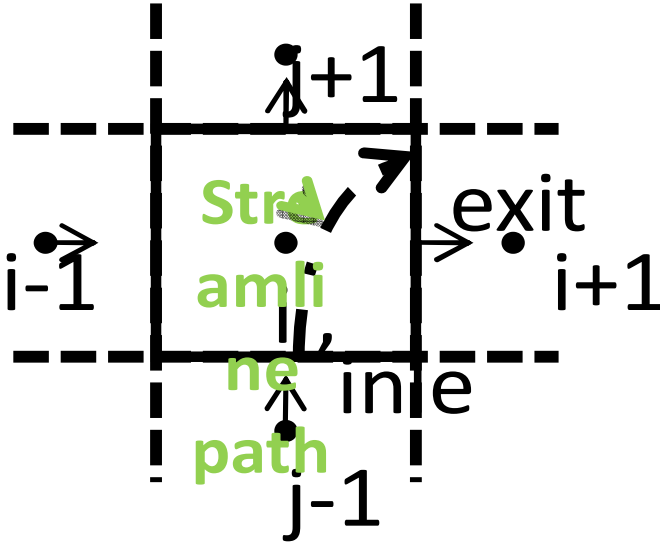


Figure 2. Pollock's three-dimensional tracing method through a Cartesian cell. Given an arbitrary entry point, the time to exit and the exit point can be determined analytically.

$$\Delta t_x = \frac{1}{g_x} \ln \left[\frac{v_{x0} + g_x(x_e - x_0)}{v_{x0} + g_x(x_i - x_0)} \right] \quad (40)$$

$$\Delta t_y = \frac{1}{g_y} \ln \left[\frac{v_{y0} + g_y(y_e - y_0)}{v_{y0} + g_y(y_i - y_0)} \right] \quad (41)$$

$$\Delta t_z = \frac{1}{g_z} \ln \left[\frac{v_{z0} + g_z(z_e - z_0)}{v_{z0} + g_z(z_i - z_0)} \right] \quad (42)$$

Since the streamline must exit from the face having the smallest travel time, $\Delta t_m = \text{MIN}(\Delta t_x, \Delta t_y, \Delta t_z)$, the exit locations are calculated by re-solving for x_e , y_e , and z_e using the minimum time:

$$x_e = \frac{1}{g_x} \ln [v_{xi} \exp(g_x \Delta t_m) - v_{x0}] + x_0 \quad (43)$$

$$y_e = \frac{1}{g_y} \ln [v_{yi} \exp(g_y \Delta t_m) - v_{y0}] + y_0 \quad (44)$$

$$z_e = \frac{1}{g_z} \ln [v_{zi} \exp(g_z \Delta t_m) - v_{z0}] + z_0 \quad (45)$$

Pollock's equations are derived assuming orthogonal grid blocks, but very few real reservoirs models use such a strict Cartesian framework anymore. Using an isoparametric transformation, it is possible to transform

corner-point geometry grids (CPG) into unit cubes, apply Pollock's method, and then transform the exit coordinate back to physical space.

The understanding that using a TOF-variable along streamlines rather than a volume-variable along is in terms of streamtubes came through the reformulation of the three-dimensional mass conservation equation in terms of TOF. For incompressible and immiscible flow without gravity, the conservation equation for a phase j can be written as:

$$\phi \frac{\partial S_j}{\partial t} + \vec{v}_i \cdot \nabla f_j = 0 \quad (46)$$

Where S_j is the saturation of phase, j , $\vec{v}_i = \sum \vec{v}_j$ is the total velocity and f_j is the fractional flow of phase j . By defining a coordinate ξ that is parallel to v (i.e. streamline) it is possible to write that

$$\vec{v}_i \cdot \nabla = |\vec{v}_i| \frac{\partial}{\partial \xi} \quad (47)$$

Now consider the definition of the TOF, which leads to the following expression:

$$\tau = \int \frac{\phi}{|\vec{v}_i|} d\xi \quad (48)$$

$$\frac{\partial \tau}{\partial \xi} = \frac{\phi}{|\vec{v}_i|} \quad (49)$$

$$|\vec{v}_i| \frac{\partial}{\partial \xi} = \vec{v}_i \cdot \nabla = \phi \frac{\partial}{\partial \tau} \quad (50)$$

