THE UNIVERSITY OF TULSA
THE GRADUATE SCHOOL

PRODUCTION OPTIMIZATION IN THE CLOSED-LOOP RESERVOIR MANAGEMENT

by
Chunhong Wang

A thesis submitted in partial fulfillment of
the requirements for the degree of Master of Science
in the Discipline of Petroleum Engineeing

The Graduate School
The University of Tulsa

2007

T H E U N I V E R S I T Y O F T U L S A

THE GRADUATE SCHOOL

PRODUCTION OPTIMIZATION IN THE

CLOSED-LOOP RESERVOIR MANAGEMENT

by

Chunhong Wang

A THESIS

APPROVED FOR THE DISCIPLINE OF

PETROLEUM ENGINEEING

By Thesis Committee

_____, Co-advisor

Dr. Gaoming Li

_____, Co-advisor

Dr. Albert C. Reynolds

_____

Dr. Richard A. Redner

ABSTRACT

Chunhong Wang (Master of Science in Petroleum Engineeing)
Production Optimization in the Closed-Loop Reservoir Management
Directed by Dr. Gaoming Li and Dr. Albert C. Reynolds
84 pp., Chapter 6: Conclusions

(303 words)

Modern instrumentation providing essentially continuous data streams combined with smart well technology which allows for changes in the operational settings of wells gave rise to the concept of closed-loop reservoir management (CLRM). In CLMR, one periodically updates the reservoir models by integrating production data, and then solves an optimal control problem to determine optimum operating conditions to maximize production or net present value (NPV) for the remaining expected life of the reservoir. The cycle of updating and optimization is repeated through the whole production time of reservoir.

Here, we use the ensemble Kalman filter (EnKF) for reservoir model update to account for geological uncertainty. Three different optimization algorithms (EnKF, SPSA and steepest ascent method) for solution of the well control optimization are tried and compared. Two simple but representative synthetic examples, indicate that the steepest ascent method is the best of those tried. For the small case used for CLMR, it is shown that EnKF provides a reliable characterization of geological uncertainty by integrating production data. At the same time, it is also shown that NPV is a nonlinear function of the controls, the final controls from cases with both known true geology and uncertain geology present "Bang-Bang" behavior.

In the process of reservoir development, we always wish to drill wells at the optimal locations so that more hydrocarbons can be extracted at a lower cost. This can be treated as an optimization problem. Well locations in the reservoir simulator

are normally treated as discrete variables, gradient-based optimization algorithms can not be used for this problem. Here we present a novel method of converting the problem of optimizing on discrete variables into an optimization problem on continuous variables and use gradient based optimization algorithms for the optimal well placement. This method is tested and validated by two simple synthetic cases.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to Dr. Albert Reynolds and Dr. Gaoming Li for their support , guidance, insight and encouragement during my M.S. study. I would like to thank my master committee member, Dr. Richard A. Redner for his instructions, comments, critiques and suggestions.

Special appreciation is extended to Mrs.Judy Teal and Mrs.Loreta M. Watkins for their efficient administrative assistance and kindness during my study in University of Tulsa. I would like to extend my gratitude to all professors and graduate students of Petroleum Engineering department of the University of Tulsa for their course instruction and encouragement during my study.

I am thankful to my husband, Tao Gang, my son, Chengxi Gang for their understanding, encouragement and unconditional love over the last few years. This work is also dedicated to my parents Bingliang Wang and Guiling Zhang for their unselfish support and love during my M.S. study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Page

CHAPTER 1

INTRODUCTION

With the increase of world population and development of world economy, the demand for oil and gas is increasing every year. At the same time, the number of new significant discoveries of oilfields is decreasing greatly every year and most existing major oilfields are already on their later production stage. In order to satisfy the increasing worldwide demand for oil and gas, it is becoming more and more imperative to recover the existing reservoirs as efficiently as possible, while simultaneously decreasing development and operating costs.

Waterflooding is a popular secondary recovery method. In waterflooding, water is injected into the reservoir to displace oil towards producing wells to recover the remaining oil after primary recovery. However, because of heterogeneity of the reservoir, efficiency has a big impact on the recovery of water flooding. Since the injected water finds its way more easily through conductive fractures or high permeability channels, which will result in early water breakthrough, therefore it is crucial to consider the heterogeneity when we design a waterflooding project. The aim of this study is to improve the recovery efficiency of waterflooding by adjusting the well controls considering the geological heterogeneity and uncertainty. The well controls can be bottom hole pressure (BHP) of producers, injection rate of injectors or fluid production rate of producers etc.

1.1 Literature Review

In recent years, the concept of "closed-loop" reservoir management has attracted intensive research interest (3; 18; 33; 32). This approach enables one to adjust the reservoir production control parameters to optimize the reservoir production performance with geological uncertainty, while assimilating dynamic production

data in real-time. There are two optimization steps in this approach: the first step is the dynamic data assimilation (history matching), which is to reduce reservoir model uncertainty by integrating dynamic production data and the second step is to optimize the reservoir production performance by adjusting the well controls based on the history-matched uncertain reservoir models (production optimization). Studies in the literatures have been focusing on one of the steps of closed-loop reservoir management. A few researchers investigated the conjunction of the two. Brouwer (10) demonstrate how the combination of the ensemble Kalman filter for continuous model updating with automated adjoint-based water flood optimization algorithm leads to significant improvements in net present value (NPV) of the waterflooding process. Sarma (34) discussed a simplified implementation of the closed-loop approach that combines efficient optimal control and model-updating algorithms for real-time production optimization. An adjoint model is applied to determine optimal well setting and Bayesian inversion theory is used in combination with an optimal representation of the unknown parameter field in terms of a Karhunen-Loeve expansion.

For data integration problems of interest in reservoir modeling and characterization, Bayesian statistics provides a convenient framework for characterizing and evaluating geological uncertainty. The method introduced into reservoir characterization by Oliver et al.(30) and also considered by Kitanidis(19), which is now most commonly referred to as Randomized Maximum Likelihood (RML). RML has frequently been used to generate an approximate sampling of the pdf for a reservoir model conditional to production and/or seismic data(45; 46; 9). However, this method often takes the computational equivalent of 50 to 100 reservoir simulation runs to generate a single plausible reservoir model (realization) and its implementation requires an efficient adjoint code. The implementation of the adjoint method is not a trivial task and it appears that an optimal implementation of the adjoint can only be done with detailed knowledge of the reservoir simulator. It is impractical and time consuming for large scale and complicated reservoirs. In contrast, the ensemble Kalman filter (EnKF) method requires only one reservoir simulation

run per ensemble member. EnKF was proposed by Evensen(12) in the context of ocean dynamics literature as a Monte Carlo approximation of the Kalman filter. Since its introduction into the petroleum engineering literature(26; 23), it has been used by many researchers for assimilating production and seismic data to update reservoir variables including gridblock rock properties(15; 43), boundaries between facies(47) and initial fluid contacts(37). Ensemble Kalman filter (EnKF) is a sequential data assimilation algorithm. This sequential algorithm includes two steps: The first step is a forecast step which is equivalent to running the simulation to predict data at the next time step, the second step is an analysis or updating step which updates the reservoir model(s) considering the difference between predicted data and measurement data.

