Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

A research on the optimization of the reservoir model

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

Yinglei Yu

Delft, the Netherlands
July 30

Abstract

With a high demand of oil, optimization of oil output in the reservoir field is urgent and significant. Practical production problems often involve large, highly complex reservoir models, with up to thousands of unknowns and many constraints. Moreover, our understanding of the reservoir is always highly uncertain, and this uncertainty can be reflected in the reservoir models. Consequentially, performance prediction and performance optimization, which are the ultimate goals of the entire modeling and simulation process, depend a lot on the data assimilation process and uncertainty from reservoir models themselves.

In this thesis, the main goal is to discuss two ways of optimizing oil production and do research on the uncertainty of optimization according to data assimilation and reservoir models. Chapter 2 states an efficient method of data assimilation—Asynchronous Ensemble Kalman Fliter and its application in a square reservoir model. Chapter 4 and 5 discuss the two aspects of production optimization respectively: optimization of control of well settings and optimization of well placement. I use gradient-based method and implement it into simsim simulator, which is a simple simulator of reservoir model. In Chapter 5, I apply Particle Swarm Optimization method to optimize well placement given a fixed well settings. In Chapter 6, I do a research on the uncertainty of optimization results due to data assimilation and then study on the optimization process given a true permeability and porosity field. I found a stable performance of the optimization when applying field data estimated by data assimilation and demonstrate the effects of two types of optimization. Some disadvantage and future recommendations are presented in the last part.

Key words: petroleum reservoir model, EnKF, Gradient-based Optimization, Particle Swarm Optimization

Acknowledgements

When I came to TUD two years ago at the first time, I had no idea about what applied mathematics really is. I have got a high average score in my Bachelor courses but a weak numerical concept and weak analytical skills for real life problems. It is fantastic for me to discover the beauty of applying mathematics to real problems and implementing formula to computer during working through my graduation project.

Firstly, I would like to thank to Prof. A.W. Heemink and Dr Remus Hanea for their instructions and help on my project. They taught me a lot of knowledge about reservoir management and numerical analysis. I also appreciate my supervisor of the first year—Dr. D. Kurowicka, who also gave me a lot of instructions not only in knowledge but also in life.

I would also like to thank to all my dear friends in TU Delft and my dear parents in China. They give me selfless support and always like to help me when I am in trouble.

Delft, August 2011
Yinglei Yu

Delft University of Technology

MSc THESIS APPLIED MATHEMATICS

"A research on the optimization of the reservoir model"

Yinglei Yu

Delft University of Technology

Daily supervisor	**Responsible professor**
Dr. Remus Hanea	Prof. A.W.Heemink

Other thesis committee members

Dr. D. Kurowicka

March, 2007 Delft, the Netherlands

Contents

1. Introduction

These years it is more demanding to increase the production from petroleum reservoirs as a sequence of resource shortage. From 1950s, the numerical reservoir simulation became applicable to solve the complicated filtration problem in petroleum reservoir, which can provide a scientific reference to exploit oil underground efficiently, and this thesis provides an efficient way to achieve the optimal production through a model-based dynamic optimization of well placements and their production settings.

1.1 Demands for oil

After more than 150 years when it was discovered for the first time, oil continues to play an essential role in the global economy. While it remains the top source of energy, oil has fallen off its pedestal since the energy shocks of the 1970s and 1980s, which proved how reliant the developed world had become on petroleum products, and how vulnerable it was to shortfalls in supplies. The figure below from EIA gives the forecast of demand of oil and other sources next 20 years, which implies that there is still a very urgent requirement to promote the production of petroleum reservoirs. The Exploration and Production industry is struggling to keep up with this demand.

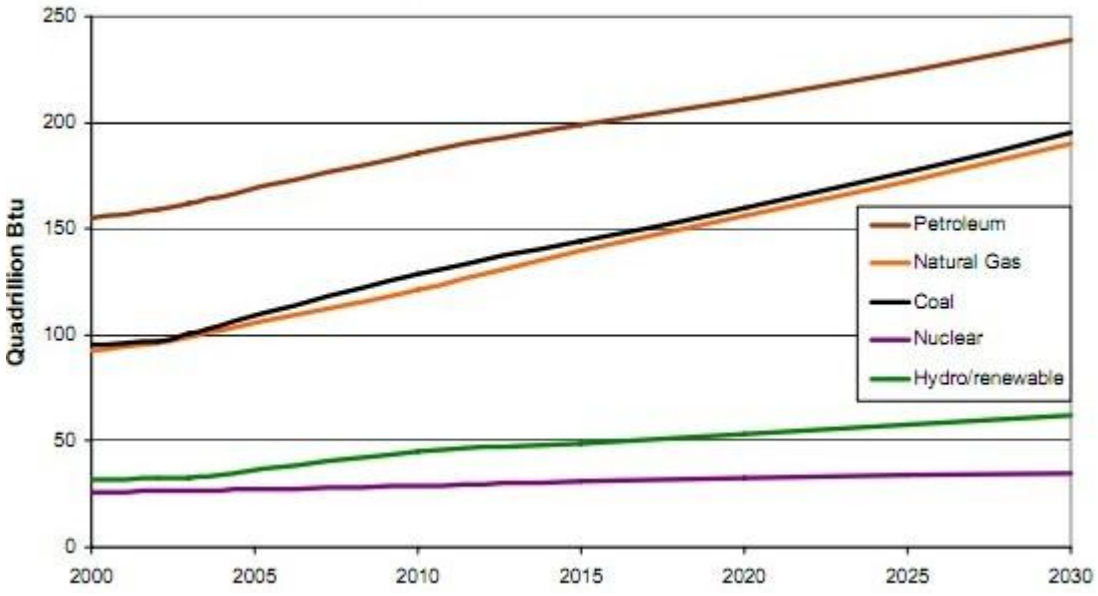


Figure 1.1 the forecast of global demand of energies(" Energy Projections 2006 – 2030",EIA)

A brief introduction of scheme and history of E&P process is given as follows.

1.2 Reservoir management

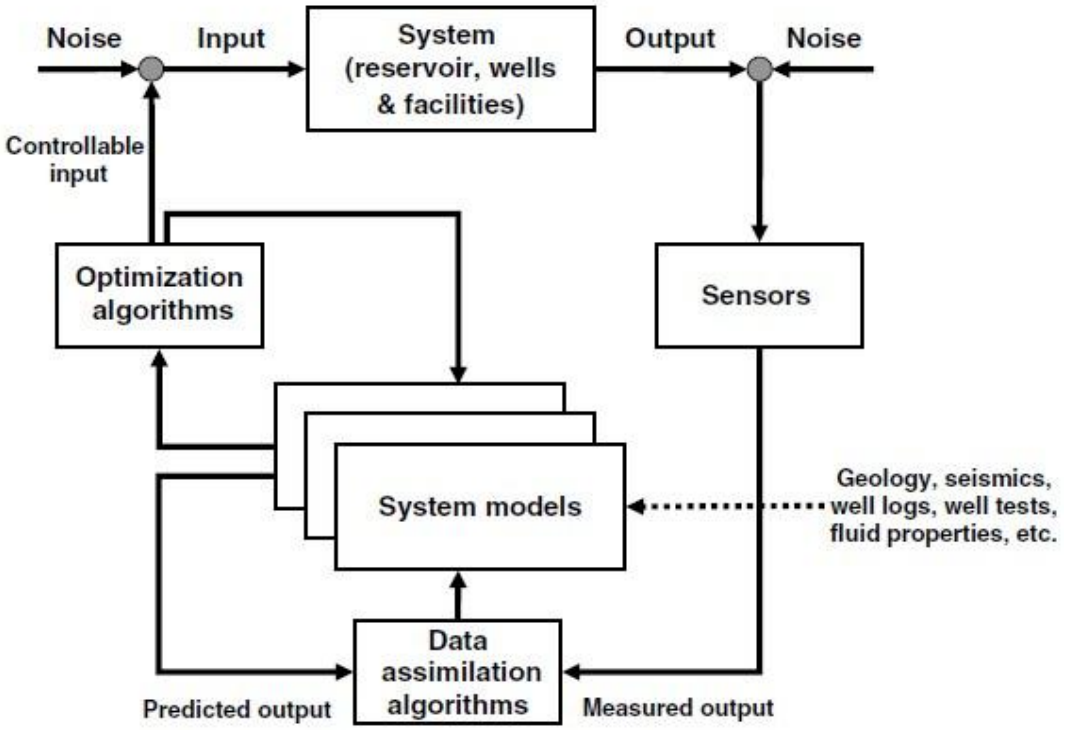


Figure 1.2 reservoir management depicted as a model-based controlled process(Jansen et al(2005))

The figure above describes the reservoir management as a model-based controlled process. Since the subsurface is very heterogeneous and the parameters relevant to flow are correlated at different length scales, the uncertainties in the model parameters of the subsurface part are very large. It is therefore necessary to construct multiple subsurface models to simulate the low of fluids according to different geological conditions. The table 1.1 shows the related terminology.

Name	Description
inputs	Decisions on wells, surface facilities, infrastructure
outputs	Oil and gas production
State variables	Fluid pressure and saturations
parameters	Fluid and geological properties

Table 1.1

It is possible to make a decision for the oil recovery process, which concerns determining the location and number of injection and production wells or the optimal water injection and oil production flow rates over the whole life the reservoir.

In the process of oil production, measurements are performed at the top of the well and in the facilities and gives an indication of pressures and phase rates in the

surface part of the production system. In the past, these measurements are performed monthly or quarterly without a good accuracy. During these years, after introducing an increasing amount of sensors into the reservoir system, the measurements become nearly continuous both at surface and subsurface to some extent. Moreover, there are also other measurement techniques emerging that can give a depiction of the changes in reservoir pressure and fluid saturation between the wells. With the aid of systematic algorithms for data assimilation, it is possible to adjust the parameters of individual grid blocks of the numerical models to some extent so that the simulated response matches the measured data better. In further discuss, it is hopeful to give a well forecast of the future system response.

As a result, the optimization process is model-based, since reservoir models are used to measure the effect of field development and production operation decisions on future hydrocarbon production.

1.3 Model-based optimization

1.3.1 Optimal control of production system settings

Given a reservoir model and a certain configuration and fixed location of wells, optimal control of production settings is to find the time-varying production settings over a lifecycle of the reservoir that maximize the recovery factor and then the production. There are several methods to compute the solutions to optimal control problems involving nonlinear systems and non-quadratic performance measures, such as the gradient, simultaneous, shooting or dynamic programming methods (Bryson and Ho(1975),Srinivasan et al.(2002b)). Among these methods, only the gradient method is applicable when dealing with extremely large number of reservoir model states. The idea of gradient method is to iteratively improve upon an initial guess of the optimal control using a gradient-based method until a local optimal solution is reached. Conceptually, the easiest approach is to approximate each individual component of gradient by finite differences, however, it is demanding since each approximation requires an evaluation of the performance measure, which in turn needs a reservoir simulation. So the available approach is to compute the gradient using a so-called adjoint model.(Kirk(1970),Stengel(1986).)

There are many applications of adjoint-based optimization of production settings in the petroleum engineering literature. Some of the earliest ones are by Ramirez and co-workers , summarized in Ramirez(1987), who considered tertiary recovery techniques.

This was quickly followed by Asheim(1987), Asheim(1988),Virnovsky(1991),Zakirov et al(1996), and Sudaryanto and Yortsos(2000) who considered secondary recovery techniques/ Although the type of production setting differ(e.g. from concentrations of injected chemicals to water injection rates), they are all applications of the same technique: gradient-based optimization with gradients computed using an adjoint model. In this respect, it is interesting that the method had not received significant

attention until Brouwer and Jansen(2004) demonstrated the possibility to significantly increase the recovery factor using smart wells. Numerous applications emerged then, several of which involve the particularly difficult problem of including state constraints - see Sarma et al. (2006a), de Montleau et al. (2006), and Kraaijevanger et al. (2007). Since state constraints (e.g. bounds on the reservoir pressure or the amount of produced water) are particularly important in production operations, state constraint handling is a topic of ongoing research. Another relevant open issue is the shape of optimal solutions: Sudaryanto and Yortsos (2000) and Sudaryanto and Yortsos (2001) state that these will sometimes be of the bang-bang (i.e. on-off) type, which have an obvious advantage over smooth solutions in that they can be implemented with simple on-off valves. Interestingly, this statement is supported by some, but not all, applications in Brouwer (2004) and Brouwer and Jansen (2004). In other words, it is unclear why and under what conditions optimal production setting problems can be expected to have bang-bang optimal solutions. This is important, because variable-setting valves are much more expensive than simple on-off ones.

1.3.2 Optimization of well placements

During the modeling process, reservoirs are essentially divided into a finite number of 'grid blocks', the properties of which are assumed to be homogeneous. Wells are then simply source or sink terms (depending on whether they produce or inject fluids) into or from certain of these grid blocks. Optimization of the well trajectory and its location is thereby an integer problem - Kosmidis et al. (2005), Bangerth et al. (2006). For example, if a single well is to be placed in 1 out of N grid blocks, the problem clearly involves N discrete possible choices. Determining the number of wells is also clearly an integer problem, and the combination with the optimization of production settings leads to a mixed-integer nonlinear problem, or MINLP. MINLP's also frequently arise in the chemical process industry, and there are several methods to deal with them - see Kallrath (2000). Most of these methods, however, require far too many evaluations of the performance measure to be applicable to reservoir models. In practice, well optimization is therefore mostly done manually, although there are several publications on automatic well optimization. These applications can be broadly classified into local or global optimization methods. Local optimization methods try to iteratively improve upon an initial well configuration, much as in the previous optimization of production settings, until a local optimal solution is reached. The main challenge in this application, again as in the optimization of production settings, is to effectively find improving directions3 in which to alter the well configuration. Global methods, on the other hand, will sometimes tolerate lower performance measures in the hope of finding the global, as opposed to local, optimal solution. There are many applications of global methods to the well optimization problem: Beckner and Song (1995) applied simulated annealing, Centilmen et al. (1999) neural networks, Bittencourt and Horne (1997),Montes et al. (2001) and Aitokhuehi et al. (2004) genetic

algorithms, and Yeten (2003) a combination of the latter two. Although these applications have the virtue of simplicity (a global optimization algorithm of choice is coupled with a reservoir simulator to evaluate the performance measure), they generally require many reservoir simulations to converge to an adequate solution. Bangerth et al. (2006) compares two local methods for optimizing the location of vertical wells in a 2D reservoir model. The first one is the Finite Difference Gradient (FDG) method, which as the name suggests tries to find improving directions by perturbing each well location by one grid block in each direction. This has the obvious drawback of requiring m + 1 reservoir simulations to compute an improving direction of m to-be-placed wells. The second method is the simultaneous perturbation stochastic approximation (SPSA) method of Spall (1992), which basically chooses a random direction to alter the wells and, if this does not yield an improvement in the performance measure, assumes that the opposite direction will. The obvious advantage is that an improving direction is almost always found in at most 2 reservoir simulations, with the disadvantage that this direction is generally far from the 'steepest' one. In other words, an efficient method to find (almost) steepest improving directions using a limited number of reservoir simulations is currently lacking.

1.4 Problem Formulation

In this thesis, we suppose a 5-spot (1 injection well, 4 production wells) reservoir system, and detect an efficient tool for dynamic optimization of well locations and their production settings to maximize the recovery factor of petroleum reservoirs based on uncertain reservoir models.

Steps to solve the problem:

Step1. Set the reservoir model

Formulate the reservoir models as well as the production constraints used throughout the thesis.

Step2. Applying data assimilation with EnKF to petroleum reservoir model

Step3. Optimal control of production settings

Investigate the robust optimization of production settings to reduce the negative effect of model uncertainty given a fixed well location.

Step4. Optimal well placement

Investigate how we can effectively find optimal well locations when we fix the Bottom Hole Pressure(BHP) and flow rates of injection and production.

Step5. Combination of the two optimizations and case study

Step6. Conclusions and recommendations

Reach a conclusion that is useful for further study of the optimization methods for reservoir models.

2. Reservoir Model

Mathematical models of petroleum reservoirs have been applied since the late 1800s, which contains a series of equations that describe the flow of fluids in the reservoir. In this section we discuss the black oil formulation that contains only oil and water in the reservoir system without considering several important physical aspects such as gravity, capillary pressure and the presence of an aquifer. The application of finite difference methods to fluid flow in porous media that we present in this chapter is mostly based on the textbook by Peaceman(1977) and Aziz and Settari(1979).

2.1 The black oil formulation

2.1.1 Mass balance equations

The equations as follows give a brief overview of water(w)-oil(o) flow according to the mass balances.

$$\frac{\partial}{\partial t}(\phi \rho_i S_i) = -\nabla \bullet (\rho_i \bar{u_i}) + q_i, i \in \{o, w\} \tag{2.1}$$

where t denotes time, ϕ the porosity, ρ_i the density of phase i, S_i the saturation of phase i, $\nabla \bullet$ the divergence operator, $\bar{u_i}$ the superficial velocity and the source/sink terms q_i is called the Neumann boundary conditions. In this model, we assume there is no flow across the boundaries of the reservoir geometry over which (2.1) has defined, other than through the source/sink terms.(the so called "Neumann boundary conditions")The superficial velocity can be computed by the semi-empirical Darcy's equation(Muskat(1937), Hubbert(1956)):

$$\bar{u_i} = -k \frac{k_{ri}}{\mu_i} \nabla p_i, i \in \{o, w\} \tag{2.2}$$

and p_i is the pressure of phase I, k the permeability, k_{ri} the relative permeability, and μ_i the viscosity of phase i. Generally the relative permeability are highly nonlinearly dependent on the saturation and in an interval [0,1]. The figure below depicts a typical curves of k_{ro}, k_{rw}.

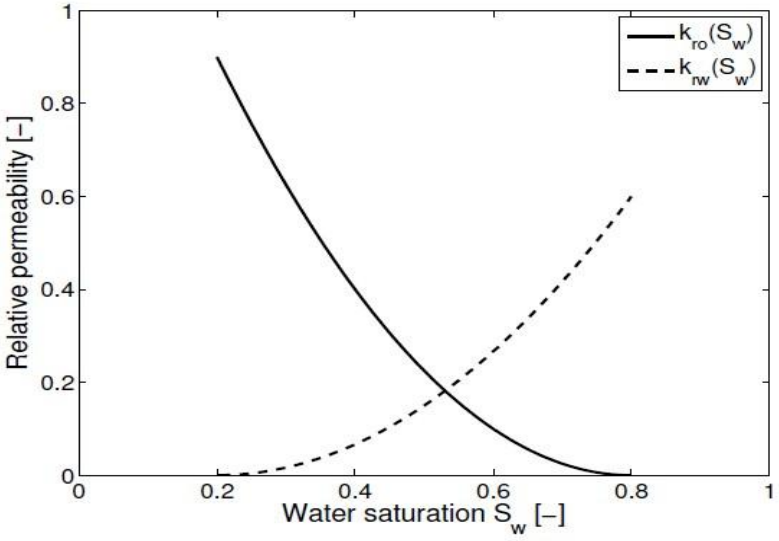


Figure 2.1 typical curves of relative permeability on saturations

From the assumption that there are only 2 phases and that we ignore the capillary pressure which dependent on water saturation, we have the following functions:

$$S_o + S_w = 1 \tag{2.3}$$

$$p_w = p_o \tag{2.4}$$

We take water saturation and oil pressure according to the common practice in reservoir simulation and lead to the following PDE's:

$$\begin{aligned} \frac{\partial}{\partial t}(\phi\rho_o(1-S)) &= \nabla\bullet(k\frac{k_{ro}}{\mu_o}\rho_o\nabla p) + q_o \\ \frac{\partial}{\partial t}(\phi\rho_w S) &= \nabla\bullet(k\frac{k_{rw}}{\mu_w}\rho_w\nabla p) + q_w \end{aligned} \tag{2.5}$$

2.1.2 Discretization of state vector

In order to solve (2.5), we need to pay attention that the geological properties in oil reservoirs generally vary significantly over space, which is so-called "heterogeneous". It cannot be solved analytically, but numerically instead. In this case, it is necessary to make a spatial discretization by dividing the reservoir area into a finite number of grid blocks with homogeneous geological properties. We denote the state vector with N grid blocks by x:

$$x := [p^T \quad s^T]^T \tag{2.6}$$

$$p := [p^1, p^2, ..., p^N] \tag{2.7}$$

$$s := [s^1, s^2, ..., s^N] \tag{2.8}$$

where p and s denote the oil pressure and water saturation in all grid blocks.