Allowing the three-dimensional conservation equation to be re-written as:

$$\frac{\partial S_j}{\partial t} + \frac{\partial f_j}{\partial \tau} = 0 \quad (51)$$

There are a number of assumptions buried in this derivation. For example, that the flow rate along each streamline is constant, that the streamlines do not change over time and that the one-dimensional solutions must have the same boundary and initial conditions as the three-dimensional problem. But the derivation shows (Figure 3) that a three-dimensional transport problem can be re-written in terms multiple, one-dimensional problems along streamlines. While this was known intuitively from the work on streamtubes, the TOF formulation offers a compelling mathematical framework. For the simple case

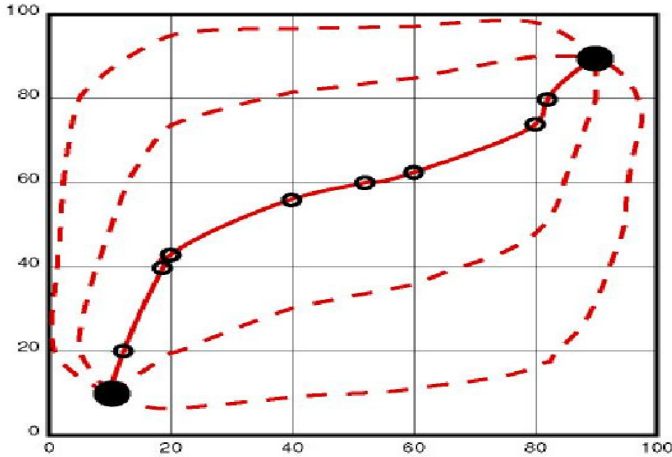


Figure 3. The exit point of one cell becomes the entry point for the next cell. By connecting exit and entry points a streamline is traced from injector to producer.

of an incompressible water flood, it is thus, possible to write:

$$\phi \frac{\partial S_j}{\partial t} + \vec{v}_i \cdot \nabla f_i = 0 = \sum_{streamlines}^{all} \frac{\partial S_j}{\partial t} + \frac{\partial f_j}{\partial \tau} \quad (52)$$

The most important detail about this equation is that the total velocity in the three-dimensional problem has disappeared into the TOF of each individual streamline. It is this decoupling of a three-dimensional heterogeneous system into a series of one-dimensional homogenous systems in terms of TOF that makes the SL method so attractive.

THE PROPOSED LINEAR PROXY ASSISTED TWO-DIMENSIONAL KALMAN FILTER IN HISTORY MATCHING

Two-dimensional Kalman filter was originally developed to update the states, but history matching is often posed as a parameter estimation problem, rather than a state estimation problem. If the parameters of interest are simply added to the state vector, the problem can convert to the state estimation problem.

At the beginning, the parameters are far from reality, thus, after a few steps of filtering to update the states and parameters, it is necessary to update the linear model to reduce the possible error.

Although the parameters are static, but estimated ones will change during time steps. Therefore, in order to achieve the best answer of all estimations over time, we use weighted average of interested parameters. The weights are inverse of errors in each iteration:

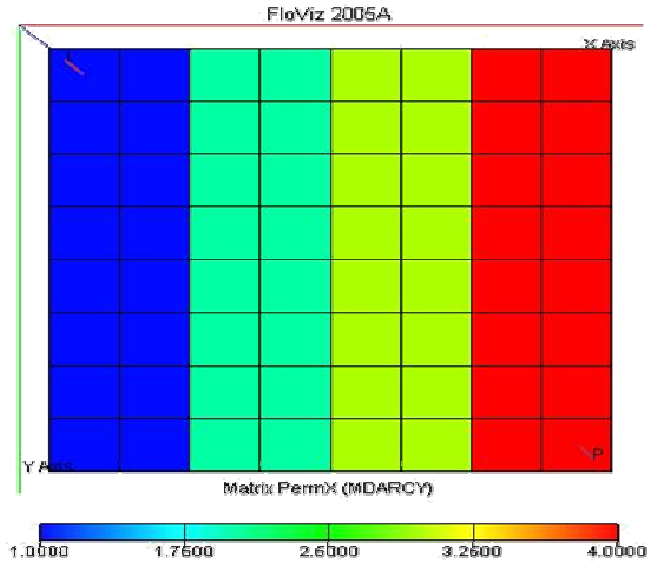


Figure 4. Permeability in x-direction of top layer.

$$k_{final} = \frac{\sum_{i=1}^n k(i) |e(i)|}{\sum_{i=1}^n 1/|e(i)|} \quad (53)$$

In history matching process, consistency of model with two sets of data is necessary:

- 1) Geological data: the model should be based on sound geological and geostatistical principles;
- 2) Matching between real data and simulated data.