Although production optimization can be applied to any time of the reservoir life, most of the studies in this area focus on optimizing the reservoir performance under waterflooding(3; 18; 33; 32; 1). One of the reasons for this trend is because waterflooding is by far the most commonly used method to enhance oil recovery after primary depletion. The efficiency of a water flooding project relies largely on sweep efficiency, which strongly depends on the heterogeneity of the reservoir (i.e. the distribution of the high or low permeability streaks). Therefore, previous efforts on optimizing water flooding projects focus on controlling the water front by limiting the water injection into high permeability streaks, which will slow down water breakthrough into producers and increase oil recovery. The problem of production optimization requires the maximization of a certain objective function. Generally net present value (NPV) or accumulative oil production is employed as the objective function of production optimization problems. Asheim(2) investigated the optimization of the net present value (NPV) of water flooding with multiple vertical injectors and a vertical producer by rate allocation based on permeability-thickness product. Brouwer et al.(4) studied static water flooding optimization, in which they kept the inflow control valves constant during the displacement process until water breakthrough. Later, Brouwer et al.(3) explored the dynamic water flooding optimization. The gradient calculated with the adjoint method was used to dynamically optimize

the production performance with optimal control theory in an horizontal injector-producer system. In the paper, they consider the simple constraint where the total field injection rate is equal to the total field production rate. Sarma et al.(32) studied production optimization with adjoint gradient under nonlinear constraint. This paper compared different existing methods for nonlinear path constraints and focuses on the approximate feasible direction algorithm, which lumps all the nonlinear path constraints into one equation and hence requires only one adjoint for the constraint part. The implementation of the adjoint method is not an easy task, and it requires detailed knowledge of the reservoir simulator. To overcome this disadvantage, Lorentzen et al.(24) proposed to use EnKF as an optimization algorithm. Nwaozo(29) extended the concept and used an average gradient of net present value (NPV) to control variables from the ensemble of realizations. However, they did not consider constraints of the control variables in this optimization process.

Sudaryanto and Yortsos (41; 42) suggest that the optimal solution of waterflooding problems is a "Bang-Bang" control, i.e. each component of the control vector takes either its minimum or maximum allowed values. Zandvliet et al.(11; 44) investigated why and under what conditions waterflooding problems have optimal solutions at "Bang-Bang" control. They derived the sufficient and necessary conditions for "Bang-Bang" control optimal solutions and concluded that the waterflooding problems with simple upper and lower bound constraints where valve settings are the controls sometimes have "Bang-Bang" optimal solutions, while problems with other general inequality and/or equality constraints where rates are the controls will have a smooth optimal solution.

Production optimization not only includes the optimization of well control variables (bottom hole pressure, flow rate, valve setting etc.) but also incorporates the well placement optimization. In the process of reservoir development, the determination of the number and location of wells to be drilled is very crucial. Since the drilling cost of a new well is very expensive, the number of wells should be kept as small as possible to decrease the development cost. While at the same time, the ideal location of wells is also subject to optimization to achieve the high oil and/or

gas recovery. The determination of well location is very complicated since the well locations in the reservoir simulator are discrete variables. Therefore well placement optimization problem has a non-differential objective function and gradient-based optimization algorithms are not applicable. Generally, non-gradient based optimization algorithm is used for well placement optimization. A hybrid optimization algorithms based on generic algorithm (GA) was proposed by Baris at al.(13) to optimize placement of water-injection wells. Their result shows an increase of NPV of about $154 million after optimal location of three injectors has been determined. Burak at al.(40) have done some research about well placement optimization including well type and trajectory for nonconventional wells. They use GA algorithms combined with other advanced methods such as artificial neural network in the optimization process. Up to 30% increase in NPV has been achieved. But all the work has been done using non-gradient based optimization algorithms.

Recently Handels et al.(17) uses the gradient directly to solve the optimal well placement problem. The method of Handels et al. is explained most simply by considering the problem of determining the optimal location of a single infill well, e.g., the location of a new water injection well in a reservoir that already has completely-penetrating production and injection wells. Assuming flow is only in the $x - y$ plane, we use a 2D simulation grid. In this case, given the current proposed (initial condition) placement of the injection well, which is not in a grid block adjacent to the reservoir boundary, a "pseudo-well" produced at a low rate is placed in each of the eight "neighboring" grid blocks. Then the gradient of net present value (NPV) over the reservoir production life with respect to the rate at each pseudo-well is computed. The largest positive value among these eight gradient values determines the direction in which we should move the actual well to increase NPV the fastest, i.e., we should move the injection well in the direction defined by the line segment connecting the (x, y) coordinates of the center of the current well grid block to the (x, y) coordinates of the pseudo-well grid block corresponding to the largest gradient value.

1.2 Research Scope

In this work, we focus on production optimization under bound constraints on the control variables while geological description is uncertain. EnKF is implemented to update the reservoir geological models as data become available during production. This model updating step will typically reduce the geological uncertainty by integrating dynamic production data. Based on the updated geological model with reduced uncertainty, the control variables are optimized by a certain optimization algorithm to maximize net present value (NPV). Three different types of optimization algorithms (EnKF, SPSA and steepest ascent method) are considered and compared for this purpose. In this study, we consider bottom hole pressure (BHP) of producers as control variables with only upper and lower bound constraints. A simple synthetic case is used for the production optimization in the closed-loop reservoir management. By analysis of final results, we show that the updated permeability and porosity fields can capture the main geological features of the true fields. Moreover, the final control variables obtained from production optimization with geological uncertainty is similar to those obtained with the true geology.

A novel idea for a gradient based solution of the optimal well placement problem is presented here. We actually initialize the optimization problem by putting an injection well in every gridblock that does not contain a producing well. The injection rates of injectors are optimized by maximizing the net present value. Some wells are eliminated when well rate becomes zero. The net present value (NPV) is subtraction of the traditional NPV term and cost term of drilling injection wells. The steepest ascent algorithm is used to optimize injection rates of injector wells to maximize objective function over a specified reservoir lifetime. As the optimization proceeds, some injection well rates are decreased to zero and are removed from the system. The number of injection wells decreases during the optimization process. The optimal location of injection well(s) can be determined in the end.