We have the following nonlinear equations:

$$\begin{bmatrix} V_{wp}(s) & V_{ws} \\ V_{op}(s) & V_{os} \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{s} \end{bmatrix} + \begin{bmatrix} T_w(s) & 0 \\ T_o(s) & 0 \end{bmatrix} \begin{bmatrix} p \\ s \end{bmatrix} = \begin{bmatrix} q_w \\ q_o \end{bmatrix} \tag{2.9}$$

where V and T are matrices containing accumulation terms and transmissibility terms which are functions of s, and q_w, q_o regard to the flow rate(water and oil) that can be controlled separately.

For example, grid block (i,j) contains flow rates with units in m^3 / s:

$$\begin{aligned} q_w^T &\triangleq [...(q_w)_{i,j}, ...], \\ q_o^T &\triangleq [...(q_o)_{i,j}, ...], \end{aligned} \tag{2.10}$$

The accumulation and transmissibility matrices are:

$$\begin{aligned} V_{wp} &\triangleq V(c_w + c_r)[0, ...0, \phi_{i,j} \times (S_w)_{i,j}, 0, ..., 0]; \\ V_{op} &\triangleq V(c_w + c_r)[0, ...0, \phi_{i,j} \times (1 - S_w)_{i,j}, 0, ..., 0]; \end{aligned} \tag{2.11}$$

$$\begin{aligned} V_{ws} &\triangleq V[0, ..., 0, \phi_{i,j}, 0, ..., 0]; \\ V_{os} &\triangleq -V[0, ..., 0, \phi_{i,j}, 0, ..., 0]; \end{aligned} \tag{2.12}$$

where V denotes the volume of a grid block.

$$T_o \triangleq \left[-(T_o)_{i,j-\frac{1}{2}}, ... -(T_o)_{i-\frac{1}{2},j}, ((T_o)_{i,j-\frac{1}{2}} + (T_o)_{i-\frac{1}{2},j} + (T_o)_{i+\frac{1}{2},j} + (T_o)_{i,j+\frac{1}{2}}), -(T_w)_{i+\frac{1}{2},j}, ..., -(T_w)_{i,j+\frac{1}{2}} \right]; \tag{2.13}$$

and $$(T_w)_{i,j} \triangleq \frac{\Delta y}{\Delta x} \frac{h}{\mu_w} (k k_{rw})_{i-\frac{1}{2},j} \tag{2.14}$$

$$(k_{rw})_{i+\frac{1}{2},j} \triangleq \begin{cases} (k_{rw})_{i,j} & if \quad p_{i,j} \geq p_{i+1,j} \\ (k_{rw})_{i+1,j} & if \quad p_{i,j} < p_{i+1,j} \end{cases}, \tag{2.15}$$

where h, Δx, Δy denotes the height, length, width of a grid block.

For convenience, we would like to rewrite (2.9) as:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} A_1(x(t)) & 0 \\ A_2(x(t)) & 0 \end{bmatrix} x(t) + \begin{bmatrix} B_1(x(t)) \\ B_2(x(t)) \end{bmatrix} q(t); \\ x(0) &= x_{initial} \end{aligned} \tag{2.16}$$

with $A(t) = \begin{bmatrix} A_1(x(t)) & 0 \\ A_2(x(t)) & 0 \end{bmatrix} = -\begin{bmatrix} V_{wp} & V_{ws} \\ V_{op} & V_{os} \end{bmatrix}^{-1} \begin{bmatrix} T_w & 0 \\ T_s & 0 \end{bmatrix}$ (2.17)

$$B(t) = \begin{bmatrix} B_1(t) \\ B_2(t) \end{bmatrix} = \begin{bmatrix} V_{wp} & V_{ws} \\ V_{op} & V_{os} \end{bmatrix}^{-1} \tag{2.18}$$

$$q(t) = \begin{bmatrix} q_w(t) \\ q_o(t) \end{bmatrix} = \begin{bmatrix} F_w \\ F_o \end{bmatrix} q_t \tag{2.19}$$

If an injection well is located in grid block i, we can directly control the source terms q_o^i and q_w^i (in $kg/m^3 s$). Only water is injected so the oil flow rate is 0.

$$\begin{aligned} q_o^i &= 0; \\ q_w^i &= \frac{\rho_w(p^i)}{v^i} q^i, i \in N_{inj} \end{aligned} \tag{2.20}$$

If a producer well is perforated in grid block j, then we can indirectly control the flow rates as:

$$\begin{aligned} f_w^j &:= \frac{\frac{k_{rw}(s^j)}{\mu_w}}{\frac{k_{rw}(s^j)}{\mu_w} + \frac{k_{ro}(s^j)}{\mu_o}} \\ q_o^j &= \frac{\rho_o(p^j)}{v^j}(1 - f_w^j) q^j; \\ q_w^j &= \frac{\rho_w(p^j)}{v^j} f_w^j q^j, i \in N_{prod} \end{aligned} \tag{2.21}$$

Here, f_w^j is the fractional flow rate of water, N_{inj}, N_{prod} denote the number of injection well and production wells separately.

2.1.3 Time discretization

To solve an equation as (2.16) by computer, it is common to make a discretization of time t and then do the analysis. Here we use the first-order Euler scheme to discrete by t as:

$$\dot{x}(t) \approx \frac{x((k+1)\Delta t) - x(k\Delta t)}{\Delta t} \tag{2.22}$$

where Δt denotes the time step size.

We can get:

$$p_{k+1} = (I + A_1\Delta t)\, p_k + B_1 \Delta t q_k,\ p_0 = p_{init}; \tag{2.23}$$

$$s_{k+1} = (I + A_2\Delta t) s_k + B_2 \Delta t q_k, s_0 = s_{init}; \tag{2.24}$$

where $q_k = q(k\Delta t)$,and the time discretization step is usually set as follows to get a more accurate capture of all dynamics, which is named as "Nyquist-Shannon sampling time".(Astrom, Wittenmark(1990)).

$$\Delta t = \frac{0.5}{\max(|\lambda_{\min}(A_1)|, |\lambda_{\min}(A_2)|)} \tag{2.25}$$

2.2 The well model

2.2.1 General equation

For the case of prescribed bottom hole pressures and well flow rates, the flow rates q at time t in the grid block of the production wells where we want to prescribe the pressure have a general formulation as:

$$q_t = J_p(\hat{p}_{well} - p) \tag{2.26}$$

where J_p is called well index or production index denoting the well geometric factors and geological factors like rock and fluid properties (permeability and saturation ,etc) around the well.

$\hat{p}_{well}$ is the prescribed bottom hole pressure, and p is the grid block pressure. For convention followed by the statements above, a negative flow rate indicate production here.

For the grid block of the injection wells where we prescribe the injection rates, similarly we have the relative formulation as:

$$q_t = \hat{q}_{well} \tag{2.27}$$

where $\hat{q}_{well}$, a positive value, denotes the prescribed injection rates of the injection wells.

In addition, we can also compute the bottom hole pressure in the wells where the flow rates are prescribed , in this case, we need a diagonal matrix of well indices Jq.

$\hat{q}_{well} = J_q(p_{well} - p)$ and thus we obtain: $p_{well} = p + J_q^{-1}\hat{q}_{well}$ (2.28)

In this thesis, through injection wells generally only water is injected, thus the

amount of water injected can be controlled directly. In a production well, however, the amount of the oil and water that will be produced are influenced by their flow around the well and by pressure.

2.2.2 State space representation

Concerning (2.9), we can rewrite the equation as:

$$\begin{bmatrix} V_{wp}^*(s) & V_{ws}^* \\ V_{op}^*(s) & V_{os}^* \end{bmatrix} \begin{bmatrix} \dot{p}^* \\ \dot{s}^* \end{bmatrix} + \begin{bmatrix} T_w^*(s) + F_w^* J_p^* & 0 \\ T_o^*(s) + F_o^* J_p^* & 0 \end{bmatrix} \begin{bmatrix} p^* \\ s^* \end{bmatrix} = \begin{bmatrix} F_w^* \\ F_o^* \end{bmatrix} (I_q^* + J_p^*) q_{weill}^* \tag{2.29}$$

where the input vector of the prescribed terms is:

$$q_{well}^* = \begin{bmatrix} 0 \\ \hat{q}_{well} \\ \hat{p}_{well} \end{bmatrix} \tag{2.30}$$

,and the other two factors in (2.26) are defined below:

$$I_q^* = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad J_p^* = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & J_p \end{bmatrix} \tag{2.31}$$

To bring equations above in state space form, we define the input vector as u and the state vector as x:

$$u^* = \begin{bmatrix} \hat{q}_{well} \\ \hat{p}_{well} \end{bmatrix}, \qquad x^* = \begin{bmatrix} p \\ s \end{bmatrix} \tag{2.32}$$

and the location matrix as:

$$L_{q^* u} = \begin{bmatrix} 0 & 0 \\ I & 0 \\ 0 & I \end{bmatrix} \tag{2.33}$$

We denote y^* as the output vector, in our case, we extend it to include the prescribed pressure and flow rates.:

$$y^* \triangleq \begin{bmatrix} \bar{p}_{well} \\ \bar{q}_{well,w} \\ \bar{q}_{well,o} \\ \hat{p}_{well} \\ \hat{q}_{well,w} \\ \hat{q}_{well,o} \end{bmatrix} \tag{2.34}$$

Then we have equations in state space form as:

$$\dot{x}^* = A^* x^* + B^* u^* \tag{2.35}$$

$$y^* = C^* x^* + D^* u^* \tag{2.36}$$

where

$$A^* = -\begin{bmatrix} V^*_{wp} & V^*_{ws} \\ V^*_{op} & V^*_{os} \end{bmatrix}^{-1} \begin{bmatrix} T^*_w + F^*_w J^*_p & 0 \\ T^*_o + F^*_o J^*_p & 0 \end{bmatrix}, B^* = -\begin{bmatrix} V^*_{wp} & V^*_{ws} \\ V^*_{op} & V^*_{os} \end{bmatrix}^{-1} \begin{bmatrix} F^*_w \\ F^*_o \end{bmatrix} (I^*_q + J^*_p) L_{q^*u}$$

$$C^* = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & -F^*_{w,33} J^*_p \\ 0 & 0 & -F^*_{o,33} J^*_p \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad D^* = \begin{bmatrix} J_q^{-1} & 0 \\ 0 & F^*_{w,33} J_p \\ 0 & F^*_{o,33} J_p \\ 0 & I \\ F_{w,22} & 0 \\ F_{o,22} & 0 \end{bmatrix} \tag{2.37}$$

2.2.3 Well settings constraints

It is common sense that in reality, there are constraints of the input values concerning the limits of the ability of the surface facilities. Water injection wells are often operated on pressure constraints to avoid the fracturing of the formation around the wells. Production wells are often constrained to operate at a tubing head pressure above a certain minimum, which is determined by the working pressure of the first separator and some additional pressure to transport the liquid through the tube. Although in practice, the situation is more complicated since the constraints may change because the changes in the reservoir pressure, in reservoir simulation we can prescribe the pressures and flow rates between a minimum and a maximum. So it will be reflected by mathematical formulation as:

$$u_{\min} \le u(t) \le u_{\max}, \; for \forall t \tag{2.38}$$

2.3 Computational aspects

Equations (2.31) and (2.32) can be implemented in a simple two-dimensional, two-phase MATLAB simulator "simsim", which can simulate the flow in a horizontal rectangular reservoir. We discuss some numerical aspects(two-phase system) as follows:

(1) Most of the matrices we concern so far are sparse: the accumulation sub-matrices are diagonal, the transmissibility sub-matrices are penta-diagonally banded with two sub-diagonals with two diagonals, and the fractional flow and well index sub-matrices are sparse. Most of the elements in matrices are 0 and thus can save the computational time and occupy less memory.

(2) The reordering of vector and matrix elements with permutation matrices is not very essential in the computer implementation. We may just use matrices with elements that correspond to the relevant state or input variables at the appropriate locations

(3) Computation of an element of a transmissibility sub-matrix corresponding to a specific grid block involves computing the transmissibility for flow to or from the four neighboring grid blocks. Therefore, assembly of the transmissibility matrices requires knowledge of the connectivity of the grid blocks.

(4) The elements in the two-phase state vector $x = [p^T \; s^T]^T$ have different physical dimensions and strongly varying magnitudes. If we express the pressure in Pa, they are in the order of $10^6 \sim 10^7$, however, the saturation values are between 0 and 1.This may influence the accuracy of the result because of the different magnitudes. We can avoid this problem by rescaling the elements of p and s in order to reduce the difference of magnitudes.

(5) In an injection well, we have $q_t = q_w$, and we expect that soon after injection has started the fractional flows for water and oil close to the injection will approach 0 and 1 respectively. However, before injection, the initial condition for the saturation is usually equal to the connate water saturation, which mean that fractional flows for water and oil are 0 and 1 respectively. In theory, it would then be impossible to ever inject water. This paradox is solved by a strategy implemented in simsim.

In next few chapters, we will base our experiments on simsim. For more details, it is better to refer to *Systems theory for reservoir management by* Jan-Dirk Jansen.

3 Data assimilation with EnKF and its diagnostics tool for sensitivity

3.1 Asynchronous data assimilation with EnKF

Before EnKF was applied in the reservoir history matching, Bayesian framework is usually used in the inverse problem of estimating parameters combined with state vectors in the model. The Ensemble Kalman Filter(EnKF) was introduced by Evensen(1994) for updating non-linear models. The EnKF uses a Monte Carlo approach for representing and evolving the joint probability density function for the model state and parameters, and it computes the recursive updates steps by introducing an approximation where only the first and second order moments of the predicted pdf are used to compute the update increments.

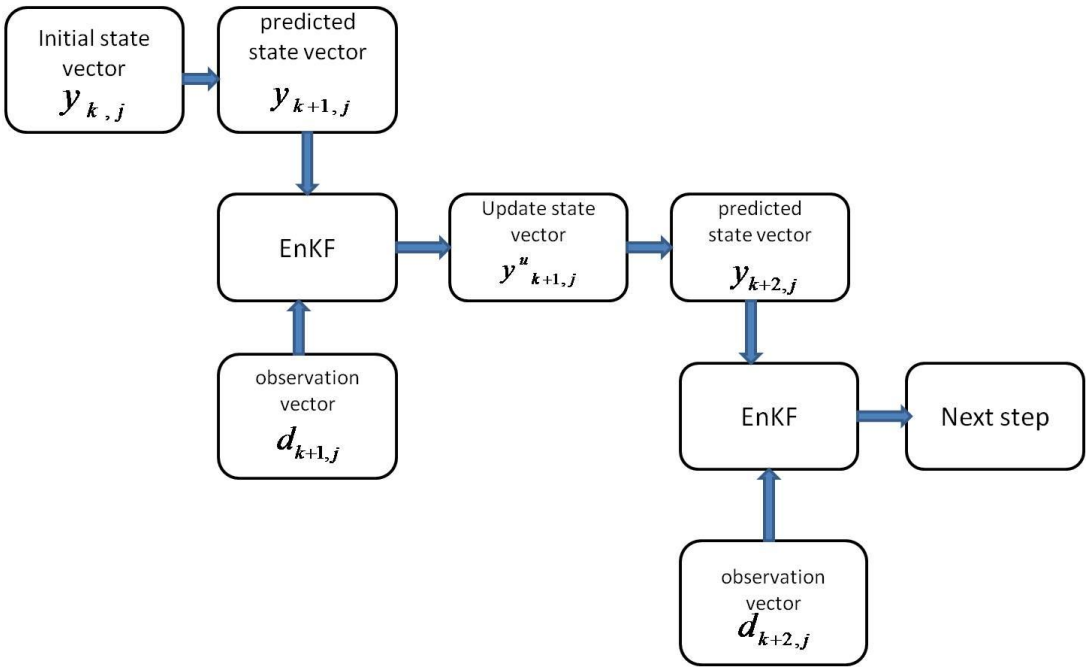


Figure 3.1 The procedure of Synchronous assimilation with EnKF

The figure above shows the process of Synchronous assimilation with EnKF, or “3D” assimilation, indicating that observations are assumed to be taken at the assimilation time. In the following section, we will introduce the Asynchronous data assimilation with EnKF used in this thesis, that is, observation vectors can be taken at the same time with measurement vectors. The $d_{k+1,j}, d_{k+2,j}$ in the above figure can be replaced by other time points(see Figure 2). The reservoir simulation model need

asynchronous data assimilation and EnKF turns out to work well.(P.Sakov et al,2010)

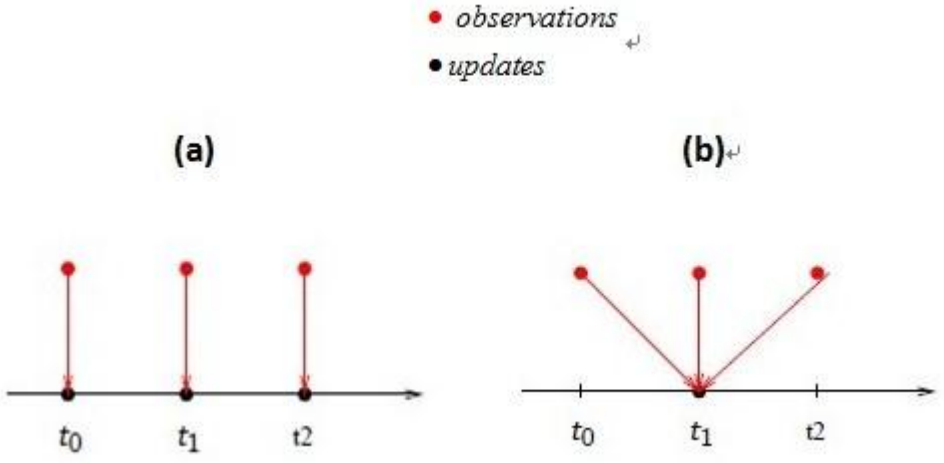


Figure 3.2 Synchronous data assimilation and asynchronous data assimilation

3.1.1 State vector formulation

In this thesis we settle state vectors combined state variables with parameters for history matching as:

$$(x_k)_{i,j} = \begin{pmatrix} p_k \\ s_k \\ \log(-\log(\phi)) \\ \log(k) \end{pmatrix}_{i,j} \tag{3.1}$$

Where p_k denotes the pressure at time k, s_k the saturation at time k, ϕ the porosity, k the permeability and thus $(x_k)_{i,j}$ denotes the state and parameter vector at time k in grid block (i,j). Consequentially, the total number of variables and parameters to be estimated is 4 times the number of active grid blocks.