In proposed algorithm, two cases are guaranteed which are further shown by simulations.

Example

In this example, we consider a three-dimensional reservoir, which is discretized by $8 \times 8 \times 2$ grids of 75^m length along each horizon and 30^m depth (Cardoso et al., 2009; Cardoso and Durlofsky, 2010; Chi-tsong, 1999; Maciejowski, 1989). The permeability distribution of the top layer is shown in Figure 4. The model contains one production well and a single water injector. Two constant values of 0.19 and 100 are used for the porosity in all grid cells of layer #1 ($z=1$) and layer #2 ($z=2$), respectively. True permeability varies linearly as a function of x-coordinate in the interval $[1,4]^mD$ and $[100,400]^mD$ in layer #1 and layer #2, respectively. These distributions of permeability are chosen according to the example in Nævdal et al. (2005).

The production well is controlled by liquid rate target of

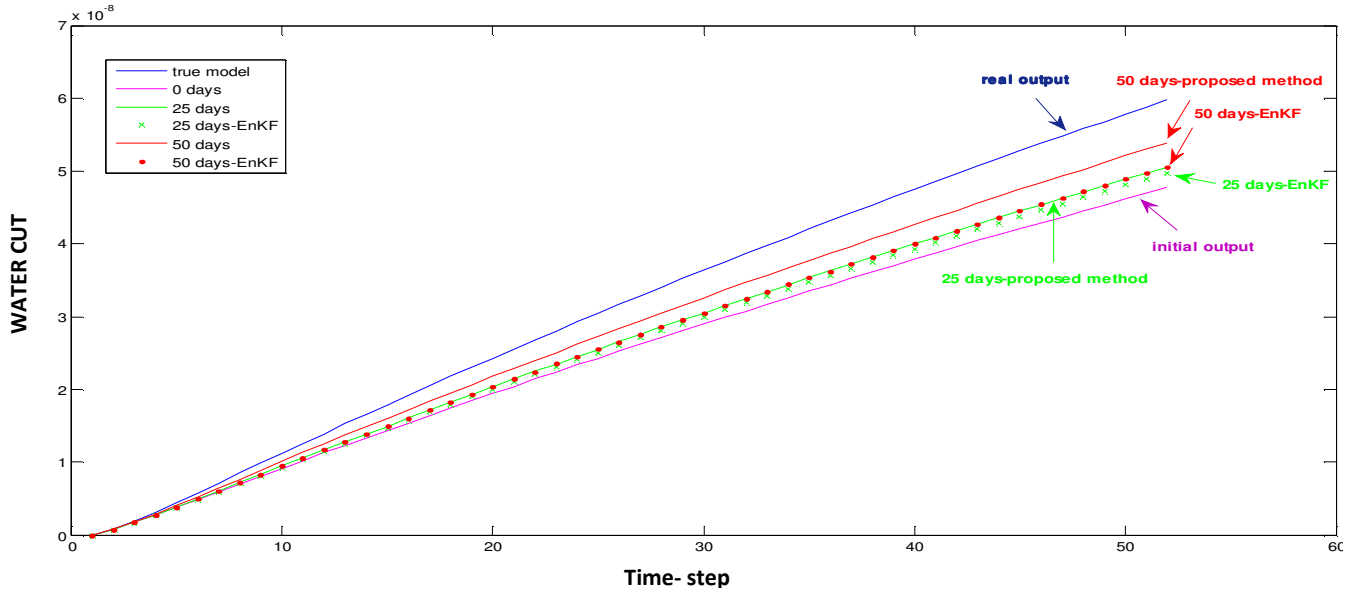


Figure 5. WCT of the producer well.

$210 \text{ Sm}^3/\text{day}$ and the injection well is controlled by surface flow rate target of $200 \text{ Sm}^3/\text{day}$. The reservoir is initially saturated with oil with constant initial pressure of 3981.2 psi . The measurements are generated by running the simulator with “true” permeability and adding noise to the resulting water cut of the producer well. Measurement is assimilated once every 100 days for 50 times. The initial estimated permeability is equal to 3^{mD} for all grids at top layer.