CHAPTER 2

OPTIMIZATION ALGORITHMS

2.1 Problem Formulation

For production optimization problems, net present value (NPV) is generally used as an objective (cost) function. The well control variables are adjusted to maximize this objective function (NPV). Generally, net present value (NPV) is formulated as:

$$J = \sum_{n=1}^{N} \left[\sum_{j=1}^{N_{\text{prod}}} \left(r_o q_{o,j}^n - r_w q_{w,j}^n \right) - \sum_{l=1}^{N_{\text{inj}}} r_{w,inj} q_{inj,l}^n \right] \frac{\Delta t^n}{(1+b)^{t^n}} \tag{2.1}$$

where, N is the total number of simulation time steps; N_{prod} is the total number of producers; N_{inj} is the total number of injectors; r_o is oil revenue (\$/STB); r_w is water production cost (\$/STB); $r_{w,\text{inj}}$ is water injection costs (\$/STB); $q_{o,j}^n$ and $q_{w,j}^n$ are average oil and water production rates of the jth producer (STB/day) over the nth simulation time step, respectively; $q_{\text{inj},l}^n$ is the average injection rate of injector l (STB/day) over the nth simulation time step; b is annual interest rate (%); t^n is the cumulative time up to the nth simulation time step (year); Δt^n is the time interval of the nth simulation time step (day). For the production optimization problem considered here, we put zero cost on the water injection i.e. $r_{w,inj} = 0$, so the second term in the bracket of Eq. 2.1 is neglected.

Since oil and water production rates in the NPV of Eq. 2.1 are functions of the dynamic state vector x of the system, which includes pressure and saturation in every gridblock, as well as the well control vector u, the net present value (J) of Eq. 2.1 can be written as a function of x and u,

$$J = J[x, u]. \tag{2.2}$$

Where, control vector u has elements of control variables to be optimized.

For the production optimization problem, we wish to maximize the net present value J of Eq. 2.1 by adjusting the control vector u subject to the constraints shown in the following equations:

$$g[x, u, m] = 0 \tag{2.3}$$

$$Au \leq b \tag{2.4}$$

$$u^{\text{low}} \leq u \leq u^{\text{up}} \tag{2.5}$$

Eq. 2.3 represents the dynamic system, i.e., the reservoir simulation equation. Here m refers to model parameters (permeability, porosity, etc.). Eq. 2.4 and Eq. 2.5 give the constraints on the well control vector u, where u^{low} and u^{up} are lower and upper bounds of the control vector u. In this study, for simplicity, we only consider upper and lower bound constraints while ignoring all other inequality constraints.

With the upper and lower bounds of the well control variables, production optimization is a constrained optimization problem, which is hard to handle. One way to approach the problem is to transform the constrained optimization problem into a unconstrained optimization problem. One way to deal with the upper and lower bound of the control variables is to use a log-transformation as Gao et al.(14). For the ith control variable u_i, we define the transformed new variable s_i such that

$$s_i = \ln\Big(\frac{u_i - u_i^{\text{low}}}{u_i^{\text{up}} - u_i}\Big). \tag{2.6}$$

From Eq. 2.6, as u_i approaches its lower bound u_i^{low}, the transformed variable s_i approaches $-\infty$ and as u_i approaches its upper bound u_i^{up}, s_i approaches $+\infty$. If the upper and lower bounds are the only constraints on the control variables, the constrained optimization problem can be transformed to an unconstrained optimization problem using this log-transformation. When the log-transformation is applied

during optimization, all the operations are done in the transformed domain and the actual control variables are obtained using the inverse log-transformation:

$$\begin{aligned} u_i &= \frac{\exp(s_i)u_i^{\text{up}} + u_i^{\text{low}}}{1+\exp(s_i)} \\ &= \frac{u_i^{\text{up}} + u_i^{\text{low}}\exp(-s_i)}{1+\exp(-s_i)} \end{aligned} \tag{2.7}$$

2.2 Optimization Algorithms

For the production optimization problems, various optimization algorithms have been investigated by many researchers. Two major categories of optimization algorithms have been developed: gradient-based algorithms and stochastic algorithms. Gradient-based algorithms require an efficient technique to calculate the gradient of the objective function with respect to control variables. Gradient-based methods using adjoint equations require explicit knowledge of the simulation model equations used to describe the dynamic system. Another way to obtain the gradient is by the finite difference method. However, practical production optimization problems typically involve large, highly complex reservoir models, thousands of unknowns and many nonlinear constraints, which make the numerical calculation of the gradients for the optimization process impractical. On the other hand, the stochastic algorithms such as genetic algorithm and simulated annealing require many forward model evaluations but are theoretically capable of finding a global optimum with a sufficiently large number of simulation runs. Relationship between the objective function and control variables can be obtained from several forward models. However, this method becomes inefficient when the number of control variables are large.

In our research, we test three kinds of optimization algorithms for production optimization: steepest ascent method, SPSA and EnKF. In the steepest ascent method, since we do not have adjoint code to obtain the gradient of the objective function with respect to control variables, we temporarily use the finite-difference method to derive the gradient. Simultaneous Perturbation Perturbation Stochastic Approximation (SPSA), as its name implies, simultaneously perturbs all the variables to calculate a stochastic gradient. The expectation of the stochastic gradient

from SPSA is the true gradient. For ensemble Kalman filter (EnKF), the average gradient is derived from the ensembles using statistics.

2.2.1 Steepest Ascent Method

Steepest ascent method finds uphill direction to maximize the objective function. For the steepest ascent method, we use the finite-difference method (perturbation method) to calculate the gradient. To obtain the gradient of the net present value J with respect to control variable u_i, we use the central two-sided perturbation:

$$\frac{dJ}{du_i} = \frac{J(u)|_{u_i+\delta u_i} - J(u)|_{u_i-\delta u_i}}{\delta u_i},\ i = 1, \ldots, N_u, \tag{2.8}$$

where N_u is the total number of control variables and δu_i is the perturbation size. Since control variable is perturbed two times to get the gradient and each gradient calculation requires two simulation runs to evaluate $J(u)|_{u_i+\delta u_i}$ and $J(u)|_{u_i-\delta u_i}$, therefore, the method is not applicable when there is a large number of controls to adjust, although it is easy to be coupled with any reservoir simulator. However, when the control variable reaches its bound, one-sided perturbation is employed to calculate the gradient. In this study, our focus is not to find an efficient way for gradient calculation, but to compare the steepest ascent method to other optimization algorithms assuming one can obtain the gradient efficiently.