3.1.2 State vector update with AEnKF

System model

The non-linear statistic model and measurement function are:

Measurement Function: $$x^f_{k+1\ j} = F(x^u_{k\ j,})$$ (3.2)

Observation Function: $$d_{k,j} = H_k x^f_{k,j} + v(k)$$ (3.3)

where $x_{k+1,j}^{f}$ mean the forecasting state vector at time k+1, j represents the jth realization of the ensemble, the ensemble size is N, and thus j=1,2,…,N. f denotes forecasting one and u denotes updated one. $x_{k,j}^{f}$ consists of N vertical vectors, $x_k^f = [x_{k,1}^f, x_{k,2}^f, x_{k,3}^f, ..., x_{k,N}^f]$. $d_{k,j}$ denotes the observation vector, and H_k is called "measurement vector", which links the observation vector and measurement vector. F() represents the relationship between the neighboring time points. v(k) is the observation derivation, which is a white noise.

Recurrence Formula

Measurement Covariance matrix: $$C_{x,j}^{f} = \frac{1}{N-1}(x_k^f - \bar{x}_k^f)(x_k^f - \bar{x}_k^f)^T \quad (3.4)$$

EnKF gain: $$K_k = C_{x,k}^f H_k^T (H_k C_{x,k}^f H_k{}^T + C_{d,k})^{-1} \quad (3.5)$$

Measurements update: $$x_{k,j}^u = x_{k,j}^f + K_k(d_{k,j} - H_k x_{k,j}^f) \quad (3.6)$$

Covariance matrix update: $$C_{x,k}^u = (I - K_k H_k) C_{x,k}^f \quad (3.7)$$

Here, $C_{x,k}$ is the covariance matrix of measurement error. N is the ensemble number. $\bar{x}_k^f$ is the average value of measurement vectors. K_k is the Kalman Filter at time k.The number of grid blocks is Ng, updated state vector contains porosity, permeability, and then every ensemble has 2Ng parameters. Water saturation and reservoir pressure belongs to the production data, and the number is N0. As a result, the state vector is a (2Ng+No)$\times$Ne matrix, and the observation vector is a N0$\times$Ne matrix. H_k is the observation factor, which is a Ne $\times$ (2Ng+N0) matrix, $H_k = [O_{Ne \times 2Ng} \mid I_{N_0 \times N_0}]$.

Here we can use asynchronous observations and run the simulation in the following sections.

3.1.3 Experimental setup and AEnKF results

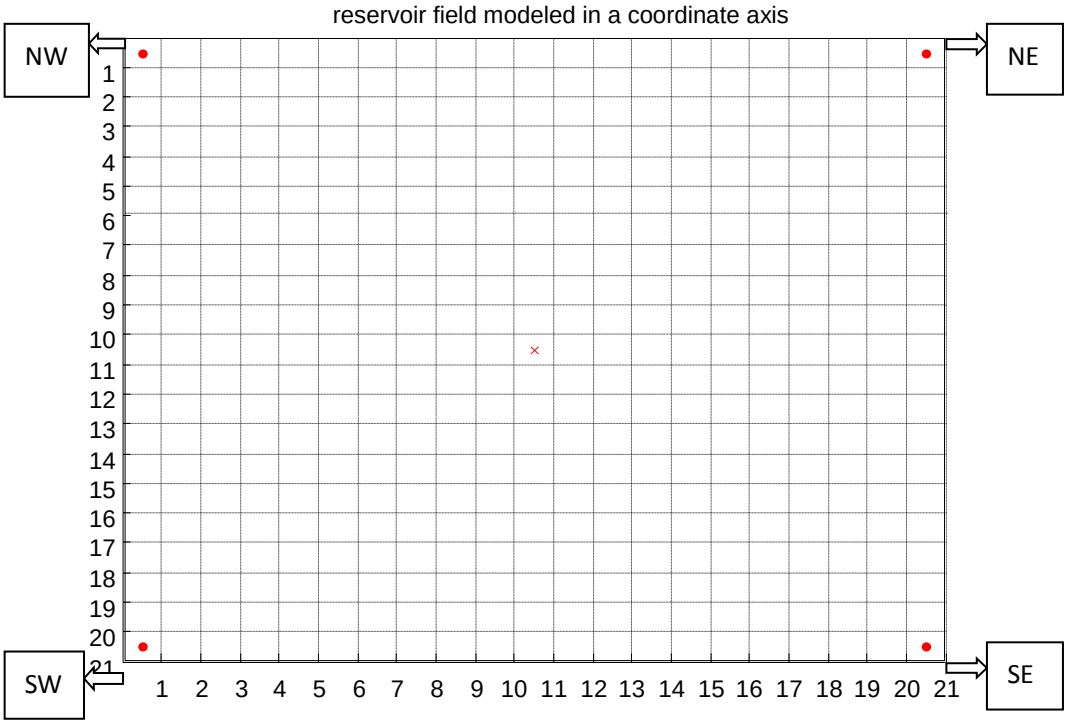
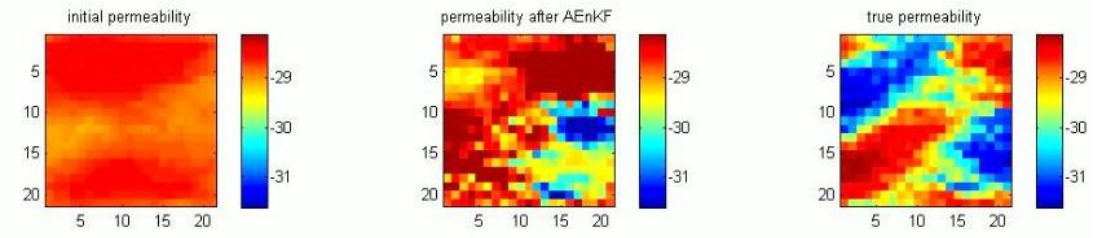


Figure 3.3 a 21 by 21 grids reservoir with fixed well locations

We set a twin experiment to describe 2D 2 phase (water-oil) flow. In this thesis, we consider a square reservoir of 700(m)×700(m)×2(m) vertically homogeneous, with heterogeneous permeability and porosity fields modeled with uniform Cartesian grid of 441(21×21) blocks, as we can see above, in which case we neglect gravity forces and capillary pressure.

Take the comparison of 30 and 60 ensembles for an example, we can see the initial, AEnKF, and true one of permeability and porosity of the reservoir separately.



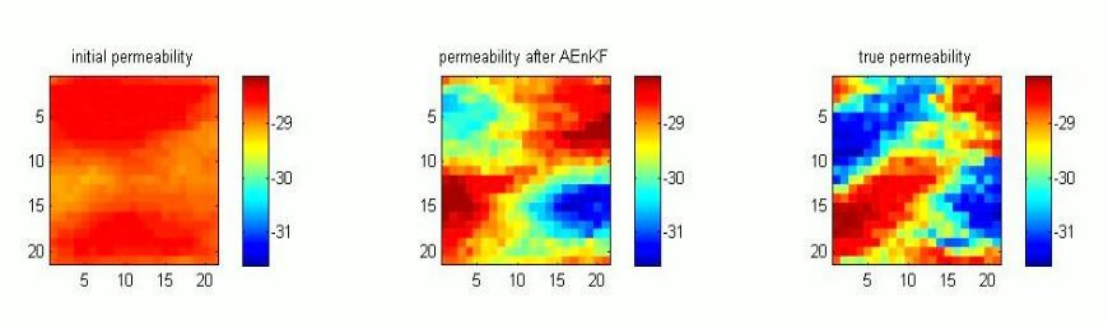


Figure 3.4 permeability(in log(-log())) after AEnKF compared to true one
(above:10 ensembles;down:60 ensembles)

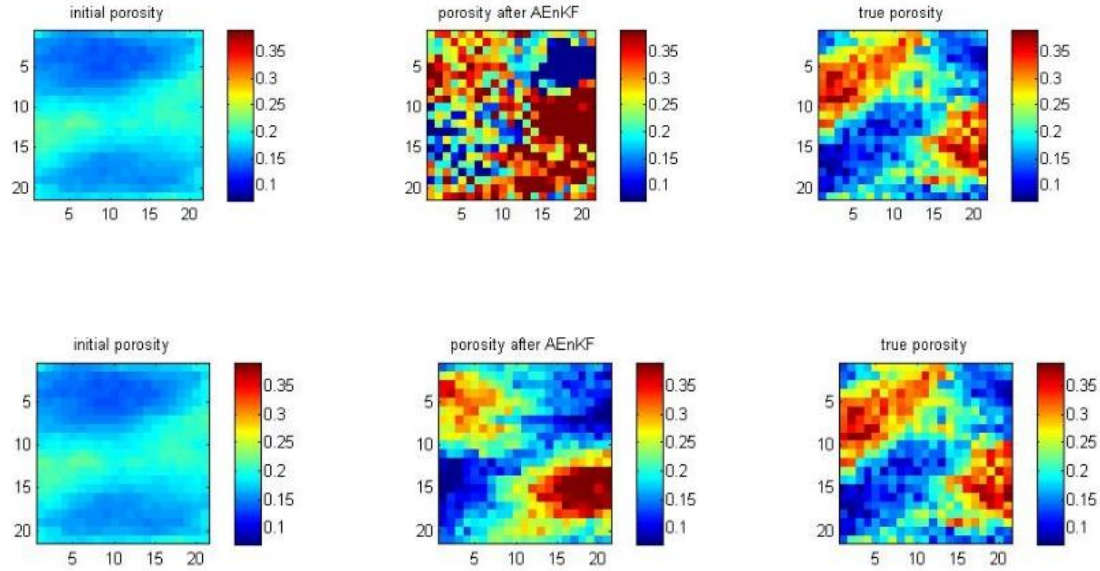


Figure 3.5 porosity(in log())after AEnKF compared to true one
(above:10 ensembles;down:60 ensembles)

From Figure 3 and Figure 4, we see that AEnKF performs a well simulation of the parameters in the reservoir model. However, obviously it is more close to the true distribution when there are 60 ensembles than 10. In the following section, we would like to do a research on what is influential to the history matching and how large is the influence.

3.2 Influence-matrix diagnostic of the data assimilation

The data assimilation with EnKF has already developed for years. Since the system is complicated, it is necessary to monitor the data assimilation (DA) process effectively. A set of complex measures is needed to indicate how different variables and parameters influence the process. The influence-matrix that we will regard as follows provides an effective tool to analyse the DA process.

3.2.1 Observational influence and self-sensitivity for DA with EnKF

Self-sensitivity

As mentioned above, the measurements update with EnKF can be written:

$$x^f = x^u + K(d - Hx^u) \text{, with } K = C_x^u H^T (HC_x^u H^T + C_d)^{-1} \quad (3.7)$$

where x^u denotes prior information, x^f estimate of the state, d the observations, K the Kalman filter, H the observational matrix, C_x^u measurements covariance matrix, H the observation factor.

Analysis estimate in the observation space:

$$\hat{d} = Hx^f = Hx^u + HK(d - Hx^u) = (I - HK)Hx^u + HKd \quad (3.8)$$

Sensitivity matrix:

$$S = \frac{\partial \hat{d}}{\partial d} = K^T H^T \quad (3.9)$$

According to (4.3), S_{ii} denotes the rate of change of estimate $\hat{d}_i$ with regard to variations in the corresponding measurement y_i.

Similarly, the analysis sensitivity w.r.t the background information is given by:

$$\frac{\partial \hat{d}}{\partial (Hx^f)} = I_p - K^T H^T = I_p - S \quad (3.10)$$

where p is the number of the observations. This is easy to understand, for example，if the self-sensitivity with respect to ith observation is S_{ii}, then the self-sensitivity with respect to the background is 1- S_{ii}.

Trace diagnostic

We define the globally averaged observation influence (GAI):

$$GAI = \frac{tr(S)}{p} \quad (3.11)$$

GAI gives a diagnostic of the total observational influence on the DA scheme and sequentially the average background global influence to the analysis at the

observation points is 1-GAI.

Another index of interest is the partial influence for any selected subset of data is given by:

$$PAI = \frac{\sum_{i \in I} S_{ii}}{p_I} \tag{3.12}$$

where p_I is the number of data in subset I. This shows the influence of a chosen group of data set to the analysis. We can use GAI and PAI as a diagnostic tool to analyze DA scheme with EnKF in this thesis.

3.2.2 Experiment and Results

In this section, we will do research on how large is the influence of the ensemble size, the prior and observational data on the data assimilation process.

The influence of the ensemble size

We do five independent experiments, and each one contains 10 to 100 ensembles. We will test the root mean square derivation (RMSE) of the parameters of the reservoir as permeability and porosity and GAI.

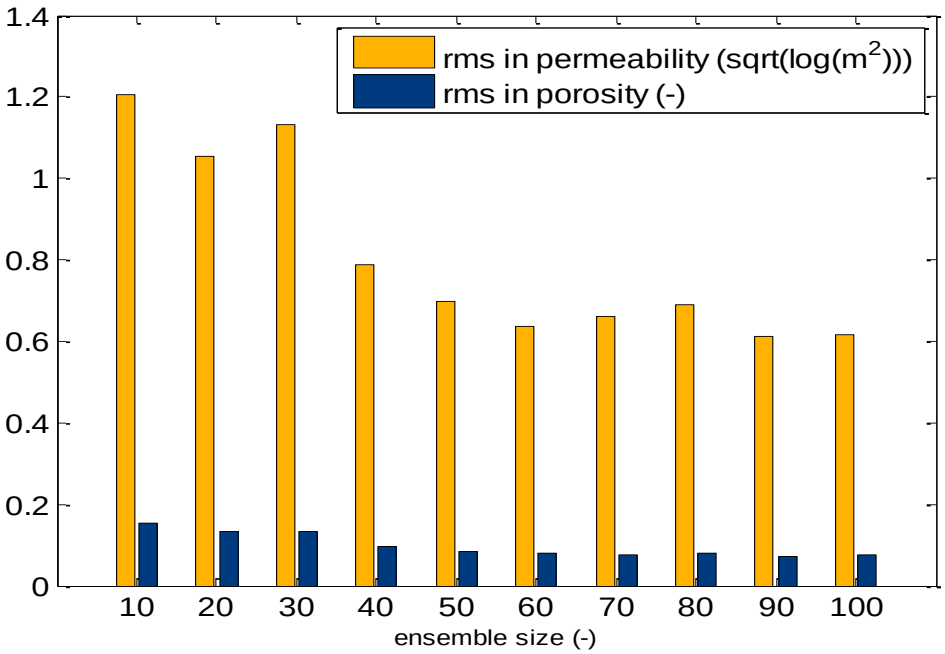


Figure 3.6 RMSE of permeability and porosity

As can be seen from the figure 5, when the ensemble size is larger than 60, the RMSE of both permeability and porosity is fluctuating around a relatively small value. However, when the ensemble size is large, it is time-consuming to get the result. Consequently, we can choose ensemble size as 60 to get the parameters more close

to the true one effectively.

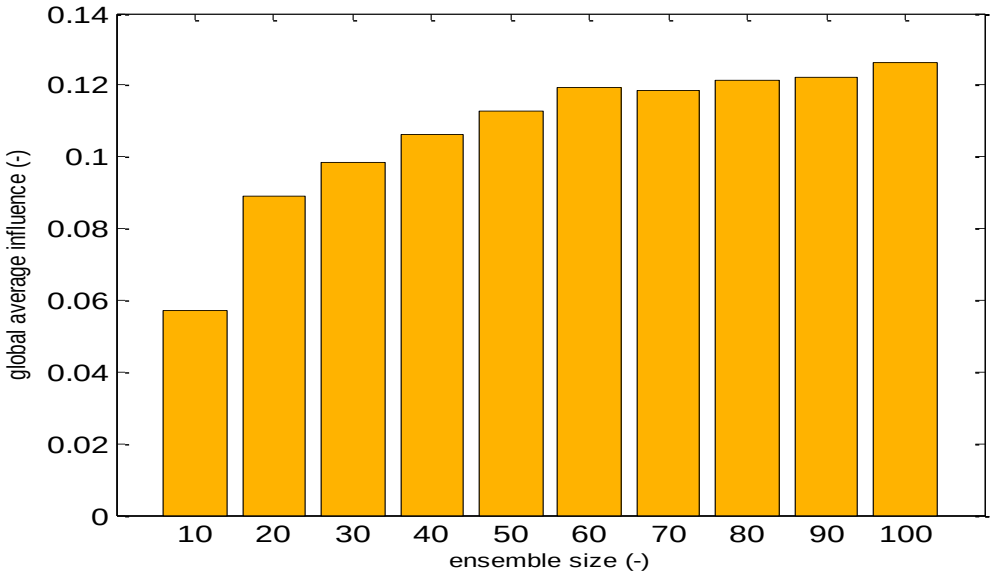


Figure 3.7 Global average influence to analysis of different ensemble size

According to Figure 6, it is not difficult to notice that when the ensemble size is larger than 60, the GAI will not change too much and reaches a relatively high value as 0.12, which means the observational data plays 12% influence of the data assimilation and the prior information 88%.

GAI of other variables

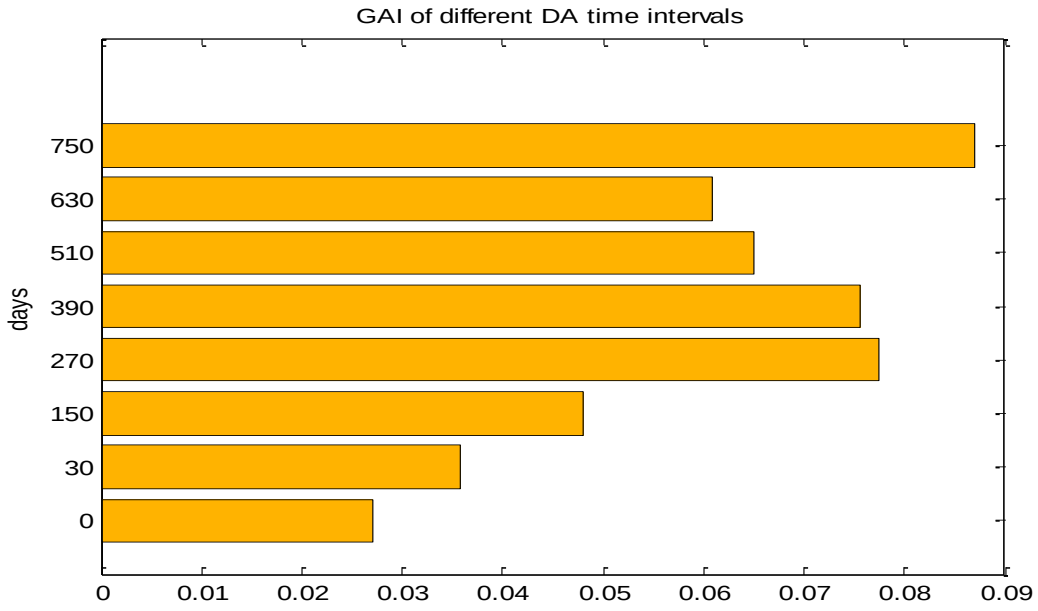


Figure 3.8 Global average influence of different time intervals

We choose an experiment of 60 ensembles and compute GAI with a time step of

120 days in order to find the rule of the observational data 's influence to the analysis. In figure 7, it shows the global average influence to the analysis doesn't react very sensitively to different time intervals, and after 270 days, it reaches a comparatively high GAI value. As a result, the history matching process becomes more sensitive to the observation data as time goes on.