Running the EnKF all quantities are kept constant except for the permeability and the state variables, pressures and saturations. The ensemble consists of 10 members. The initial ensemble is generated using a mean correlation length of 3 grid blocks with a standard deviation of 1.

In this case, the number of state variables of the state vector x_d is 128. By using SVD, only 14 state variables remind. Ratio of the lost energy to the total energy is equal to 10^{-7} indicates that the error resulting from reducing state vector is very low and negligible and also shows all the energy is in the 14 remaining state variables. We realized these states are near the wells. It is obvious the computational time of implementing EnKF for 128 state variables is very greater than 14 state variables.

The water cut of the producer well is closer to the real values over time-steps. In Figure 5, forecast based on the ensemble mean after 25 and 50 time-steps is shown. The dash lines are result of using EnKF and the solid lines is result of the proposed method. The result of the proposed method is better than the previous method (Nævdal et al., 2003). The proposed method not only has less computational time but also has a better matching than

the previous method. In Figure 6, the estimation of permeability field of two methods are shown. It is shown that the permeability field of proposed method is closer to the real one. It seems two estimated fields are far from real one. In Figure 7, the estimated permeability field using two-dimensional based method is shown. The permeability in near of the production well is more important than other grids. The proposed method has accurate field than others. In Figure 8, the comparison between real output and estimated output is shown. The estimated output is matched to real one completely.

In order to compare quality of solution of three methods, normalized root mean square (NRMS) error is used as the criterion, which is defined as:

$$\sqrt{\frac{\sum (Y_j - \hat{Y}_j)^2}{\sum Y_j^2}} \quad (54)$$

Where, n is number of time-steps (in this case it equals to 50), \hat{Y}_j is forecast of output at j^{th} time-step and Y_j is the real output at j^{th} time-step. In Table 1, RMS of three methods is compared, quantitatively, which shows that, the proposed two-dimensional based method is more accurate. In addition, numbers of simulator running to obtain estimated field are compared. This number of two-dimensional based method is much less than others. The computational burden of this method is greatly reduced.

CONCLUSIONS

The paper has considered an application of a recursive technique, the EnKF, to history matching problem. An algorithm is simplified in a way of reducing the order of