After the gradient of the net present value to all the control variables are calculated using Eq. 2.8, we maximize the objective function (J) with the steepest ascent method:

$$u^{k+1} = u^k + \alpha_k \nabla_{u^k} J, \tag{2.9}$$

where α_k is the step size for the kth iteration. In Eq. 2.9, $\nabla_{u^k} J$ is the gradient of the net present value J with respect to the actual control vector u evaluated at u^k, i.e.

$$\nabla_{u^k} J = \Big[\frac{dJ}{du_1}, \ldots, \frac{dJ}{du_{N_u}}\Big]^T_{u^k}. \tag{2.10}$$

When there are upper and lower bounds on the control variables, we use

log-transformation (Eq. 2.6) to eliminate this linear constraint and do production optimization based on the transformed domain. The gradient of the net present value (J) to the transformed control variable s_i is calculated using Eq. 2.11 similar to Eq. 2.8

$$\frac{dJ}{ds_i} = \frac{J(s)|_{s_i+\delta s_i} - J(s)|_{s_i-\delta s_i}}{\delta s_i},\ i = 1, \ldots, N_u, \tag{2.11}$$

where, δs_i is the perturbation size on the transformed control variable. Since the log-transformation is a nonlinear transformation, attention needs to be paid to the choice of the perturbation size of the transformed control variables, especially when the actual control variable is close to its boundary. A small perturbation of the control variable when it is close to the boundary will give a negligible change on the actual control, which hence results in no change on the net present value and zero gradient. For this reason, the perturbation size on the transformed variable is calculated using a fixed perturbation size on the actual control variable. Based on experiments, we choose 0.01 psi for the perturbation of the BHP controls of producers in the following example. This ensures that each perturbation on the transformed variable influences the net present value.

The transformed control vector is updated with steepest ascent formulation:

$$s^{k+1} = s^k + \alpha_k \nabla_{s^k} J, \tag{2.12}$$

After the transformed control vector is updated, it is transformed back to the actual control variable before it is input into the simulator for net present value evaluation.

When the optimization is done in the transformed space, a trial step size is used in Eq. 2.12 to update the transformed control vector s, followed by a quadratic fit to determine α_k. If objective function $J(s^{k+1})$ increases with the α_k determined by quadratic fit, we accept this α_k as step size for kth iteration and go to the next iteration. If the objective function $J(s^{k+1})$ does not increase with this step size, α_k is cut by half until the objective $J(s^{k+1})$ is greater than $J(s^k)$.

2.2.2 Simultaneous Perturbation Stochastic Approximation (SPSA)

The SPSA algorithm uses stochastic simultaneous perturbation of all model parameters to generate a search direction at each iteration. Although the direction appears to be random, it can be chosen so that it is downhill for history matching or uphill for production optimization problems. Most importantly, the expectation of the search direction is the steepest descent direction. SPSA algorithm also can be easily combined with any reservoir simulator and easy to implement. Explicit knowledge of the simulator is not required.

In the previous section, we have calculated the gradient of NPV with respect to control variables using a finite difference method, since we do not have the adjoint code coupled with the reservoir simulator Eclipse. As we mentioned, this method is very time-consuming due to the fact that each perturbation requires at least one simulation run. SPSA is a compromised solution to the finite-difference method, in which all the parameters are perturbed at one time stochastically. As a result of this stochastic perturbation, the calculated gradient is also stochastic, however, it can be chosen so that it is downhill direction for history matching or uphill direction for production optimization problems. Most importantly, the expectation of the stochastic gradient is the true gradient. Since all the parameters are perturbed together, SPSA only requires two simulation runs for the one-sided perturbation and three for the central difference, which greatly saves the number of simulation runs compared to the finite-difference gradient calculation of the previous section. Since all the problems we are dealing with here have upper and lower bounds, we optimize on the log-transformed control vector s.

The stochastic gradient is calculated using a central difference based on simultaneous perturbation of the transformed control vector s as follows,

$$\hat{g}_k(s^k) = \frac{J(s^k + c_k\Delta_k) - J(s^k - c_k\Delta_k)}{2c_k} \times \Delta_k^{-1}, \tag{2.13}$$

where Δ_k is an N_u dimensional random column vector,

$$\Delta_k = [\Delta_{k,1}, \Delta_{k,2}, \ldots, \Delta_{k,N_u}]^T, \tag{2.14}$$

and Δ_k^{-1} is defined as

$$\Delta_k^{-1} = \left[\Delta_{k,1}^{-1}, \Delta_{k,2}^{-1}, \ldots, \Delta_{k,N_u}^{-1}\right]^T, \tag{2.15}$$

and $\Delta_{k,i}, i = 1, 2 \cdots N_u$ represents independent samples from the symmetric ± 1 Bernoulli distribution. This means that $\Delta_{k,i}$ can only take either $+1$ or -1 and the probability of taking each value is 0.5, so the expectation of $\Delta_{k,i}$ is zero ($E[\Delta_{k,i}] = 0$). Note that due to the fact that $\Delta_{k,i}$ can only take either $+1$ or -1, $\Delta_k^{-1} = \Delta_k$. In Eq. 2.13, c_k is a positive coefficient, which controls the size of perturbation and is chosen in the same way as the perturbation size in the finite-difference method of the last section.

Similar to the steepest ascent method, the transformed control vector s is updated at the kth iteration using the following equation:

$$s^{k+1} = s^k + \alpha_k \hat{g}_k(s^k). \tag{2.16}$$

Note that $\hat{g}_k(s^k)$ is a random vector due to the fact that Δ_k is a random vector, therefore Eq. 2.16 seems to be a random search direction, however this random direction is always uphill from Eq. 2.13. If $\Delta_k^{-1} = \Delta_k$ is an uphill direction, then the scalar $\frac{J(s^k + c_k\Delta_k) - J(s_k - c_k\Delta_k)}{2c_k}$ is positive, so $\hat{g}_k$ has the same direction as Δ_k^{-1}, the uphill direction. If $\Delta_k^{-1} = \Delta_k$ is an downhill direction, then the scalar $\frac{J(s^k + c_k\Delta_k) - J(s_k - c_k\Delta_k)}{2c_k}$ is negative, therefore, $\hat{g}_k$ and Δ_k^{-1} are in the opposite directions, i.e. $\hat{g}_k$ has the uphill direction. .

Because the expectation of the stochastic gradient $\hat{g}_k(s^k)$ is the true gradient, we also tried to use the average of several samples of the stochastic gradient by

generating samples of Δ_k as a search direction, i.e.

$$s^{k+1} = s^k + \alpha_k \overline{\hat{g}_k(s^k)}, \tag{2.17}$$

where $\overline{\hat{g}_k(s^k)}$ is defined as,

$$\overline{\hat{g}_k(s^k)} = \frac{1}{M} \sum_{i=1}^{M} \hat{g}_i(s^k), \tag{2.18}$$

with M samples of the stochastic gradient.

When the transformed control vector is updated using a single stochastic gradient of Eq. 2.16, α_k is determined by a quadratic fit along the stochastic gradient direction with three points $J(s^k - c_k\Delta_k)$, $J(s^k)$ and $J(s^k + c_k\Delta_k)$. If the net present value does not increase with fitted step size, we simply reject this step, generate a new Δ_k, and repeat the above process until the increase in net present value is achieved.

When the average stochastic gradient is used to update the transformed control vector s with Eq. 2.17, the same line search used for the steepest ascent method is applied.