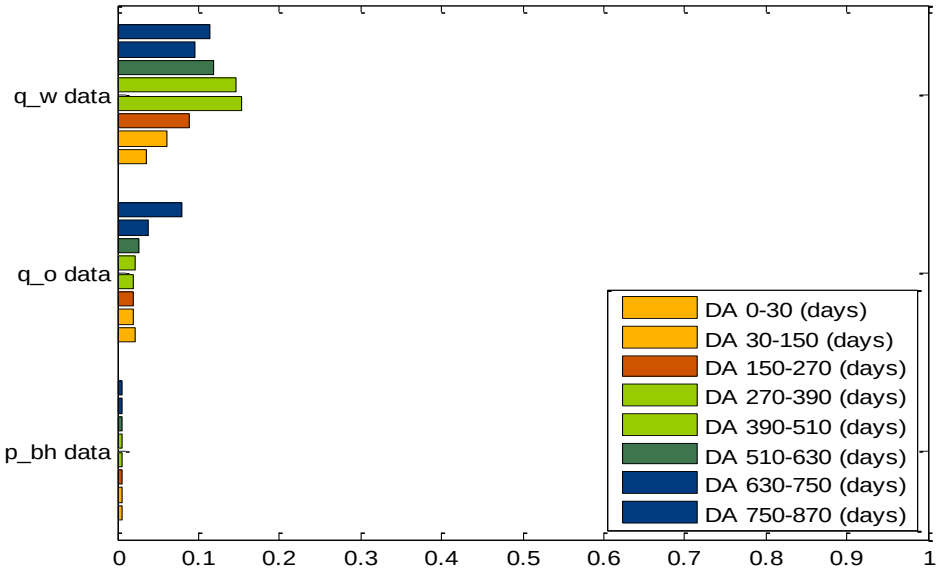


Figure 3.9 GAI of different measurement types

In Figure 8, it shows the DA sensitivity to different types of measurement, including bottom hole pressure of injection well and oil and water production rates of four production wells. From the graph, the DA is more sensitive to the measurement in the production wells.

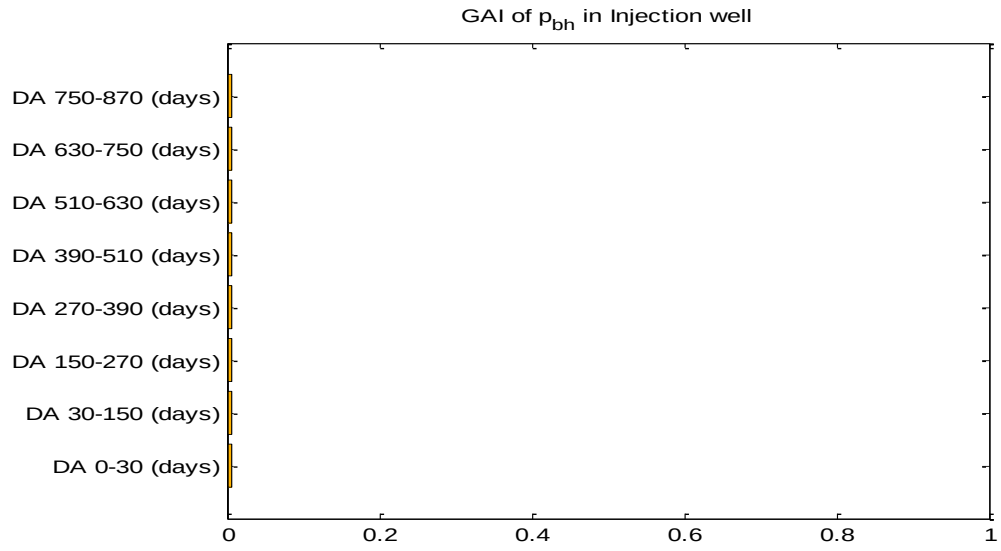


Figure 3.10 GAI of bottom hole pressure in injection well

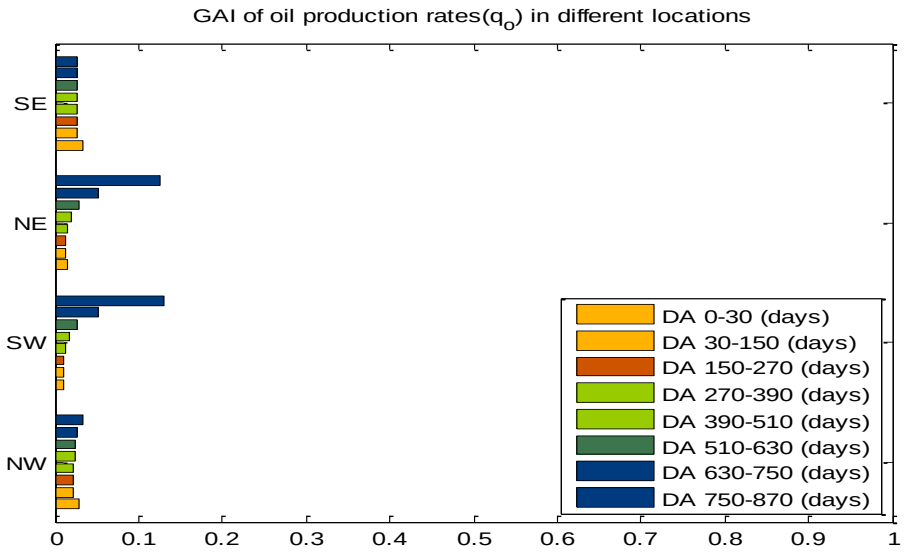


Figure 4 GAI of oil production rates in different well locations

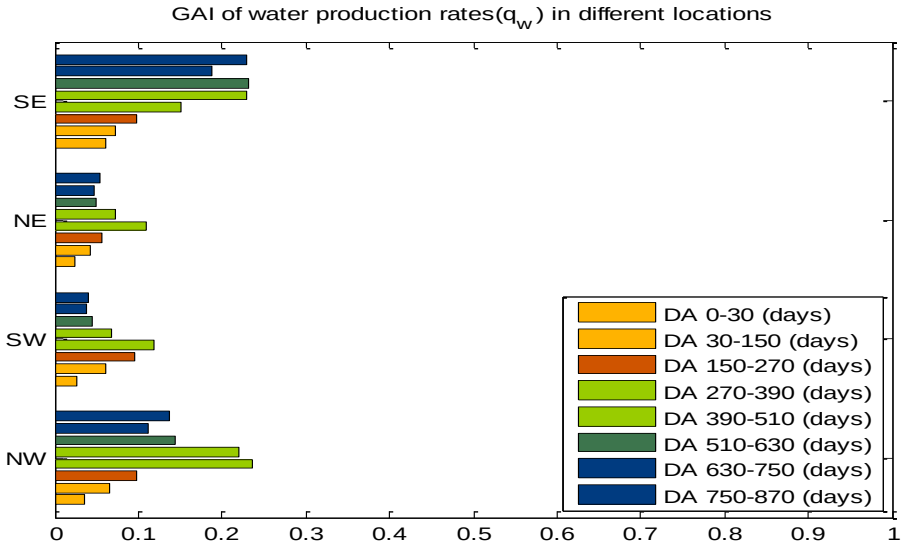


Figure 5.12 GAI of water production rates in different well locations

In Figure 9,10,11, they shows the data assimilation's sensitivity to the same measurement type in different well locations. Figure 9 indicates that the measurement of bottom hole pressure is not influential to DA. From Figure 10 and 11, we can see that the DA is more sensitive to North-east(NE) and South-west(SW) wells for oil production rates and more sensitive to South-east(SE) and North-west(NW) wells for water production rates.

In conclusion, for our model in this thesis, we choose the ensemble size as larger than 60, and get the global average influence as around 8.8% at the end of DA times. In addition, the DA is not sensitive to the measurement in injection well and shows different sensitivity to the production well locations referring to oil and water production rates.

4 Optimal control of well settings

The urgent need for the energy and the real-time reservoir management entail the optimization of the economic output under geological uncertainty, which can be achieved by control the well settings. In this section, we will refer to the procedure of dealing with production optimization and model updating along with it. We define the cost function as J, which is to be optimized. In this thesis, it is applicable to use Net Present Value(NPV) as J. Through controlling the well rates and bottom hole pressure, we can maximize J to get an optimal economic output.

4.1 Mathematical formulation

The optimization problem under uncertainty can be formulated as a mathematical model. It requires us to find set of controlled vector u, which contains bottom hole pressure and water and oil flow rates. We usually set the objective function as NPV, which is defined as the total oil revenue minus the total injection and production costs, combined with a discount factor d. Let r_{oil}, r_{wi}, r_{wp} denote the oil revenue, water-injection cost, water-production output per unit volume respectively, $q_{o,j,n}, q_{wp,j,n}, q_{wi,j,n}$ denotes the oil production rates, water production rates and water injection rates in well j at nth time step. Here, we denote an objective function to maximize:

$$J^n = \{-\underbrace{\sum_{j=1}^{n\_u} r_o q_{o,j,n}}_{\text{oil revenue}} - \underbrace{\sum_{j=1}^{n\_u} r_{wi} q_{wi,j,n}}_{\text{water injection cost}} + \underbrace{\sum_{j=1}^{n\_u} r_{wp} u_{wp,j,n}}_{\text{water disposal costs}}\} \frac{\Delta t}{(1+d)^{\frac{n\Delta t}{T}}} \quad (4.1)$$

, where n_u is the total number of wells, including production and injection wells.

We set injection rates positive and production rates negative and thus the oil revenue is positive and water injection cost and water disposal cost are negative. The objective function is positive, and we want to maximize it. In this thesis, we consider the optimal control problem with fixed terminal time and free terminal states. Optimal control is a way to find some input values like bottom hole pressure in production well and injection flow rate minimizing the cost function. Here, we denote a function that accumulates over time as:

$$J = \sum_{n=0}^{N-1} J^n(q(t)) = \sum_{n=0}^{N-1} J^n(x^n, u^n) \quad (4.2)$$

,where n is the time step to control the production setting and N is the number of the control steps. x^n and u^n are state vector and production setting at nth step of

time respectively.

Thus the problem can be formulated as follows:

$$\max_{u^n} J = \sum_{n=0}^{N-1} J^n(x^n, u^n) \tag{4.3}$$

The objective function is subjected to:

$$\begin{aligned} & g^n(x^{n+1}, x^n, u^n) = 0 \\ & x^0 = x_0 (initial\ condition) \\ & c^n(x^{n+1}, u^n) \le 0 \\ & Au^n \le b \\ & LB \le u^n \le UB \end{aligned} \tag{4.4}$$

It is a constrained optimization problem, where the constrained term is g^n,which are basically the reservoir simulation equations for each grid block at each time step.In Chapter 2, we have already got governing equation as:

$$\dot{x} = A(x(t))x + B(x(t))q(t) \tag{4.5}$$

And thus we can get g^n:

$$g^n(x^{n+1}, x^n, u^n) = x^{n+1} - (A(x^n)\Delta t + I)x^n - (B(x^n)\Delta t L_{qu})u^n = 0 \tag{4.6}$$

4.2 Gradient-based optimization method

To solve the above problem, the existing optimization algorithms can be generally classified into two categories: stochastic algorithms like Genetic Algorithm and Simulated Annealing, and gradient-based algorithms like Steepest Descent and Quasi-Newton algorithms. The formal one usually requires a large amount of forward simulations since it is stochastic, which is really time-consuming especially when the number of grid blocks is large. However, the shortcoming of the latter one is that the optimal control we get is sometimes not a global one. In practice, the number of grid blocks can be very large, and thus a single simulation can take many hours, it is not realistic to use a stochastic algorithm. As a result, even if what we get from the Gradient-based algorithms is not a global one, it can improve the whole system a lot.

4.2.1 Gradients with the Adjoint Model

A time step in a control step

We add the constrained term with Lagrange multipliers λ^n. Thus the modified objective function becomes:

$$\bar{J} = \sum_{n=0}^{N-1}[J^n(x^n,u^n) + (\lambda^{n+1})^T g^n(x^{n+1},x^n,u^n)] \tag{4.7}$$

We denote $L^n = J^n(x^n,u^n) + (\lambda^{n+1})^T g^n(x^{n+1},x^n,u^n)$ (4.8)

as an auxiliary function.

Then we can get the first order partial derivation of J as a term of $x^{n+1}, x^n, u^n, \lambda^{n+1}$:

$$\delta \bar{J} = \sum_{n=1}^{N-1}(\frac{\partial L^n}{\partial x^n})\delta x^n + \sum_{n=0}^{N-1}(\frac{\partial L^n}{\partial x^{n+1}})\delta x^{n+1} + \sum_{n=0}^{N-1}(\frac{\partial L^n}{\partial u^n})\delta u^n + \sum_{n=0}^{N-1}(\frac{\partial L^n}{\partial \lambda^{n+1}})\delta \lambda^{n+1} \tag{4.9}$$

And thus we can change (4.9) a bit as:

$$\delta \bar{J} == \sum_{n=1}^{N-1}[\frac{\partial L^{n-1}}{\partial x^n} + \frac{\partial L^n}{\partial x^n}]\delta x^n + \sum_{n=0}^{N-1}(\frac{\partial L^n}{\partial u^n})\delta u^n + \sum_{n=0}^{N-1}(\frac{\partial L^n}{\partial \lambda^{n+1}})\delta \lambda^{n+1} + (\frac{\partial L^{n-1}}{\partial x^N})\delta x^N \tag{4.10}$$

It is not difficult to find that $\frac{\partial L^n}{\partial \lambda^{n+1}} = (g^n)^T = 0^T$, and if the last term of (4.10) $\frac{\partial L^{N-1}}{\partial x^N} = 0$,which is called the Final Condition. And the first term of (4.10) can also be set to 0 by imposing:

$$\frac{\partial L^{n-1}}{\partial x^n} + \frac{\partial L^n}{\partial x^n} = 0 \tag{4.11}$$

Substitute (4.8) into (4.11), we get:

$$(\lambda^n)^T(\frac{\partial g^{n-1}}{\partial x^n}) = -(\lambda^{n+1})^T \frac{\partial g^n}{\partial x^n} - \frac{\partial J^n}{\partial x^n} \tag{4.12}$$

We can use the Final Condition to get λ^N and use (4.12) can compute backward to get all λ^n, $for\ \forall n$:

$$\begin{aligned} \lambda^N &= [\frac{\partial J^{N-1}}{\partial x^N}][\frac{\partial g^{N-1}}{\partial x^N}]^{-1}\ \ (Final\ \ Condition) \\ \lambda^n &= -[\frac{\partial J^n}{\partial x^n} + (\lambda^{n+1})\frac{\partial g^n}{\partial x^n}][\frac{\partial g^{n-1}}{\partial x^n}]^{-1} \end{aligned} \tag{4.13}$$

After all the Lagrange Multipliers are computed, we can simplified (4.9) as:

$$\delta \bar{J} = \sum_{n=0}^{N-1}(\frac{\partial L^n}{\partial u^n})\delta u^n = \sum_{n=0}^{N-1}[\frac{\partial J^n}{\partial u^n} + (\lambda^{n+1})^T \frac{\partial g^n}{\partial u^n}]\delta u^n \tag{4.14}$$

In order to get an optimized objective function, we have known that if the controls u^n are not constrained and so we have:

$$\frac{\partial L^n}{\partial u^n} = 0^T \tag{4.15}$$

If the controls are constrained, the optimal control policy optimizes L^n. For the problem, we can get u^n_{opt}:

$$L^n(x^{n+1}, x^n, u^n_{opt}, \lambda^{n+1}) \geq L^n(x^{n+1}, x^n, u^n, \lambda^{n+1}) \tag{4.15}$$

More time steps in a control step

More generally, when the time steps and control steps are not equivalent, in other words, there are more than one time step in one control step. The other conditions are as the same as what we discussed above. As a result, the production optimization process will be formulated as:

$$\begin{aligned}
&\max_{u^n} J = \sum_{n=0}^{N-1}\sum_{m=0}^{M-1} J^{n,m}(x^{n,m+1}, u^n) \\
&subject\ \ to: \\
&g^{n,m}(x^{n,m+1}, x^{n,m}, u^n) = 0 \\
&x^{0,0} = x_{0,0}(initial\ condition) \\
&c^n(x^{n+1}, u^n) \leq 0 \\
&Au^n \leq b \\
&LB \leq u^n \leq UB \\
&for\ \ \forall m \in (0,1,...,N-1)\ \ and\ \ \forall n \in (0,1,...M-1)
\end{aligned} \tag{4.16}$$

Where M is the number of time steps for each control step, m and n are the time steps and control steps respectively. The backward equations for Lagrange Multipliers are:

$$\begin{aligned}
&\lambda^{Tn,M} = [\frac{\partial J^{n,M-1}}{\partial x^{n,M}} + \lambda^{Tn+1,1}\frac{\partial g^{n+1,0}}{\partial x^{n,M}}][\frac{\partial g^{n,M-1}}{\partial x^{n,M}}]^{-1}\ \ (Final\ \ Condition) \\
&\lambda^{Tn,M} = -[\frac{\partial J^{n,M-1}}{\partial x^{n,M}} + (\lambda^{Tn,m+1})\frac{\partial g^{n,m}}{\partial x^{n,m}}][\frac{\partial g^{n,m-1}}{\partial x^{n,m}}]^{-1}, \forall m = 0,..,M-1;
\end{aligned} \tag{4.17}$$

The gradient of the NPV function is consequentially as:

$$\delta \bar{J} = \sum_{m=0}^{M-1} (\frac{\partial J^{n,m}}{\partial u^n}) \delta u^n = \sum_{m=0}^{M-1} [\frac{\partial J^{n,m}}{\partial u^n} + (\lambda^{n+1,m})^T \frac{\partial g^{n,m}}{\partial u^n}] \delta u^n$$

Note that $x^{n,M} = x^{n+1,0}, \lambda^{n+1,1} = \lambda^{n,M}$. In this case, we can choose the control steps as we expect without considering the time steps.

4.2.2 The optimization procedure

The procedure of this optimization problem is a bounded value optimization. Solution is got through repeating the following steps:

1. Find the numerical simulation of the dynamic system behavior in the time interval [0,T]. The initial conditions are x^0, u.
2. Get the objective value J with results of the forward simulation.
3. Using the Final condition to get λ^N and get the Lagrange multipliers by backward numerical solution of (3.9).
4. Calculation of $\frac{\partial L^n}{\partial u^n}$ and then get an updated control vector u.
5. Repeating step 2-4 until the global optimization u_{opt}^n reached.

4.3 Case Study (441 grids case)

As shown before, the reservoir is divided into 441 grid blocks. The five wells (four production wells in the corners and an injection well in the center) are fixed.

4.3.1 Set up the Experiments

The experiment is set up similarly with Chapter 3, but we extend experiment time to 1500 days. We intend to control the production settings—bottom hole pressures in four production wells in the corner and injection flow rate in injection well in the center of the reservoir—in order to optimize the NPV. We use the gradient-based optimization method as discussed above to get the optimal well production settings. For simplicity, we choose the control step number as 1 at time 0.