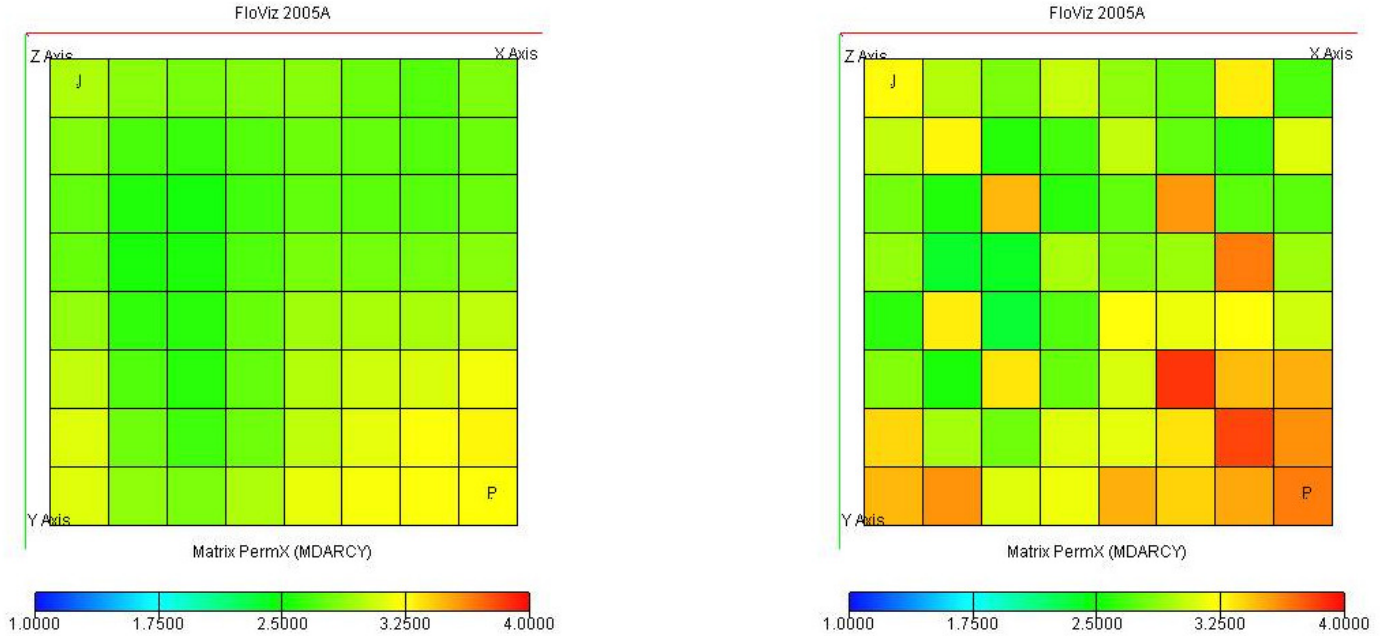


Figure 6. Permeability field of top layer. The estimation of EnKF method (Left) and the estimation of the proposed method (Right).

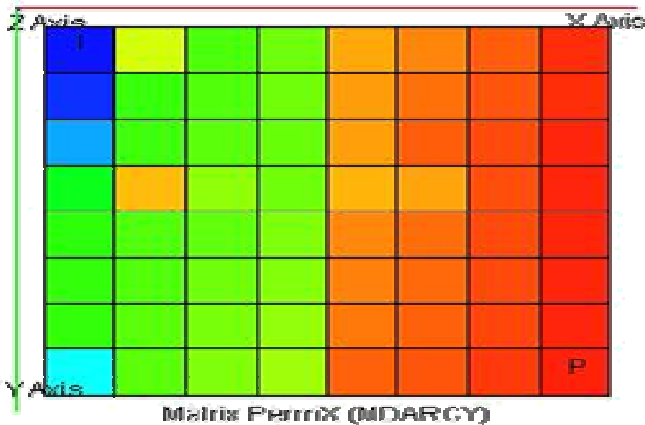


Figure 7. The estimation of permeability field by using proposed two-dimensional based methods.

the state vector to a few numbers of states corresponding to the largest singular values by using SVD. The energy of the system was preserved by eliminating the state variables corresponding to the zero singular values or close to zero ones. In this way, the computational time is greatly reduced. Computational effort required for history matching of large reservoirs is a major problem. Computational time reduction even 1%, can have dramatic impact on solving history matching problem of real field. Practical problems of this paper were too much for example hundreds of Eclipse running times was needed for each test. The algorithm of the proposed

method was described and tested by a synthetic model. The simulation results show decreasing the size of state vectors from 128 to 14. Therefore, significant reduction of the computational time of the estimation of unknown geological properties is evident in comparison with a much more expensive approach that estimates states in every grid blocks. Furthermore, the estimation results with the proposed method were better than the results obtained with the previous method. Transition matrix calculation can be done offline. Thus, the proposed method can be used as accurate and fast method for real time problems. Problems of using EnKF in history matching problem have been addressed in this paper. Two-dimensional based method was introduced to solve these problems. A new proxy based on fluid flow equations in subsurface, is introduced. This proxy is used instead of simulator during history matching process. Thus, the computational time is greatly reduced. It is shown that the equations in new coordinate axes are quite linear in comparison with highly nonlinear equations in Cartesian or Radial coordinate axes. Therefore, two-dimensional Kalman filter can be used instead of EnKF and computational time is reduced, significantly. Finally, the proposed method has been compared with previous methods that have used EnKF and results show the superior advantages of the proposed method in both accuracy and computational time.

The highly nonlinear and complex fluid flow equations are transformed into linear state space model. This model has variable size state vectors. It seems the proxy can also be used in other part of reservoir engineering such as well control.