2.2.3 Ensemble Kalman Filter (EnKF)

For the closed-loop reservoir management, the ensemble Kalman filter (EnKF) is usually used to update the geological fields by integrating dynamic production data. However, some people have used it as an optimization algorithm for production optimization purposes. We first give a short introduction of ensemble Kalman filter (EnKF) as a data assimilation method, followed by its application in production optimization.

The ensemble Kalman filter (EnKF) has recently gained popularity as a method for updating reservoir models by assimilating dynamic production data. Several studies show that the ensemble Kalman filter (EnKF) is a promising alternative to other traditional history matching methods. EnKF is a Monte-Carlo method in which an ensemble of initial reservoir models are generated by sampling

from the prior probability density function and these models are kept up-to-date as data are assimilated sequentially. Zafari and Reynolds(43) have shown that EnKF is approximately equivalent to one iteration of the Gauss-Newton method with an average sensitivity of the production data to the model parameters calculated using statistics.

The process consists of two steps : a forecast step (giving Y^f) and an analysis step (giving Y^a). Here Y represents the ensemble of state vector and its jth column is defined by:

$$y_j = \begin{bmatrix} m_j \\ p_j \\ d_j \end{bmatrix} \tag{2.19}$$

Here m_j is state vector and includes model parameters, such as gridblock permeabilities and porosities, N_m is the dimension of the model vector, p_j is the vector of primary variable, including pressure, saturation and dissolved GOR, N_p is dimension of the primary variable for every ensemble, d_j refers to predicted data from the jth ensemble. Similarly, state vector Y is:

$$Y = \begin{pmatrix} y_1 & y_2 & \cdots & y_{N_e} \end{pmatrix} \tag{2.20}$$

Here N_e is the number of ensembles.

A numerical reservoir simulator is used to perform the forecast step to propagate the state vectors (ensemble) from time step $n-1$ to time step n:

$$y^f_{n,i} = f(y^a_{n-1,i}), \tag{2.21}$$

for $i = 1, 2, \cdots$,N_e .

In the analysis step, the forecast state vectors y^f_n are updated by taking into account the mismatch between measurement data and the corresponding predictions from the ensemble members.

The analyzed state vector are now computed as:

$$y^a_{n,i} = y^f_{n,i} + K_n(d_{obs,n,i} - H_n y^f_{n,i}), \tag{2.22}$$

and

$$K_n = C^f_{Y,n} H^T_n (H_n C^f_{Y,n} H^T_n + C_{D,n})^{-1}. \tag{2.23}$$

$$H = \begin{bmatrix} O & I_{N_d} \end{bmatrix} \tag{2.24}$$

Where H_n is a matrix that is a trivial matrix with only 0 and 1 as its components and selects measured variables from the state vector. $C_{D,n}$ is the data measurement error covariance matrix and is diagonal matrix. $C^f_{Y,n}$ is the covariance matrix for the state vectors and can be estimated from the ensemble using the standard statistical formula:

$$C_Y = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (y^f_{n,i} - \overline{y}^f_n)(y^f_{n,i} - \overline{y}^f_n)^T \tag{2.25}$$

Here, $\overline{y^f_n}$ represents the average of the ensemble at the forecast time step n and is calculated by:

$$\overline{y}^f_n = \frac{1}{N_e} \sum_{i=1}^{N_e} y^f_{n,i} \tag{2.26}$$

Recently the ensemble Kalman filter (EnKF) technique has been adapted to maximize the net present value (NPV) for production optimization problems (29). It is a modified form of the ensemble Kalman filter (EnKF) used in history matching. This approach does not require the solution of the adjoint equations. No knowledge of the simulator equations is required and the simulator is run as a black box. The relationship between the objective function (NPV) and the set of control variables is obtained from the ensemble of realizations of the state vector. In this study, an ensemble of N_e realizations of the controls was generated and continuously updated after each reservoir simulation run until optimum control variables were obtained.

The optimum control variables are the control variables at which the net present value (NPV) is at its maximum.

Let u be a random vector for the well controls. Assuming the random vector follows a Gaussian distribution, its mean at the kth iteration can be approximated by N_e realizations

$$\overline{u}^k = \frac{1}{N_e}\sum_{j=1}^{N_e} u_j^k, \tag{2.27}$$

and its covariance is given by

$$C_{u^k} = E[(u^k - \overline{u}^k)(u^k - \overline{u}^k)^T], \tag{2.28}$$

which can be approximated using the N_e samples of the control vector u by

$$C_{u^k} = \frac{1}{N_e - 1}\sum_{j=1}^{N_e}\left[(u_j^k - \overline{u}^k)(u_j^k - \overline{u}^k)^T\right]. \tag{2.29}$$

Notice that the subscript j represents the jth sample for the control vector instead of the jth element of the control vector. With each sample of the control vector u_j^k, we evaluate the net present value $J(u_j^k)$. From Taylor's series following Zafari and Reynolds(43), $J(u_j^k)$ is approximated by,

$$J(u_j^k) = J(\overline{u}^k) + \overline{\nabla J}^T(u_j^k - \overline{u}^k),\ j = 1, 2, \ldots, N_e, \tag{2.30}$$

where $\overline{\nabla J}$ is the gradient of the net present value with respect to the control vector evaluated at $\overline{u}^k$. The mean of the net present value is estimated by

$$\overline{J}^k = \frac{1}{N_e}\sum_{j=1}^{N_e} J(u_j^k). \tag{2.31}$$

As in Zafari and Reynolds(43), we use the following approximation

$$\overline{J}^k = J(\overline{u}^k). \tag{2.32}$$

The covariance between the control vector u and the net present value J at the kth iteration can be approximated using the N_e samples of the control vector and the corresponding net present values as follows,

$$C_{u^k J} = \mathrm{cov}(u^k, J(u^k)) \approx \frac{1}{N_e - 1} \sum_{j=1}^{N_e} \left[(u_j^k - \overline{u}^k)(J(u_j^k) - \overline{J}^k)^T \right]. \tag{2.33}$$

Using Eqs. 2.30 and 2.32 in Eq. 2.33 gives

$$C_{u^k J} \approx \frac{1}{N_e - 1} \sum_{j=1}^{N_e} \left[(u_j^k - \overline{u}^k)(u_j^k - \overline{u}^k)^T \right] \overline{\nabla J}. \tag{2.34}$$

Substitution of Eq. 2.29 into Eq. 2.34 yields

$$C_{u^k J} \approx C_{u^k} \overline{\nabla J}. \tag{2.35}$$

The average gradient can be obtained by

$$\overline{\nabla J} \approx C_{u^k}^{-1} C_{u^k J}, \tag{2.36}$$

where C_{u^k} is approximated using Eq. 2.29 and $C_{u^k J}$ is approximated using Eq. 2.33. If C_{u^k} is rank deficient, its pseudo inverse is calculated using singular value decomposition(21). For production optimization, we use this approximate average gradient for control vector update. The well control vector can be obtained by

$$u^{k+1} = u^k + \alpha_k \overline{\nabla J}, \tag{2.37}$$

where the step size α_k is determined in the same way as in steepest ascent method. For production optimization using the ensemble Kalman filter (EnKF), log-transform is not used to delete the upper and lower bound of the well control variables. If updated control variable is greater the upper bound, we just set the updated control variable equal to upper bound. If the updated control variable is below the lower

bound, we set the updated control variable equal to lower bound.