As we discussed in Chapter 2, it is reasonable to choose the minimum and maximum of the prescribed settings. For production settings, considering the surface facilities, we choose the minimum as 10^6 Pa and maximum is constrained by the

pressures of the surrounding grid blocks in order to get an out flow. For injecting settings, the injection flow rate can be infinitely fast, however, considering the cost of the injection, it is naturally constrained in a reasonable interval. For initialization, we can choose it randomly.

The values we take for variables are shown in the tables as follows.

Table 1 shows the geological variables like porosity and viscidity, etc. Table 2 shows the oil revenue, water production output and water injection cost per unit volume (cubic meter)

Table 1-Rock and Fluid Properties		
ϕ	0.2	
c_o, c_w, c_r	1.0×10^{-9}	$1/Pa$
ρ_o, ρ_w	1000	kg/m^3
μ_o	0.5×10^{-3}	$Pa\bullet s$
μ_w	1.0×10^{-3}	$Pa\bullet s$
k_{ro}^0	0.9	
k_{rw}^0	0.6	
S_{or}, S_{wc}	0.2	

Table 4.1 the rock and fluid properties

Table 2-Values for computing NPV		
r_o	300	USD/m^3
r_{wp}	15	USD/m^3
r_{wi}	5	USD/m^3
b	0	

Table 4.2 the values for computing NPV

The table above shows the oil revenue price, water disposal cost and water injection cost per volume. b denotes the discount rate, here we suppose the discount rate is 0 for convenience.

4.3.2 Experimental results and analysis

We set the initial prescribed bottom hole pressures in four corner wells as 2.5×10^7 Pa and the injection rate as $0.002\ m^3/s$. Considering a relatively precise result, it is better to choose a smaller updating step size. When we choose the step as improving $10^5\ USD$ every step (the total in a unit of $10^7\ USD$), the NPV graph we get for all iterations shows:

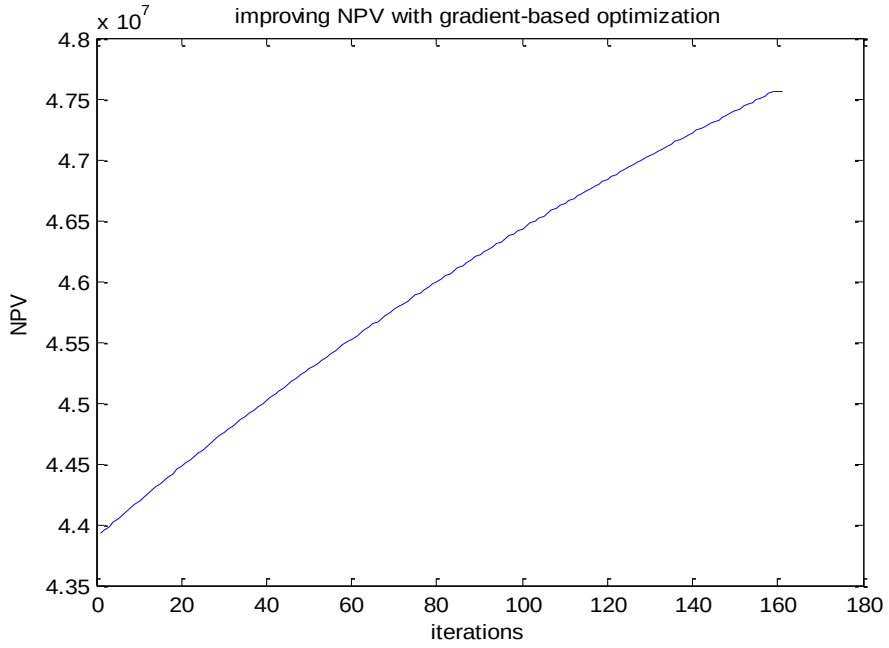


Figure 4.1 improving NPV with gradient-based optimization

	Initialization	optimum
NW pressure	2.5×10^7	1.0×10^6
NE pressure	2.5×10^7	1.1×10^6
SW pressure	2.5×10^7	1.2×10^6
SE pressure	2.5×10^7	1.0×10^6
Injection rate	0.002	0.0027
NPV	4.3926×10^7	4.7559×10^7

Table 4.3 optimization with "10^5" step

When the updating step changes, what will happen to the precision?

Next we choose the size as improving $10^6\ USD$ every step. The figure turns out:

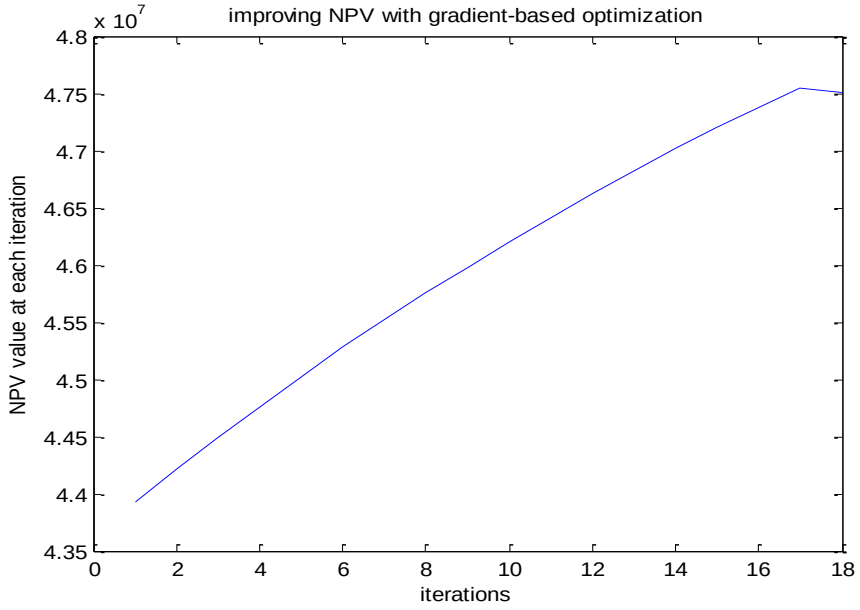


Figure 4.2 improving NPV with gradient-based optimization

	Initialization	optimum
NW pressure	2.5×10^7	1.0×10^6
NE pressure	2.5×10^7	2.0×10^6
SW pressure	2.5×10^7	3.4×10^6
SE pressure	2.5×10^7	1.0×10^6
Injection rate	0.002	0.0028
NPV	4.46×10^7	4.7503×10^7

Table 4.4 optimization with "10^6" step

It turns out that the output will not differ from the previous optimum much when we choose the step size as improving $10^6\ USD$ every step. However, it will save the cost of time a lot since the iteration number will be reduced to 17 iterations.

What happened if the initialization values change?

For example we set a random setting different from previous one and the updating

step keeps as improving $10^6\ USD$ every step.

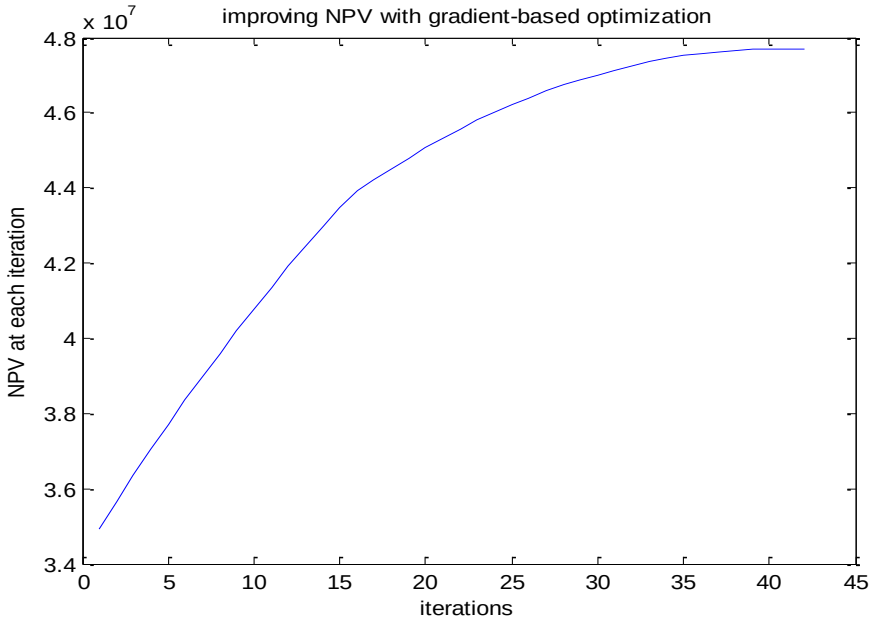


Figure 4.3 improving NPV with gradient-based optimization

	Initialization	optimum
NW pressure	2.0×10^7	1.0×10^6
NE pressure	2.0×10^7	1.87×10^6
SW pressure	2.0×10^7	3.08×10^6
SE pressure	2.0×10^7	1.0×10^6
Injection rate	0.001	0.0025
NPV	3.49×10^7	4.7535×10^7

Table 4.5 optimization with a different initial setting

After several experiments, we can conclude that all the different initializations can converge to an optimal NPV around 4.75×10^7 USD as we expect. A step of improving 10^5 USD for a step is acceptable considering precision.

In conclusion, gradient-based method with adjoint model works well and stably in simple simulation of reservoir field. Actually, there are numerous applications of adjoint-based optimization of production settings in the petroleum engineering literature. Some of the earlier ones are by Ramirez and co-workers,summarized in

Ramirez(1987), who considered tertiary recovery techniques. This was quickly followed by Asheim(1987), Asheim(1988), Virnovsky(1991), Zakirov et al.(1996), and Sudaryanto and Yortsos(2001), who considered secondary recovery rechniques Although the type of production settings differ from each other, they are all applications of the same technique: gradient-based optimization with gradients computed using an adjoint model. It received significant attention after Brouwer(2004) and Jansen(2004) demonstrated the possibility to significantly increase the recovery factor using smart wells.

5 Optimal well placement

Determining optimal well locations with fixed well settings is an efficient way to increase the Net Present Value as modeled in the reservoir simulator. Usually we have a reservoir model with a large amount of grid blocks, so it is reasonable to use stochastic optimization method such as genetic algorithm (Holland,1992),ant colony optimization(Dorigo&Gambardella,1997),etc. In this chapter, firstly, we introduce a search method based on natural systems—"Particle Swarm Optimization" (PSO)(Eberhart&Kennedy,1995), which is a population-based, self-adaptive search optimization method motivated by the observation of simplified animal social behaviors. It is becoming very popular because of the quick convergence and a relatively good solution

5.1 Particle Swarm Optimization

5.1.1 A brief introduction of PSO

Particle Swarm Optimization (PSO) was originally developed by Kenedy and Eberhart in 1995, which is a population-based evolutionary algorithm inspired by social behavior of bird flocking or fish schooling. PSO shares many similarities with evolutionary computation techniques such as Genetic Algorithms (GA). The system is initialized with a population of random solutions using uniform distribution and searches for optimal solution by updating generations. PSO is found to be very robust in solving nonlinear, multiple optimization and high dimensional problems through adaption. In a PSO system, multiple candidate solutions coexist and collaborate simultaneously. Each solution is called a "particle", which flies towards the search space and find an optimal location to land. A particle adjusts its position according to its own "experience" and others' "experience"

The key terms used in PSO are:

(1) Particle(individual, agent): each individual in the swarm
(2) Position/Location: a particle's n-dimensional coordinates which represents a solution to the problem
(3) Swarm: the entire collection of particles
(4) Fitness: the fitness function gives the interface between the physical problem and the optimization problem. It is a value representing the goodness of a given solution given by a position in solution space
(5) Generation: each iteration of optimization procedure using the PSO
(6) Pbest (personal best): the position in parameter space of the best fitness returned for a specific particle
(7) Gbest (global best): the position in parameter space of the best fitness returned

for the entire swarm

(8) $V_{\max}$: the maximum velocity value allowed in a given direction

(9) Velocity: a vector relating to the updating of the location of a particle.

Each particle keeps track of its coordinates in the problem space which are associated with the best solution it has achieved so far, pbest and gbest. The pso algorithm, at each time step, changes the velocity of each particle moving around its pbest and gbest locations. Velocity is weighted by random terms, with separate random numbers being generated for acceleration toward pbest and gbest locations, respectively. The procedure for implementing the global version of PSO is given by:

(1) Initialize a swarm of particles with random choose among the solution space.
(2) For each particle, evaluate the fitness value.
(3) Compare each particle's fitness value to the current particles pbest. If the current value is better than gbest, then set its pbest value as the current value and the pbest location to the current location in n-dimensional space.
(4) Compare the fitness value with the swarm's overall previous best. If current value is better than the gbest, then set gbest as the current value,.
(5) Change the velocity and position of the particle according to the following equations:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot u_d \cdot [p_i(t) - x_i(t)] + c_2 \cdot U_d \cdot [p_g(t) - x_i(t)] \quad (5.1)$$

$$x_i(t+1) = x_i(t) + \Delta t \cdot v_i(t+1) \quad (5.2)$$

Where i=1,2,… is the particle index, t is the time index, Δt is chosen to be 1. $x_i(t) = [x_{i1}(t), x_{i2}(t), ..., x_{in}(t)]^T$ denotes the physical location of the i-th particle. $v_i(t) = [v_{i1}(t), v_{i2}(t), ..., v_{in}(t)]^T$ represents the velocity of the i-th particle and $p_i(t) = [p_{i1}(t), p_{i2}(t), ..., p_{in}(t)]^T$ stands for the best previous position of the i-th particle. The index g shows the index of the best particle among all the particles in the group at time t. c_1 and c_2 are positive constants, which are called cognitive learning rate and social learning rate respectively. u_d and U_d are two separately generated uniformly distributed random numbers in the interval [0,1] generated according to a uniform probability distribution. w is the inertia weight factor. (5.2)shows the position update using its previous and velocity.

(6) Step 1-5 until a certain criteria is reached.

After a brief introduction to the particle swarm optimization, it is reasonable to get a try on applying it to real cases. Take 441-grid reservoir we discuss above as an example, we do the experiment to test this optimization method.

5.1.2 Parameter Selection in PSO

By observing (5.1) more carefully, it can be seen that the maximal velocity allowed actually serves as a constraint that controls the maximum global exploration ability that PSO method can have. By setting a large maximum velocity extent, PSO can have a large range of exploration ability to select. However, the inertia weight affects the searching directly and maximum velocity limitation affects indirectly.

According to "Parameter selection in Particle Swarm Optimization"(Yuhui Shi and Russell C Eberhart), a lot of experiments result in a conclusion that when Vmax is small, and inertia weight approximate to 1 is a good choice, while the Vmax is not small, then a smaller inertia weight is more appropriate. When we lack the knowledge of Vmax, we usually set Vmax a value of maximal moving distance of one direction and an inertia weight around 0.5 is a good choice. Furthermore, if a time varied inertia weight is applied to the experiment, a better performance is expecting. In this case, I choose inertia weight as a randomly value distributed normally in an interval.

In many PSO models, cognitive learning rate and social learning rate are usually set with the same value, which makes the self "experience" learning rate be in equilibrium with social experience.

5.2 Experiment and results

5.2.1 Set up the model

We still take the case of 441-grid reservoir model. In order to optimize the Net Present Value, we will determine the optimal well locations (4 production wells and 1 injection well).

In this case, we fix the well settings as: prescribed BHP as 2.5×10^7 Pa for 4 production wells and prescribed injection rate as 0.002 m^3/s.

The objective function is still the NPV value function.

Initialization of the swarm

Since it is 2-dimensional reservoir model, the particle can be formulated as the coordinates of the five wells (in the order of four production wells and injection well)

We define the swarm as: swarm (index, [location, velocity, best position, best value], [x, y components], well index), where index indicates the index of the particles in a swarm. [location, velocity, best position, best value] shows the properties of the individual particle. [x,y components] means the x and y coordinates corresponding to the location. Well index is from 1 to 5. Well 1 to 4 is production well and 5 is injection well.

I choose the swarm size as 16(considering there are only 441 grids blocks in the model) and initialize the swarm as follows:

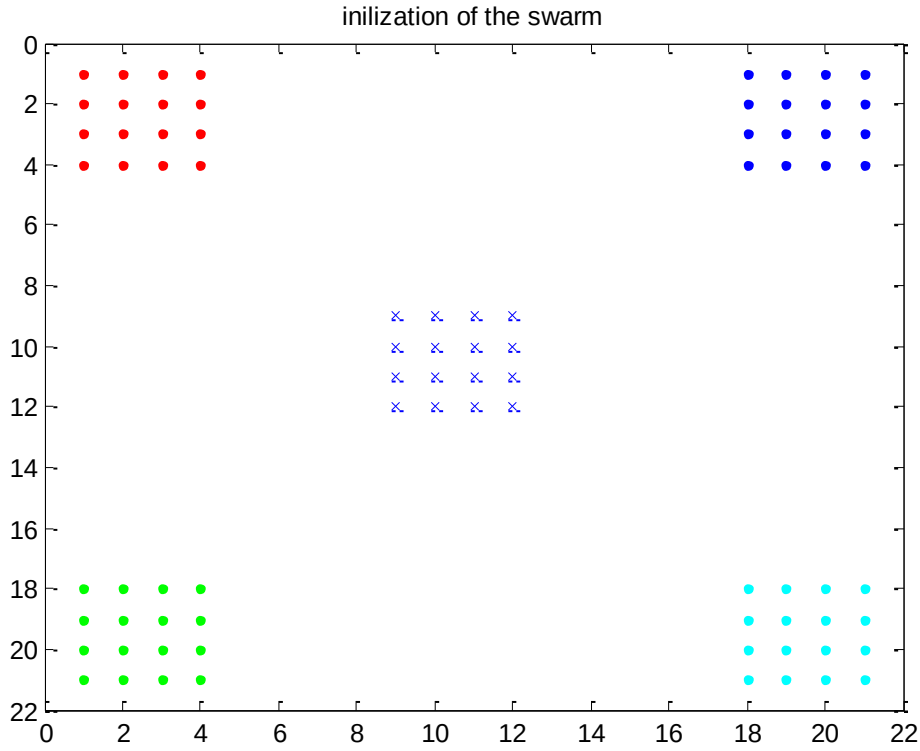


Figure 5.1 initialization of the swarm

Update of the swarm

The formula of the update velocity function (5.1) reads:

$$v_i(t+1) = w \cdot v_i(t) + c_1 \cdot u_d \cdot [p_i(t) - x_i(t)] + c_2 \cdot U_d \cdot [p_g(t) - x_i(t)] \quad (5.3)$$

For code, it can be transformed to:

swarm(i,2,w,j)=rand*inertia*swarm(i,2,w,j)+correction_factor*rand*(swarm(i,3,w,j)-swarm(i,1,w,j))+correction_factor*rand*(swarm(gbest,3,w,j)-swarm(i,1,w,j)); (5.3)

where w=1,2, i=1～swarm size, and j=1～4. “rand” means the randomly number between 0 and 1 chosen from a uniform distribution. In this case, I set the cognitive learning rate and social learning rate the same value, which is correction_factor as 2.swarm(i,3,w,j) is the personal best position of the particle i, well j. swarm(gbest,3,w,j) is the global best position of the best particle. Consequently the updating function for the position of particle i, well j is:

swarm(i,1,w,j)=swarm(i,1,w,j)+swarm(i,2,w,j). (5.4)

It is illustrated well in the following figure, which shows the velocity update in 2-dimensional space.