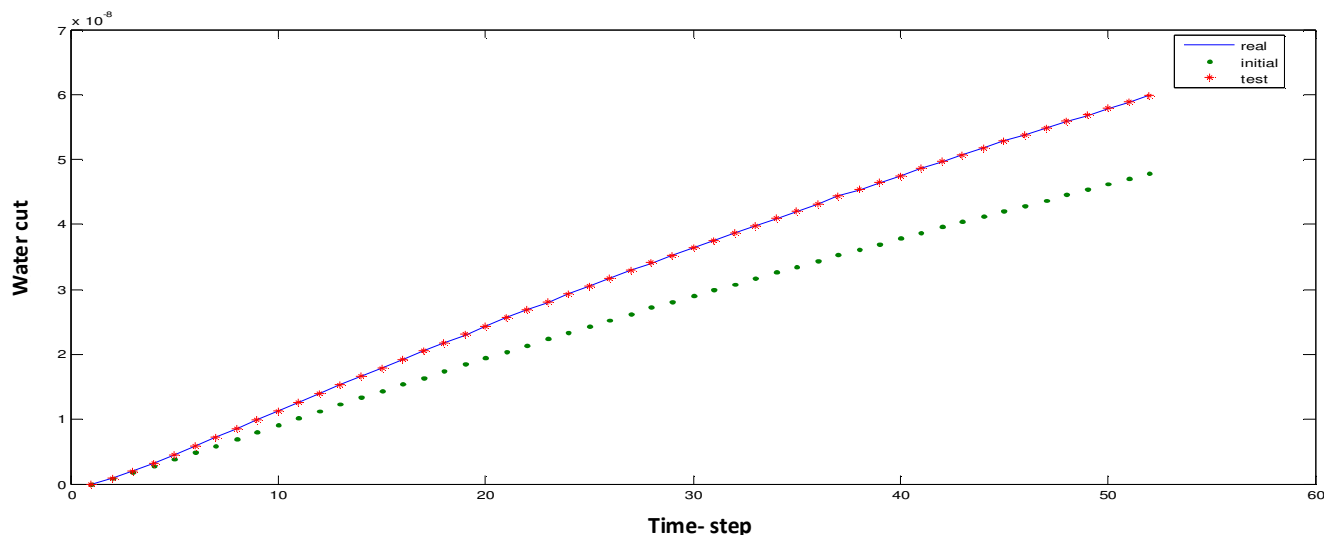


Figure 8. Water cut of the producer well- initial is green, real one is blue and estimated one is red.

Table 1. Comparison between three methods.

Method	Number of simulator running	NRMSE
Two-dimensional based method	2	1.4976×10^{-3}
EnKF	520	1.6583×10^{-1}
SVD assisted EnKF	520	1.0142×10^{-1}

REFERENCES

- Cardoso MA, LJ Durlofsky (2010) Linearized reduced-order models for subsurface flow simulation. *J. Comput. Phys.* 229(4):681-700.
- Cardoso MA, LJ Durlofsky, Sarma P (2009). Development and application of reduced-order modeling procedures for subsurface flow simulation. *Int. J. Numer. Methods Eng.* 77(3):1322-1350.
- Chi-tsong C (1999). *Linear System Theory & Design.*, 3d ed., Oxford University press pp.76-77.
- Gillijns S, Barrero O, Mendoza JC, De Moor BLR, Bernstein DS, Ridley A (2006). What is Ensemble Kalman filter and how well does it work. *IEEE American Control Conference.*
- Gu Y, Oliver DS (2005). History matching of the PUNQ-S3 reservoir model using ensemble Kalman filter. *SPE J.* 10(2):217-224.
- Jafarpour B, McLaughlin DB (2008) History matching with an ensemble Kalman filter and discrete cosine parameterization. *Comput. Geosci.* J. 12(2):227-244.
- Liu N (2005). Critical evaluation of the ensemble kalman filter on history matching of geologic facies. *SPE Reservoir Simulation Symposium*, The Woodlands, Texas.
- Liu N, Dean SO (2005). Ensemble Kalman filter for automatic history matching of geologic facies. *J. Pet. Sci. Eng.* 147(3):147-161.
- Maciejowski JM (1989). *Multivariable Feedback Design.* Addison Wesley pp.76-80.
- Nævdal G, Johnsen LM, Aanonsen SI, Vefring EH (2005). Reservoir monitoring and continuous model updating using ensemble Kalmanfilter. *SPE J.* 10(1):66-74.
- Nævdal G, Mannseth T, Vefring EH (2003) Near-Well Reservoir Monitoring Through Ensemble Kalman Filter. *SPE-75235*, In *Proceeding of SPE/DOE Improved Oil Recovery Symposium.*
- Wen XH, Chen W (2006). Real-time reservoir model updating using ensemble Kalman filter. *SPE J.* 11(4):431-442.