CHAPTER 3

PRODUCTION OPTIMIZATION WITH TRUE GEOLOGY

In the previous chapter, we have given a very detailed introduction about three optimization algorithms (steepest ascent method, SPSA and EnKF). Next, we mainly focus on comparing three different optimization algorithms assuming the true geology is known. Three optimization algorithms have been applied respectively to two synthetic five-spot waterflooding reservoirs to do production optimization. Final production optimization results are compared and the best optimization algorithm for production optimization is chosen according to the magnitude of final net present value (NPV) and convergence speed of optimization algorithms.

3.1 Case 1

3.1.1 Reservoir Description

A synthetic 2-D reservoir with no flow boundaries is used for production optimization. The reservoir consists of 11 by 11 grid blocks and the grid block system is uniform with $\Delta x = \Delta y = 200$ ft. The depth of the top surface of the reservoir is 7800 ft with a net pay thickness of 10 ft. The true porosity and permeability fields are shown in Figs. 3.1 and 3.2. The true porosity and permeability fields are generated using Sequential Gaussian Cosimulation algorithm in the Geostatical Software Library (GSLIB) (27) assuming log-normal distribution. The variogram parameters for the permeability and porosity distribution are shown in Table3.3.

The reservoir is under five-spot waterflooding, with one injector located at the center of the reservoir (6,6) and four producers placed at the four corners as shown in the permeability distribution map of Fig. 3.1. There are two high permeability channels in the reservoir: one runs from the lower left corner (P1) to the top right

Table 3.1: Geostatistical parameters of permeability and porosity for case 1

$(\ln k)_{mean}$	0.2
$(\ln k)_{mean}$	4.0
σ_ϕ	0.0025
$\sigma_{\ln k}$	0.08
$\rho_{\phi,\ln k}$	0.8
α	40
r_1 (ft)	40.0
r_2 (ft)	2.0

corner (P4); one is a short channel at the top left corner. The high permeability channel connecting the injector to the two producers P1 and P4 will cause early water breakthrough in these two wells. There is a low permeability channel at the lower right corner between the injector and the producer P2, which will slow water movement towards P2. In the time period considered, there is no water breakthrough at that producer. The water breakthrough time in P3 is after that of P1 and P4. Similar features are shown in the porosity distribution since we use a correlation of 0.8 between porosity and log-permeability to generate these two fields. During the water flooding project, we optimize the bottom hole pressure (BHP) of four producers while keeping the water injection rate constant at 1000 STB/day. The anticipated water flooding project life is 960 days and we set the control step for the producers to 120 days, so there are 8 control steps and 32 maximum number of controls for production optimization. All the producers are at BHP control with an upper bound of 6000 psi and a lower bound of 400 psi. There are only two phases in the reservoir: water and oil. No free gas was present in the reservoir. In the example, the following parameters are used: $r_o = 50\$/\text{STB}$, $r_w = 15\$/\text{STB}$ and $b = 20\%$ to calculate the objective function (NPV).

The relative permeability curve used in the reservoir model is shown in Fig. 3.3. The reservoir properties are listed in Table 3.2.

3.1.2 Initial Bottom Hole Pressure

In order to compare three different optimization algorithms (steepest ascent

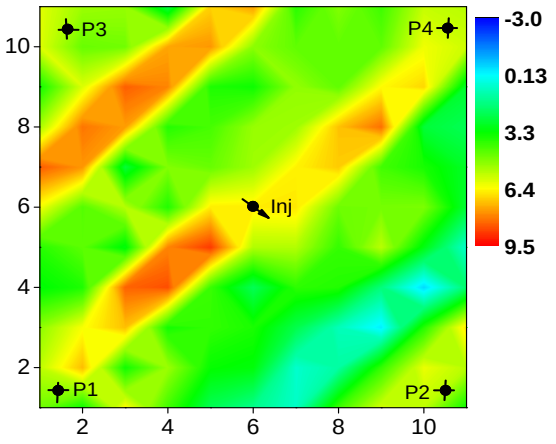


Figure 3.1: True horizontal log-permeability distribution for case 1.

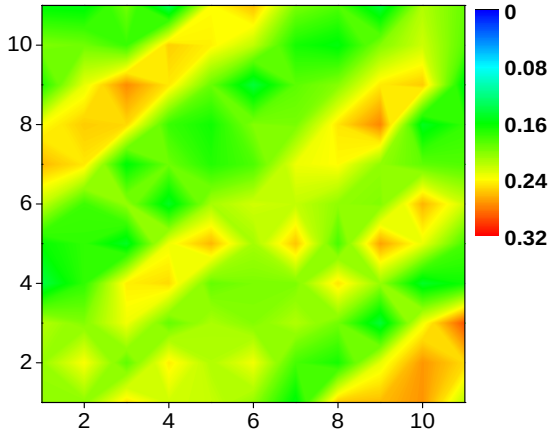


Figure 3.2: True porosity distribution for case 1.

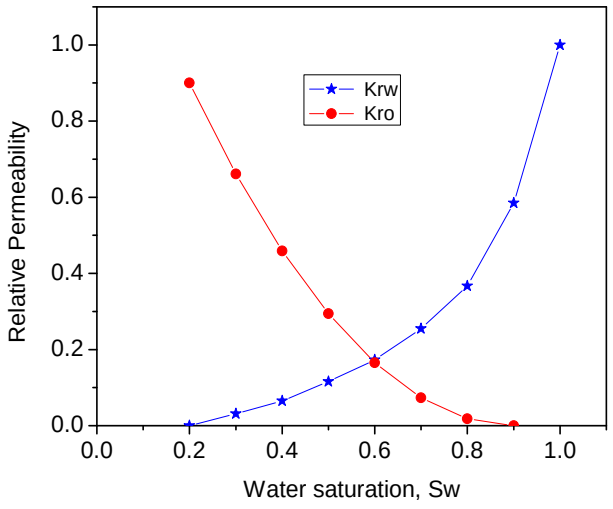


Figure 3.3: Water and oil relative permeability curve for case 1.

method, SPSA and EnKF), first we need to generate initial bottom hole pressure (BHP) profile (initial controls) for every optimization algorithm. For the steepest ascent method and SPSA, we set the initial bottom hole pressure (BHP) of four producers all equal to 1000 psi. For the EnKF method, similar to Nwaozo(29), we generate $N_e = 40$ realizations of the initial BHP for four producers using the following steps:

(i) The mean BHP for each realization of each well is independently sampled from a uniform distribution between 1000 psi and 5500 psi. Figs. 3.4 to 3.7 show the histogram of mean for the realizations of each producer.