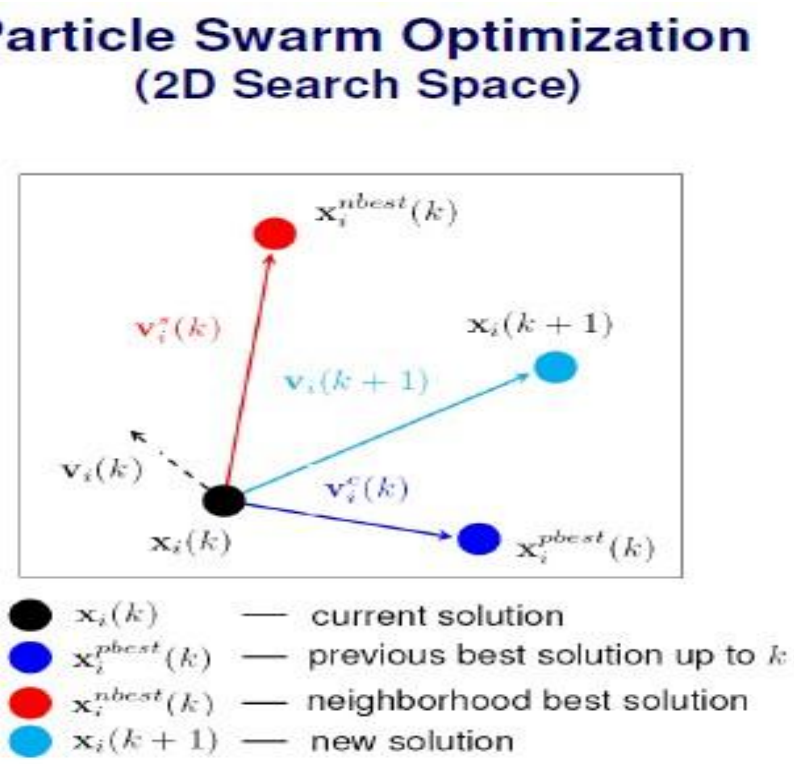
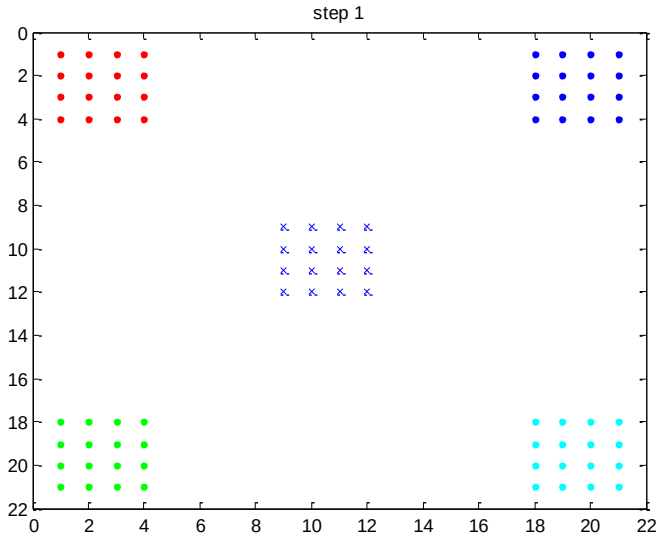
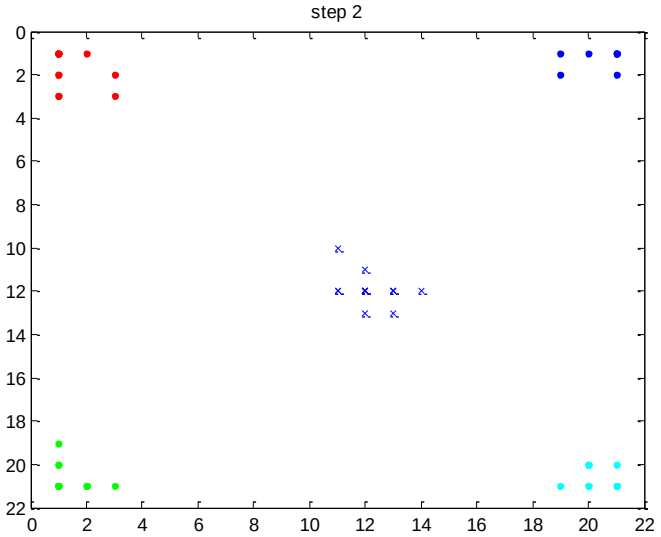


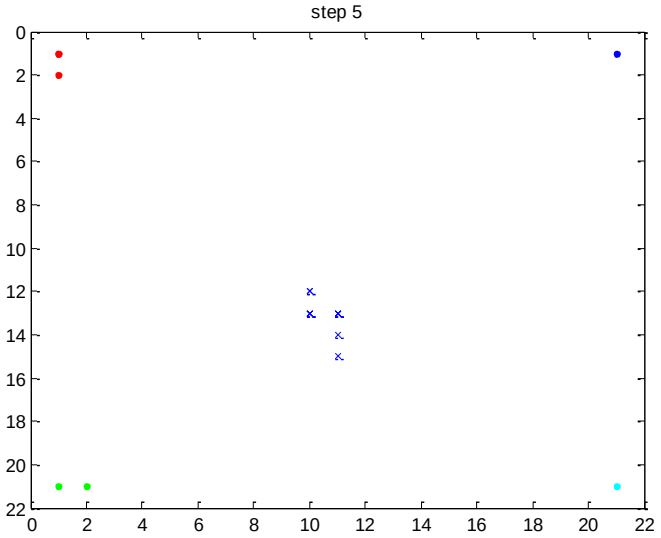
Figure 5.2 update of the velocity of particle

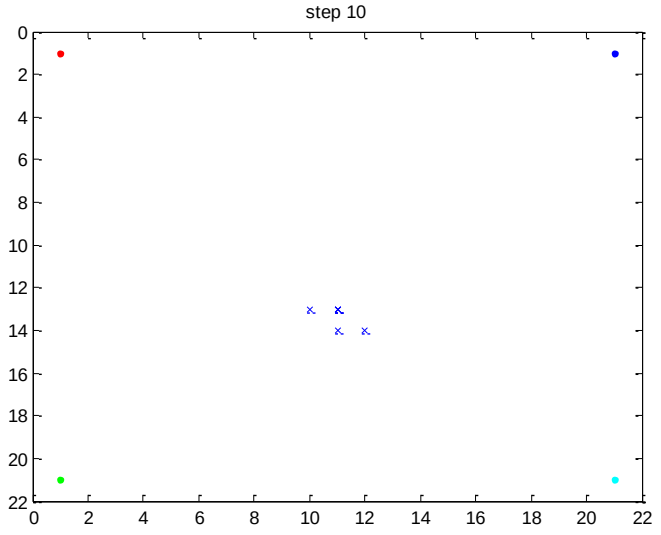
5.2.2 Results and Analysis

We can see the process of the PSO optimization as shown by figures:




step 2
0
2
4
6
8
10
12
14
16
18
20
22
0
2
4
6
8
10
12
14
16
18
20
22


step 5
0
2
4
6
8
10
12
14
16
18
20
22
0
2
4
6
8
10
12
14
16
18
20
22


step 10
0
2
4
6
8
10
12
14
16
18
20
22
0
2
4
6
8
10
12
14
16
18
20
22

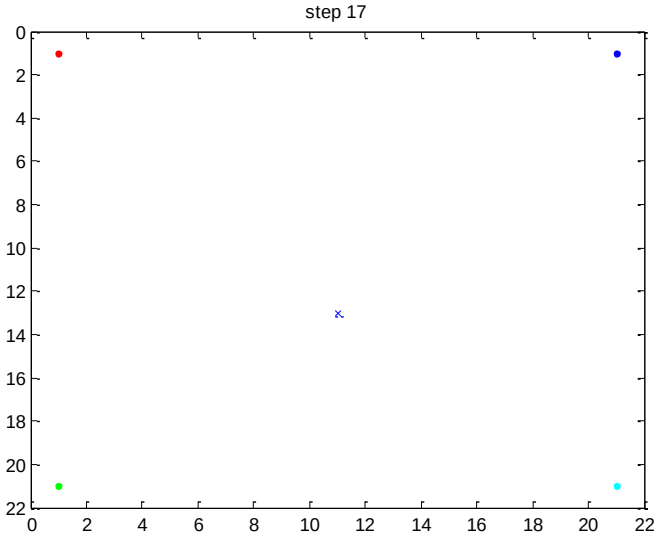


Figure 5.3 the process of PSO

This group of figures shows the steps of particle swarm optimization. After 17 steps, the particles in the swarm converge to a single one, which indicates that it is the optimal well placement from the experiment.

The change of the NPV value is as follows:

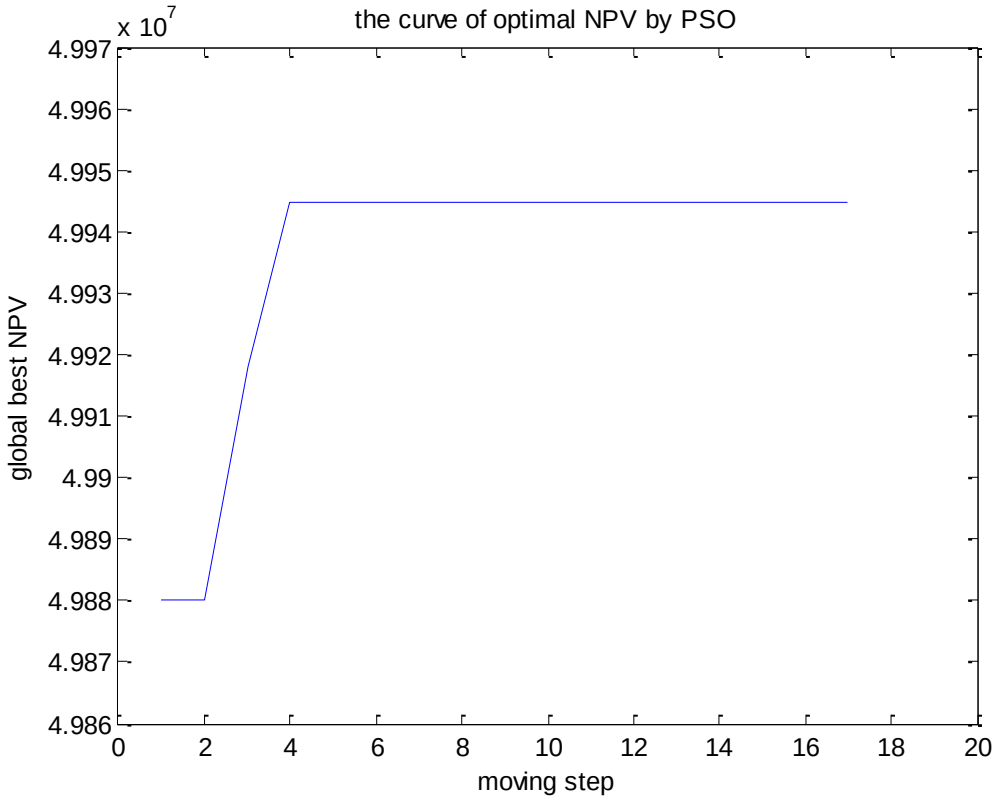


Figure 5.4 the curve of global best NPV by steps

For another example when we change the well settings as the optimal one as table 4.5 in Chapter 4. The optimization result becomes:

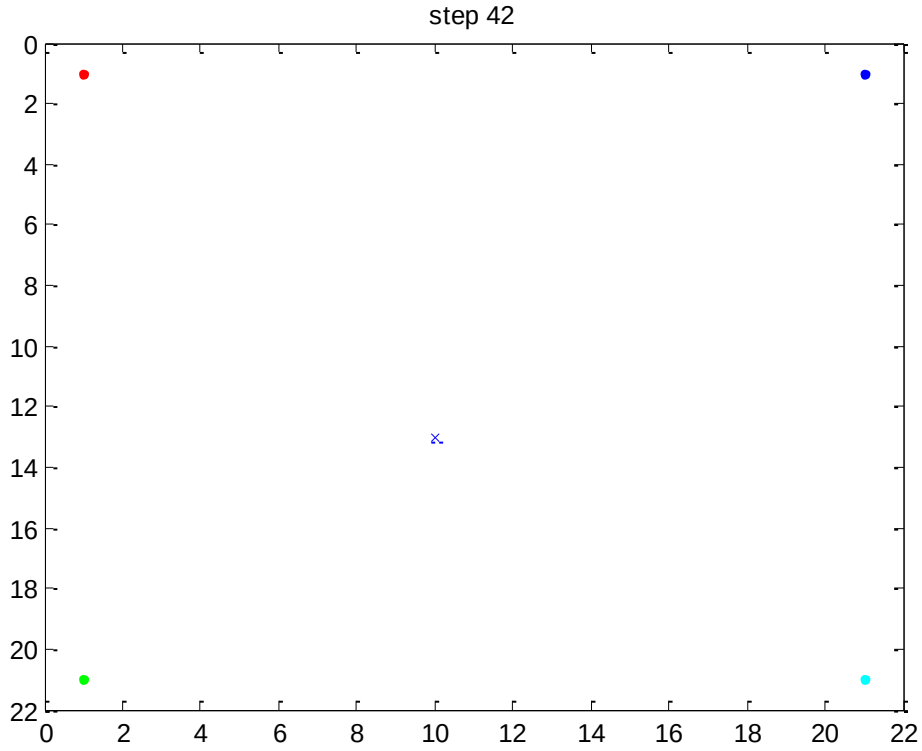
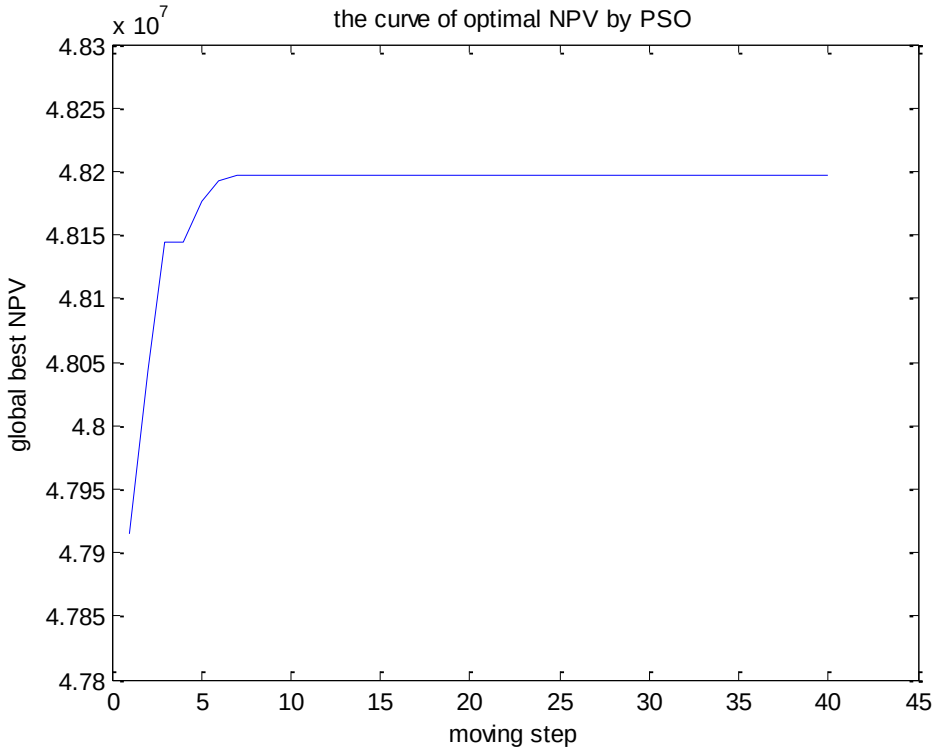


Figure 5.5 optimal well placement with optimized initial settings



It seems that we improve the NPV by selecting well placement to a very limited extent compared to changing well settings, considering the uncertainty of the reservoir model, it seems not necessary enough to change the well location if the

wells have already been fixed.

In conclusion, the different production settings can affect the well placement, that is to say, a well configuration that is optimal when the wells are operated with one type of settings may be far from optimal when the wells are operated with another type of settings. Furthermore, there is another category of methods used in optimizing well placement. The method used in this Chapter(PSO) is categorized as one of the global methods, which consist of methods like simulated annealing (Beckner and Song 1995), genetic algorithms(Montes et al.2001,Guyaguler et al.2002, Yeten et al.2003),and neural networks (Centilmen et al.1999). The other category consists of local methods such as finite-difference-gradient (FDG) (Bangerth et al.2006), simultaneous-perturbation-stochastic-approximation (Bangerth et al.2003, Spall 2003) and Nelder-Mead Simplex(Spall 2003) methods. The second catergory is generally very efficient, requiring only a few forward reservoir simulations, and increases NPV at each iteration. However, these methods can get stuck in a local optimal solution. The first category can, in theory, avoid this problem but has the disadvantages of not increasing NPV at each iteration and requiring many forward reservoir simulations.

6 Case study

This Chapter presents a discussion about the optimization of the 5-spot reservoir. As we know, there are many uncertainties when we are confronted with real problems, e.g. the permeability and porosity field might be unknown at the beginning.

In section 6.1 we intend to start the experiment from a random hypothesis of the permeability and porosity field with fixed well placement and use data assimilation to estimate the true geological properties of the field. The experiment begins from the bottom right of the figure below. As the figure shows, after the data assimilation we perform the optimization of the well settings. It is interesting to compare the NPV using optimized production settings with the true one based on the true field. The uncertainty according to data assimilation and optimization process is important to discuss.

In Section 6.2, we start from the left bottom of the figure (the true permeability and porosity field), and then optimize the NPV by determining well placements and controlling well settings.