Table 3.2: Reservoir properties for case 1

Grid block size	200 ft
Thickness	10 ft
ρ_{osc}	56.93 lb/ft^3
ρ_{wsc}	62.428 lb/ft^3
μ_o	1.46 cp
μ_w	0.5 cp
B_o	1.2
B_w	1.0042 rb/stb
Rock Compressibility	3.103×10^{-5} psi^{-1}
Top Depth	7800 ft
Residual oil saturation	0.1
Irreducible water saturation	0.2
Initial water saturation	0.2
Initial reservoir pressure	3500 psi
Total production period	960 days
Control step length	120 days

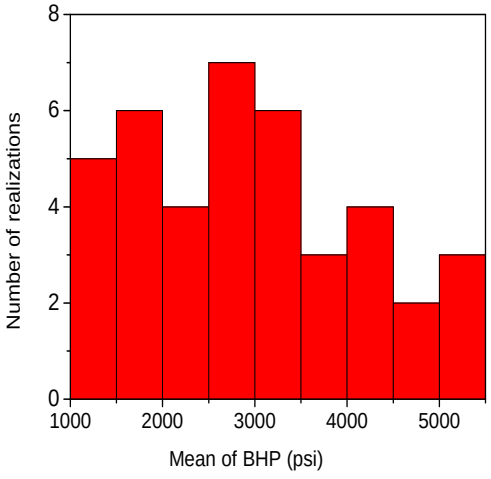


Figure 3.4: P1 initial distribution of mean of BHP for case 1.

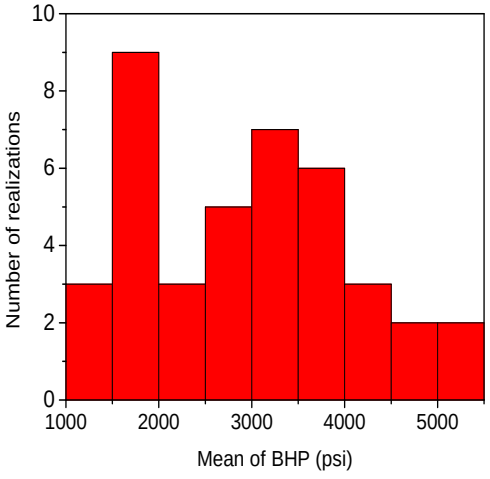


Figure 3.5: P2 initial distribution of mean of BHP for case 1.

(ii) With each realization of the mean of each well from step i, the BHP distribution as a function of time is generated by sampling a Gaussian distribution with the prescribed mean and the following covariance function:

$$C_{i,j} = \sigma^2 \exp\left[\frac{-3|i-j|}{a}\right] \tag{3.1}$$

where σ is the standard deviation (200 psi for this case); a is the correlation range

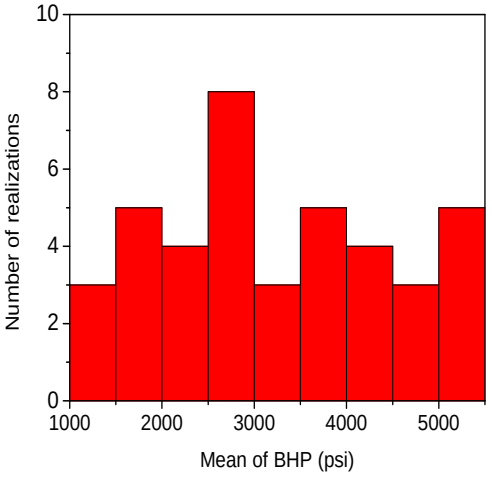


Figure 3.6: P3 initial distribution of mean of BHP for case 1.

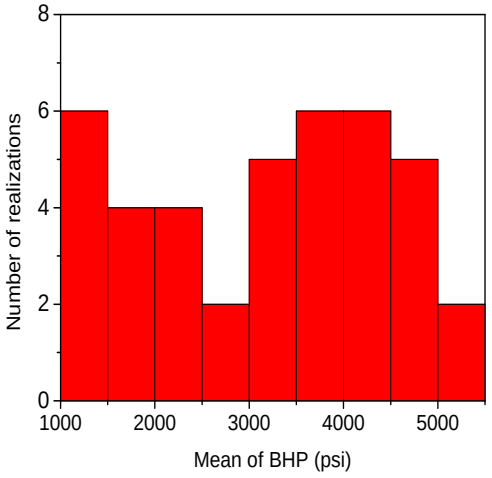


Figure 3.7: P4 initial distribution of mean of BHP for case 1.

(5 control steps are used); i, j are the control step indices.

Figs. 3.8 to 3.11 show the histogram of initial bottom hole pressure controls for all four producers.

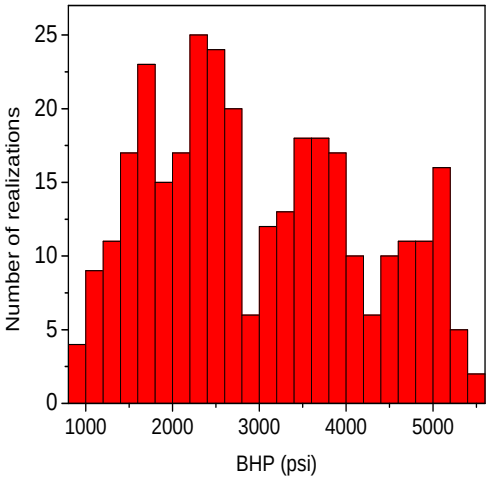


Figure 3.8: Histogram for initial BHP realizations of P1 for case 1.

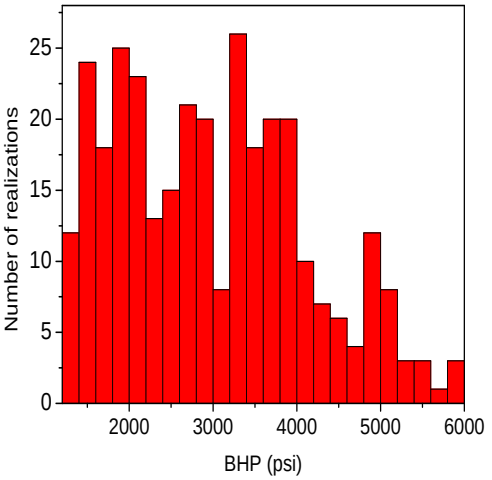


Figure 3.9: Histogram for initial BHP realizations of P2 for case 1.