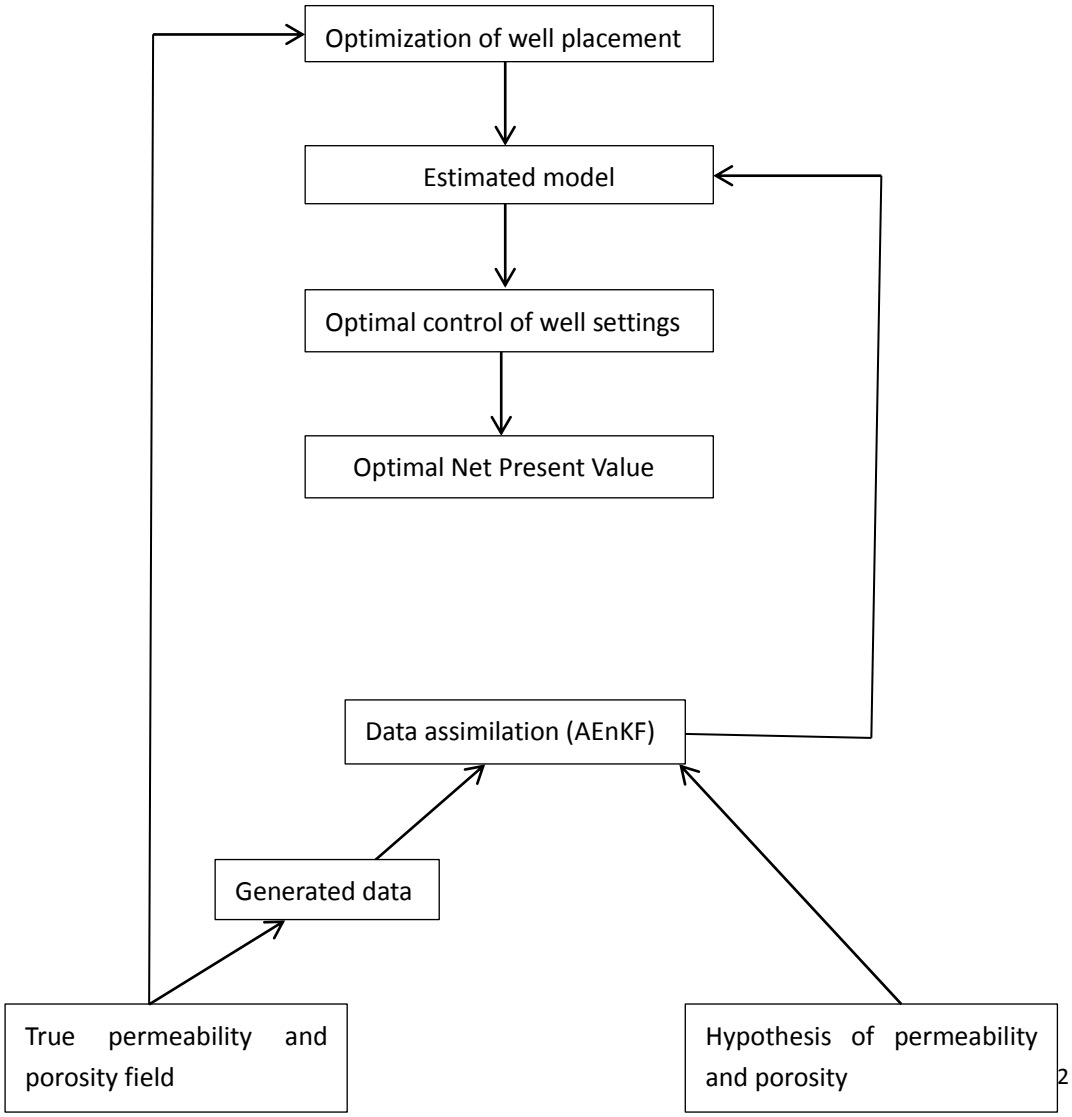


Figure 6.1

6.1 Optimization with estimated field data

As discussed in Chapter 3, it is better to choose more than 60 ensembles to estimate permeability and porosity distribution. In this case, 100 ensembles have been used to get the estimated rock property field of the reservoir. We compare the estimated results with the true permeability and porosity field.

The experiment is still done on the 441-square reservoir field as we discussed in the previous chapters. In Chapter 3, data assimilation with AEnKF has been studied, and now we use estimated permeability and porosity to start the optimization. The well locations are fixed in these experiments. (four production wells in the corner and one injection well in the center)

Original well placement

The ensemble size we choose as 100 and the data assimilation time is from 0 to 300 days. The figure 6.2 shows the estimated filed and the true one.

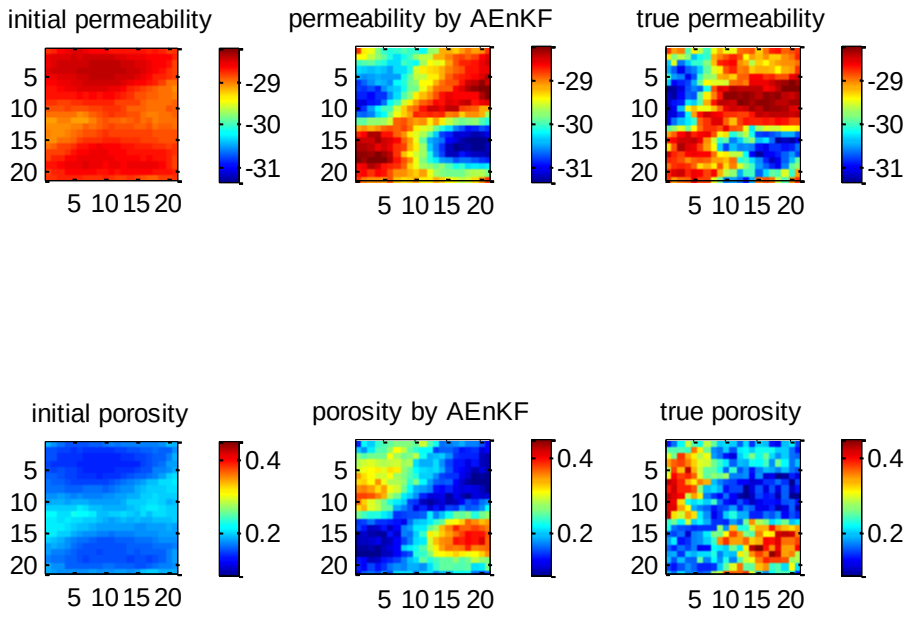


Figure 6.2 estimated field with AEnKF

Applying a gradient-based method to optimize the control of well settings using the estimated permeability and porosity field, we produce the results shown below:

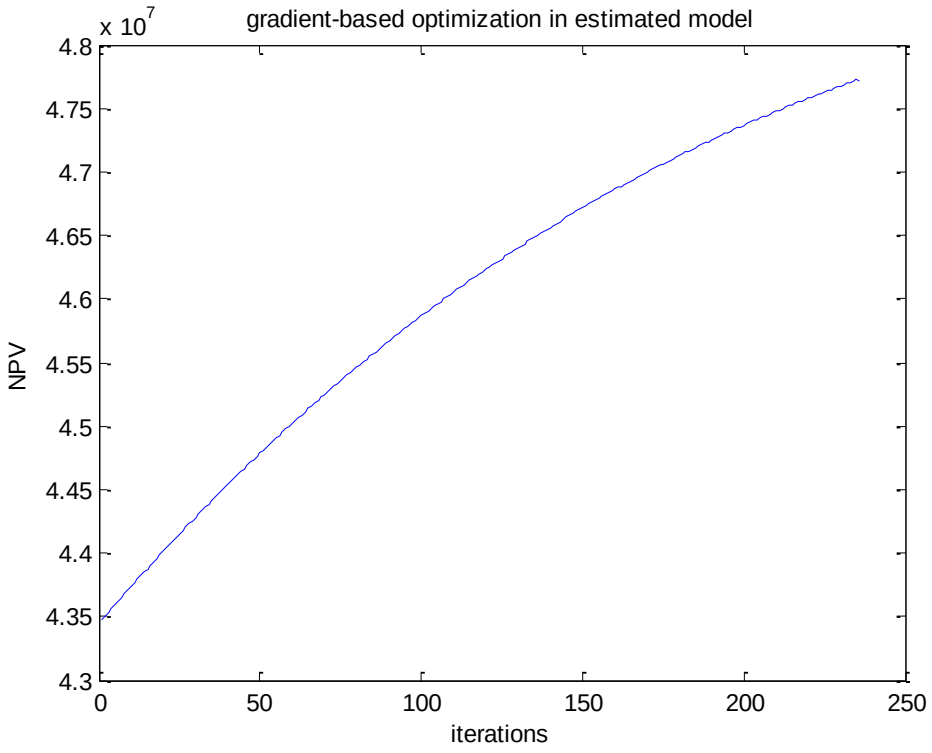


Figure 6.3 gradient-based optimization of well settings

	Initialization	optimum
NW pressure	2.5×10^7	1.0×10^6
NE pressure	2.5×10^7	1.1×10^6
SW pressure	2.5×10^7	1.2×10^6
SE pressure	2.5×10^7	1.0×10^6
Injection rate	0.002	0.0032
NPV	4.35×10^7	4.7723×10^7

Table 6.1 optimize NPV by well settings using estimated permeability and porosity field

The optimal NPV is still around 4.77×10^7, when the settings are applied with the real permeability and porosity field, the output becomes 4.6884×10^7.It is interesting to carry out more experiments of the AEnKF to get more insight into the performance of the method.

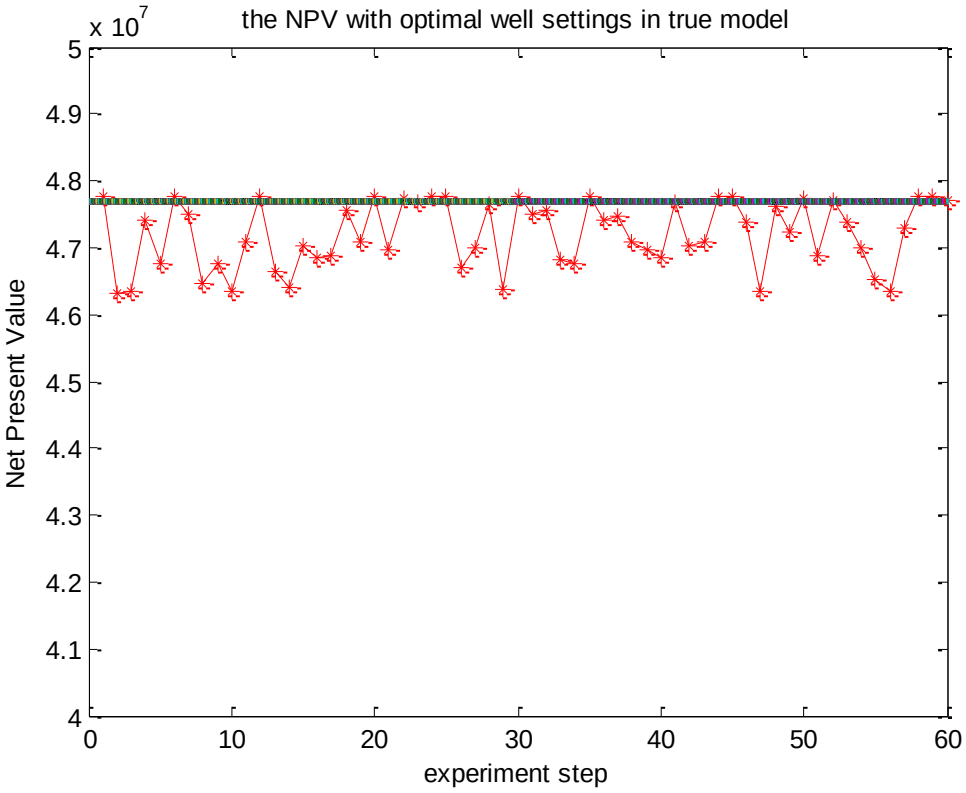
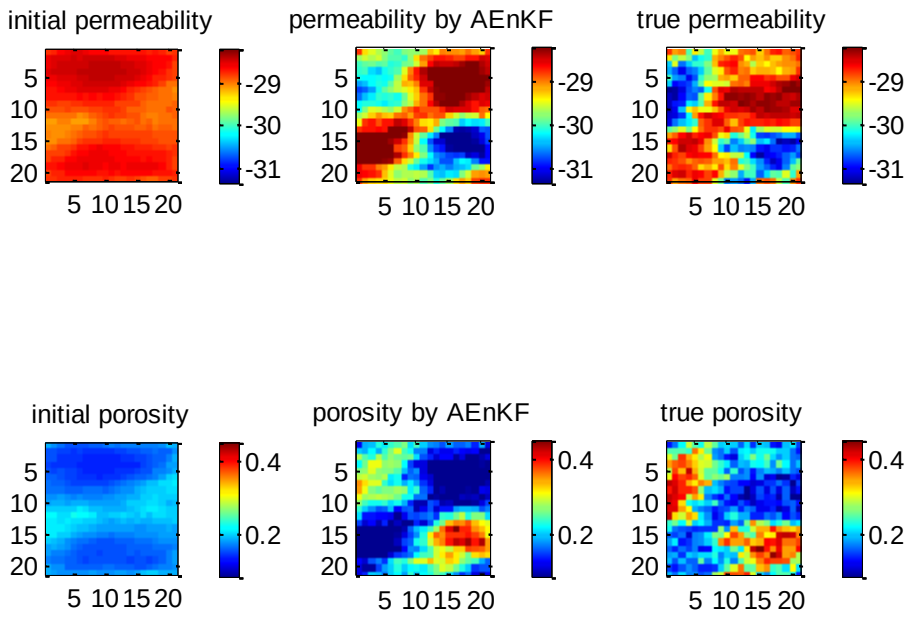


Figure 6.4 Optimal NPV from estimated model of 60 experiments

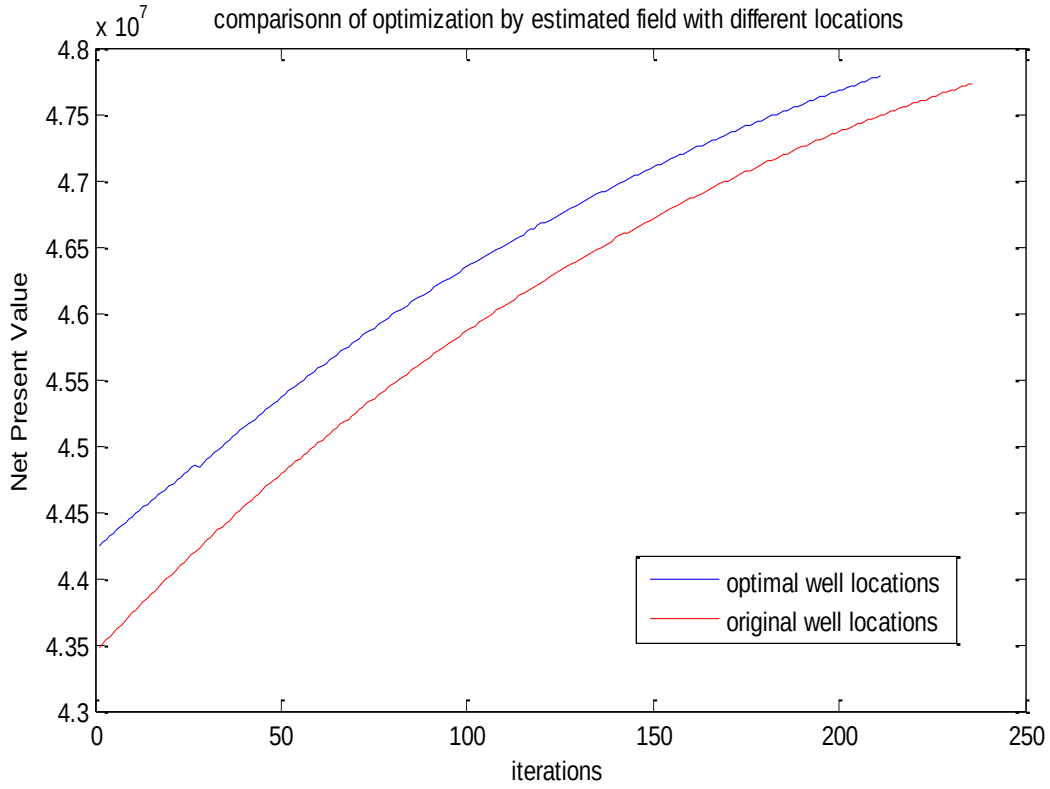
The line in the figure is $y = 4.77 \times 10^7$ .It is shown in above figure that the fluctuation interval of the optimal NPV is $(4.61 \times 10^7, 4.78 \times 10^7)$, which is acceptable compared to the magnitude of NPV.

Optimal well placement

It is discussed in Chapter 5, with the well settings as prescribed pressure of 2.5×10^7 Pa in the four production wells and prescribed flow rate of $0.002\ m^3/s$ in the injection well, the optimal well location is shown in Figure 5.3. In this part, I will generate data with AEnKF with optimal well location given in Figure 5.3 and get another estimated permeability and porosity field. What will happen to the result?



The optimization results shows below:



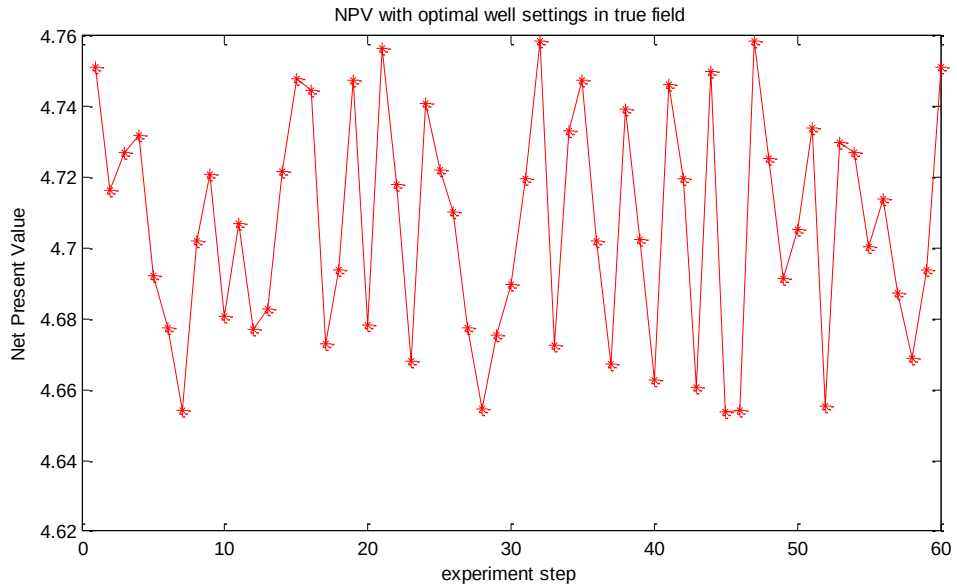
As shown in the figure, the change of well placement will influence the data assimilation results and thus the optimization results to a limited extent. The

optimization of the well placement seems not to improve the NPV a lot in our model with our parameters.

	Initialization	optimum
NW pressure	2.5×10^7	1.0×10^6
NE pressure	2.5×10^7	1.1×10^6
SW pressure	2.5×10^7	1.4×10^6
SE pressure	2.5×10^7	1.0×10^6
Injection rate	0.0020	0.0029
NPV	4.4244×10^7	4.7789×10^7

Table 6.2 optimize NPV by well settings using estimated data(optimal locations)

When we apply the settings we get in the above procedure with the real permeability and porosity field, we will get the NPV as 4.7192×10^7 USD, a little higher than what we get from the previous one in original well locations.



The above figure shows that the value of the NPV using the optimal well settings given by the estimated field falls into $(4.65\times10^7, 4.76\times10^7)$, which demonstrates that the data assimilation is effective with another well configuration.

In conclusion, when the reservoir is modeled with different well placement, the estimated output will differ from each other due to the change of the placement, however, when the settings are applied to the real model, the NPV will fall into a

reliable interval.

6.2 Optimization with true field data

Given the true field data of permeability and porosity, the first step to optimize the output is to determine the optimal well placement. As we have seen in Chapter 5, the optimal well location for the initial well settings is given as:

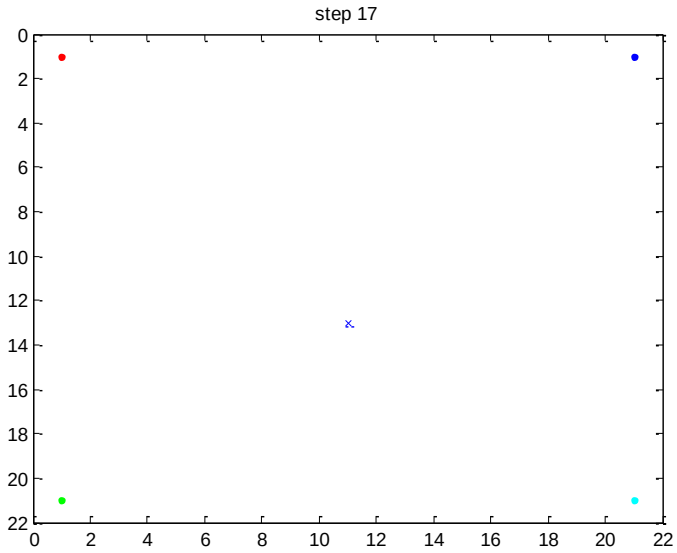


Figure 6.4

When the locations are determined, the next step is to control well settings to improve the output value. The result is shown below:

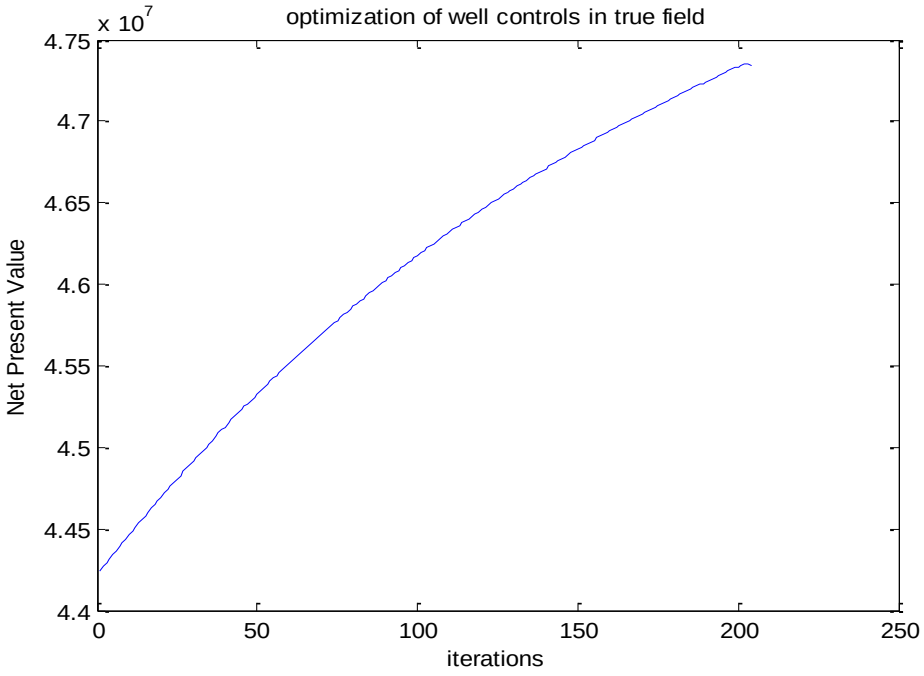


Figure 6.5

	Initialization	optimum
NW pressure	2.5×10^7	1.0×10^6
NE pressure	2.5×10^7	1.16×10^6
SW pressure	2.5×10^7	1.13×10^6
SE pressure	2.5×10^7	1.0×10^6
Injection rate	0.002	0.0030
NPV	4.43×10^7	4.7344×10^7

Table 6.2

The results seem not to improve the NPV compared to the original well locations after a few iterations.

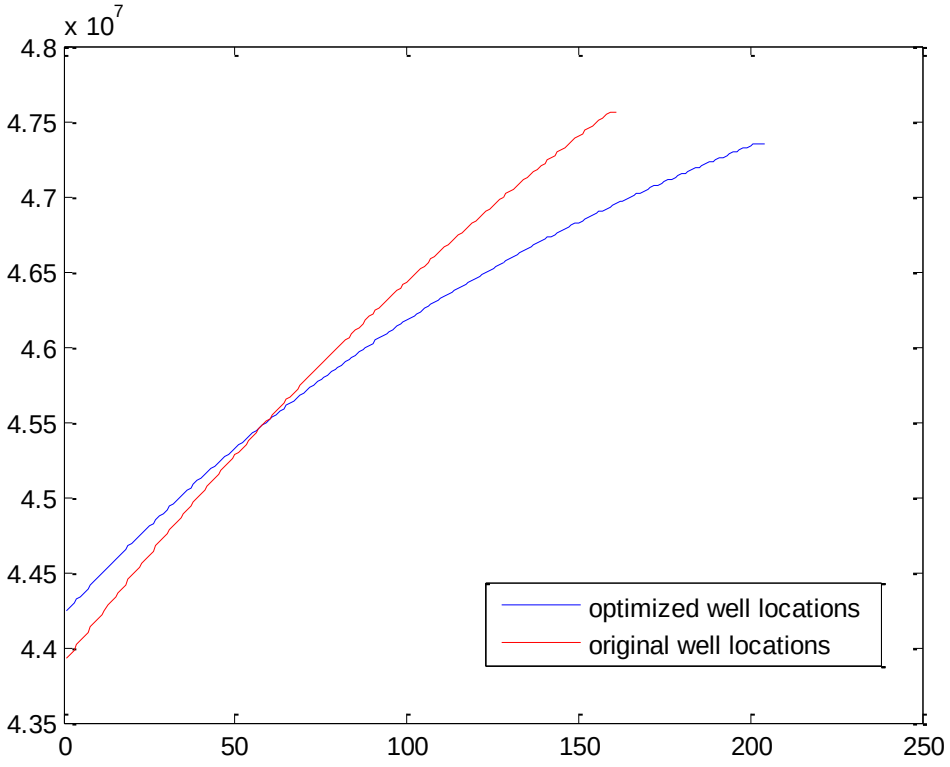


Figure 6.6 comparison figure

Compared to the original well placement, PSO can improve the output with initial well settings, however, the advantage of the optimization is weakened when the settings are also optimized. On the other hand, when the field property is known, determining the optimal well location is urgent and worthy.

Choose a random example when the wells are located like:

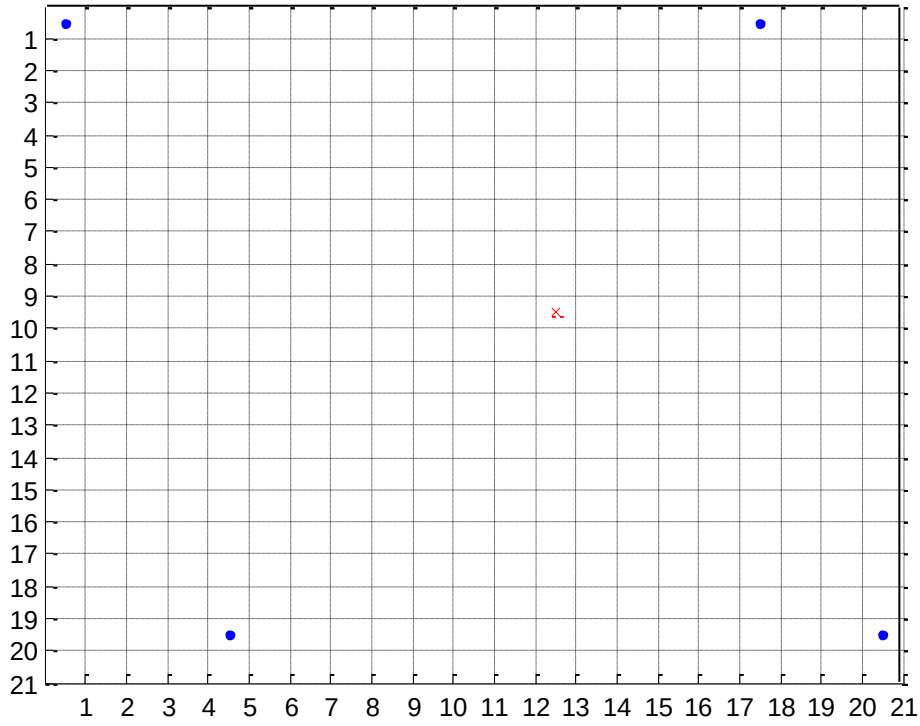


Figure 6.7 a random well location decision

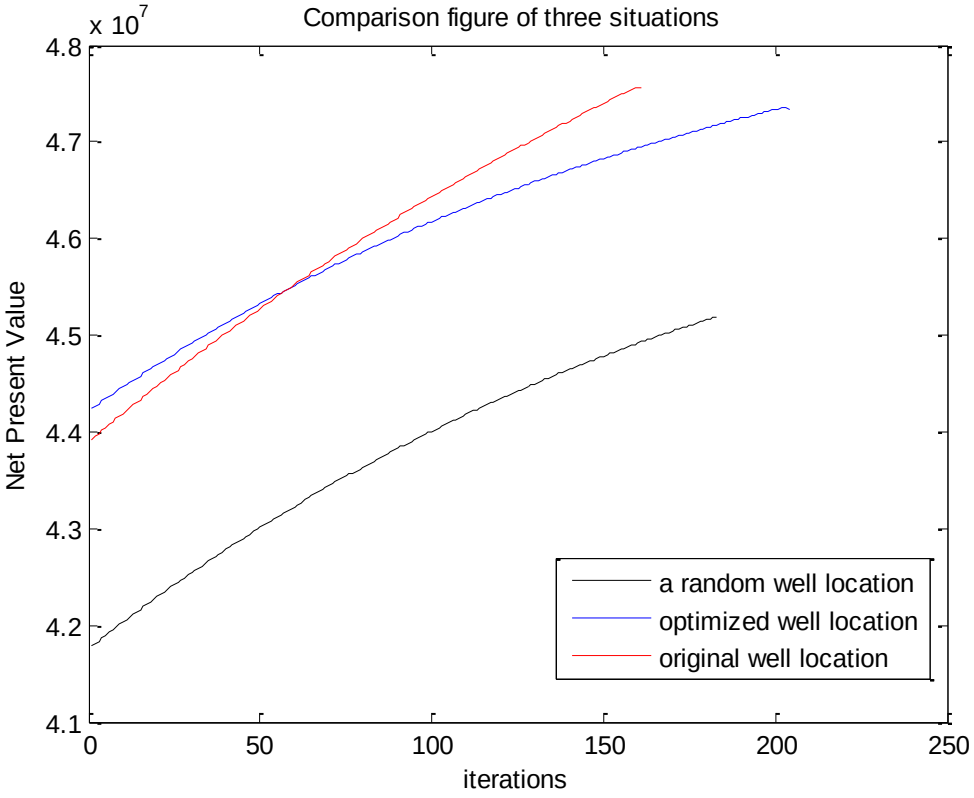


Figure 6.8 comparison figure

It can be seen from the above figure that well locations indeed influence the output a lot, consequentially, if wells are determined randomly, it could result in an economic loss. AS a result, when the true field is known, it is better to determine the well placement beforehand to make the reservoir more profited. A more effective way to get the NPV enhanced is by adjusting well settings.

7. Conclusions and Recommendations

The thesis focus on optimizing the output (NPV) in a reservoir field by two different ways: well placement and production settings, taking into account the uncertain of the reservoir models which can be estimated by using data assimilation. In this chapter, conclusions and recommendations are presented.

7.1 Conclusions

7.1.1 Estimation of field data with Asynchronous EnKF

For uncertain reservoirs, data assimilation is a very effective way to analyze the data of the field. Asynchronous EnKF provides an ensemble of updated reservoir realizations conditioned on the production data, as well as improved estimations of the model parameters, the state variables and the noise.

7.1.2 Optimal control of production settings

For optimal control of well settings, I presented a gradient-based method rather than a stochastic one, since the searching space is large and continuous. It works well with continuous and constrained optimization problem. I apply adjoint models for the efficient calculation of gradients. I also compared several different initial conditions and step size. The method is proved to be stable and efficient for the 21×21-grid reservoir model when the well placement is fixed.

7.1.3 Optimal control of production settings

When it comes to the optimization of well placement, I choose a stochastic searching method Particle Swarm Optimization instead of other ones since it is simple to implement, time-saving and efficient to solve the current problem. It is also proved that the type of production settings significantly affects the well placement problem. In other words, a well configuration that is optimal when the wells are operated with one type of settings maybe far from ones with another well setting.

7.2 Recommendations

Although I have discussed two different ways of improving economic outputs and they both can solve some problems, the research in this area still has a long way to

go. I recommend several aspects of further research. Firstly, since there are many sources of uncertainty in reservoir models, it is strongly recommended that this topic will be discussed in depth in order to be more close to reality. In addition, it is unclear which estimation technique is the most reliable one for history-matching of reservoir models. Secondly, in real reservoir management, the well settings are controlled step by step and will not be determined just at the beginning. Thirdly, the Particle Swarm Optimization method used in this thesis still needs improve with parameter selection and cost of the time to get the NPV enhanced to a higher stage. Fourthly, saving the computational load of a reservoir simulation would be very attractive and beneficial for practical application and further research. Fifthly, the model I use in this thesis is too constrained and we still need to do a lot of work to test other types of reservoirs.

Bibliography

[1].Abacioglu,Y.D.Oliver and A.Reynolds(2001). **Efficient reservoir history matching using subspace vectors**. *Computational Geosciences 5,151-172*

[2].Asheim,H.(1987). **Optimal control of water drive.** *SPE Journal(SPE 15978)*

[3].Athans,M.and P.L.Falb(1996). **Optimal Control-An introduction to the Theory and Its Applications.** *McGraw-Hill.*

[4].Andersson, E., Fisher, M.,Munro, R. and McNally, A. **Diagnosis of background errors for radiances and other observable quantities in a variational data assimilation scheme and the explanation of a case of poor convergence.** *Q. J. R. Meteorol. Soc., **126,** 1455–1472*

[5]Aziz K.and A.Settari(1979)**.Petroleum Reservoir Simulation.** *Applied Science Publishers.*

[6]Carla Cardinali, Sergio Pezzulli and Erik Andersson. **Influence-matrix diagnostic of a data assimilation system.** *Q. J. R. Meteorol. Soc. (2004), **130**, pp. 2767–2786*

[7]De Montleau,P.A.Cominelli,K.Neylon,D.Rowan,I.Pallister,O.Tesaker and I.Nygard(2006) .**Production optimization under constraints using adjoint gradients.** *In 10th European Conference on the Mathematics of Oil Recovery, Amserdam.*

[8]E. Ozcan, and C. K. Mohan. **Analysis of a Simple Particle Swarm Optimization System.** *Intelligent Engineering Systems Through Artificial Neural Networks,Vol. 8, 1998, pp. 253-258.*

[9]Evensen, G. 2007. **Data Assimilation: The Ensemble Kalman Filter**,*Springer, Berlin, 279.*

[10]Guyaguler,B. and R.N.Home(2004). **Uncertainty assessment of well-placement optimization.** *SPE Journal(SPE 87663)7(1),24-32*

[11]Handels,M.,M.J.Zandvliet,D.R.Brouwer and J.D.Jansen(2007). **Adjoint based well placement optimization under production constraints.** *In SPE Reservoir Simulation Symposium(SPE 105797),Houston.*

[12]Jansen, J. D., D. R. Brouwer, G. Naevdal and C. P. J. W. van Kruijsdijk (2005). **Closed-loop reservoir management.** *First Break 23, 43–48.*

[13]J. Kennedy. **The Behavior of Particles.** *The Seventh Annual Conf. on Evolutionary Programming,March 1998, pp. 581-591.*

[14] J. Kennedy, R. Eberhart. **Particle swarm optimization**, *Proceedings of the 1995 IEEE International Conference on Neural Networks,pp. 1942–1948.*

[15] J. Kennedy.**The particle swarm: Social adaptation of knowledge**. *Proceedings of the 1997 International Conference on Evolutionary Computation, pp. 303–308.*

[16]Kalman, R. E. (1963). **Mathematical description of linear dynamical systems.** *SIAM Journal of Control **1**(2), 152–192.*

[17]Kirk, D. E. (1970). **Optimal Control Theory - An Introduction**. *Prentice-Hall.*

[18]Kraaijevanger, J. F. B. M., P. J. P. Egberts, J. R. Valstar and H. W. Buurman (2007). **Optimal waterflood design using the adjointmethod.** *In SPE Resevoir Simulation Symposium (SPE 105764), Houston.*

[19]Leandro dos Santos Coelho, Cezar Augusto Sierakowski. **A software tool for**

teaching of particle swarm optimization fundamentals. *Industrial and Systems Engineering Graduate Program, Pontifical Catholic University of Parana' (PPGEPS/PUCPR), Curitiba, Parana' , Brazil*
[20]Montes, G., P. Bartolme and A. L. Udias (2001). **The use of genetic algorithms in well placement optimization.** *In SPE Latin American and Caribbean Petroleum Engineering Conference (SPE 69439), Buenos Aires.*
[21]Mehra, R.K., Davis, R.E. **A Generalized Gradient Method for Optimal Control Problems with Inequality Constraints and Singular Arcs.** *IEEE Transactions on Automatic Control, 17-1, 1972.*
[22]Naevdal, G., L. M. Johnsen, S. I. Aanonsen and E. H. Vefring (2005). **Reservoir monitoring and continuous model updating using ensemble kalman filter.** *SPE Journal (SPE 84372) 10(1), 66–74.*
[23]Ozdogan, U. and R. N. Horne (2006). **Optimization of well-placement optimization under time-dependent uncertainty.** *SPE Journal (SPE 90091) 9(2), 135–145.*
[24]Pavel Sakal, Geir Evensen and Laurent Bertino. **Asynchronous data assimilation with the EnKF.** *Tellus (2010), 62A, 24–29 C _ 2009 The Authors Journal compilation C _ 2010 Blackwell Munksgaard,Printed in Singapore*
[25]Rommelse, J. R., O. Kleptova, J. D. Jansen and A.W. Heemink (2006). **Data assimilation in reservoir management using the representer method and the ensemble kalman filter.** *In 10th European Conference on the Mathematics of Oil Recovery, Amsterdam.*
[26] R. Eberhart, and Y. Shi. ***Comparison between Genetic Algorithms and Particle Swarm Optimization.*** *Proceedings of the Seventh Annual Conf. on Evolutionary Programming,March 1998, pp. 611-619.*
[27]Sarma, P., L. J. Durlofsky, K. Aziz and W. H. Chen (2006b). **Efficient real-time reservoir management using adjoint-based optimal control and model updating.** *Computational Geosciences 10(1), 3–36.*
[28]Stengel, R. F. (1986). **Stochastic Optimal Control:** Theory and Application. *John Wiley& Sons*
[29]Sarma, P., Durlofsky L.J., Aziz, K., **Computational Techniques for Closed-Loop Reservoir Modeling with Application to a Realistic Reservoir**, *the 10th European Conference on the Mathematics of Oil Recovery, Amsterdam,Netherlands, 2006.*
[30]Van Doren, J. F. M., P. M. J. Van den Hof, J. D. Jansen and O. H. Bosgra (2008). **Determining identifiable parameterizations for large-scale physical models in reservoir engineering.** *In Proceedings of the 17th IFAC World Congress, Seoul.*
[31]van Essen, G. M., M. J. Zandvliet, P. M. J. Van den Hof, O. H. Bosgra and J. D. Jansen (2006a). **Robust optimization of oil reservoir flooding.** *In Proceedings of the 2006 IEEE International Conference on Control Applications, Munich.*
[32]van Hessem, D. (2004). **Stochastic inequality constrained closed-loop model predictive control**. *Ph. D. thesis, Delft University of Technology*
[33]Y. Shi, and R. Eberhart. **Parameter Selection in Particle Swarm Optimization.** *Proceedings of the Seventh Annual Conf. on Evolutionary Programming,March 1998, pp.591-601.*

[34]Zhang, F. and A. C. Reynolds (2002). **Optimization algorithms for automatic history matching of production data.** *In 8th European Conference on the Mathematics*
of Oil Recovery, Freiberg.
[35]Zhang, F. and A. C. Reynolds (2002). **Optimization algorithms for automatic history matching of production data.** *In 8th European Conference on the Mathematics*
of Oil Recovery, Freiberg.
[36]Zhou, K., J. C. Doyle and K. Glover (1996). **Robust and Optimal Control.** Prentice Hall.