To show how the initial BHP realizations change as a function of the time, we select the first ten realizations of BHP for each producer P1, P2, P3 and P4 from the N_e realizations and display them in Figs. 3.12 to 3.15. As mentioned earlier, each BHP profile follows a Gaussian distribution with its mean sampling from a uniform

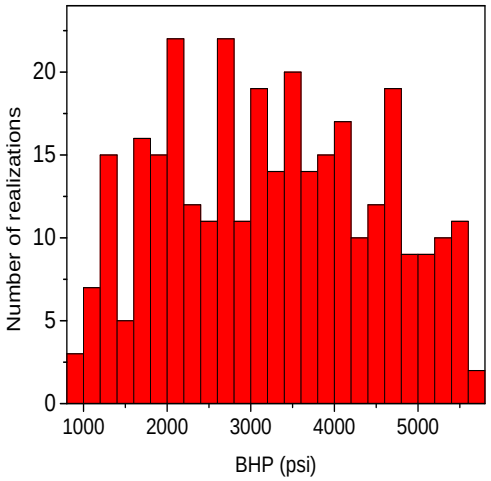


Figure 3.10: Histogram for initial BHP realizations of P3 for case 1.

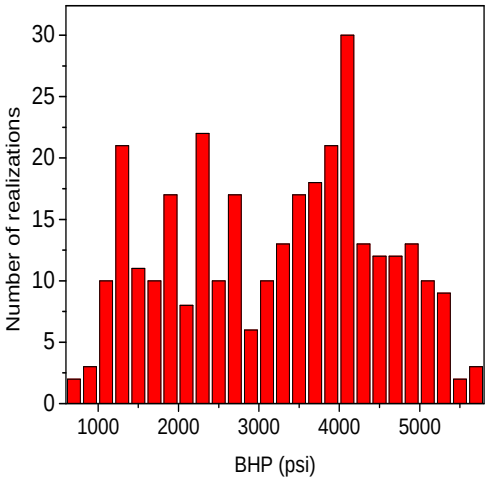


Figure 3.11: Histogram for initial BHP realizations of P4 for case 1.

distribution, variance of 200 psi, and correlation range of 5 as in Eq. 3.1.

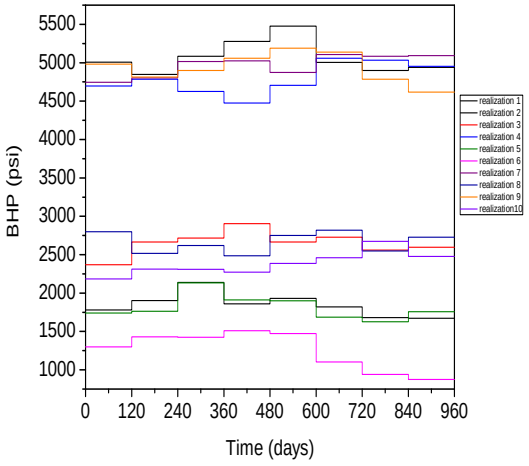


Figure 3.12: Ten realizations of initial BHP profile for P1 for case 1.

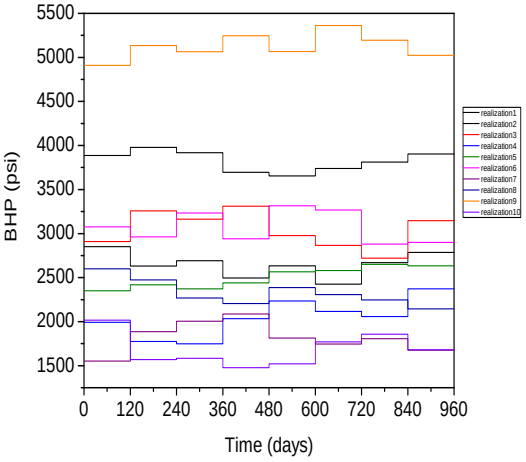


Figure 3.13: Ten realizations of initial BHP profile for P2 for case 1.

3.1.3 Comparison of Optimization Algorithms

As described earlier, the ultimate objective of the production optimization is to determine the optimum production scheme for oil and gas fields. This can be achieved by combining a reservoir simulator with a numerical optimization algorithm. The reservoir simulator in this study is Eclipse 100 from Schlumberger. In

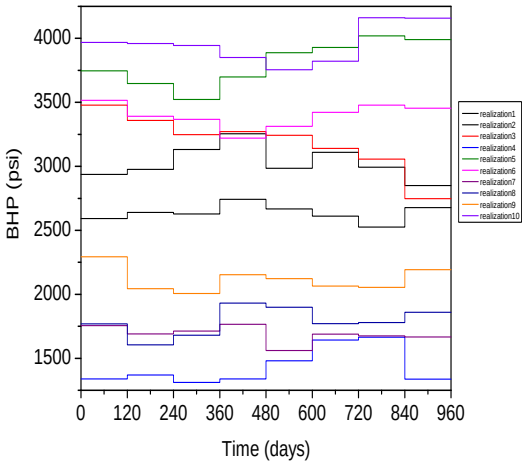


Figure 3.14: Ten realizations of initial BHP profile for P3 for case 1.

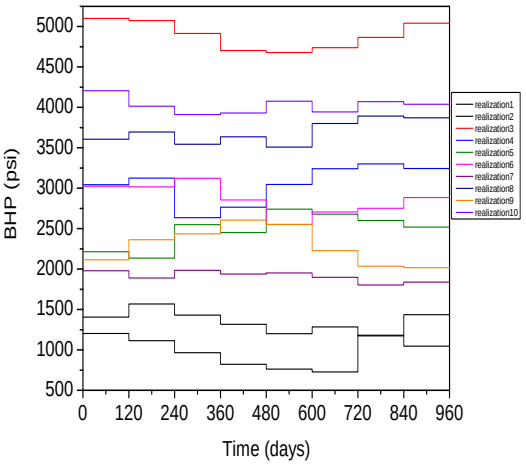


Figure 3.15: Ten realizations of initial BHP profile for P4 for case 1.

this part, we mainly do production optimization based on the small synthetic reservoir introduced in the previous section using three different optimization algorithms.

Fig. 3.16 and Fig. 3.17 respectively show the NPV changes with iteration number and number of simulation runs for different optimization algorithms. For steepest ascent method, we assume that we have adjoint code to calculate the gradient. The curve with squares shows the performance from the steepest ascent method. The algorithm converges in 5 iterations and NPV increases from $\$1.34\times10^7$ to $\$1.59\times10^7$. Only 25 simulation runs are needed. The curve with circles is the performance using the average of ten stochastic gradients from SPSA. Although this method converges to the same NPV as the steepest ascent method for the same initial guess, the convergence is slower than the steepest ascent algorithm and the number of simulation runs is much more than that of steepest ascent method. The algorithm with a single stochastic gradient from SPSA (curve with crosses) converges to an NPV of $\$1.58\times10^7$ in 80 iterations, a value only slightly lower than the value obtained from the steepest ascent method and the average SPSA gradient method. However, considering that the average SPSA gradient method uses 10 times the number of simulation run to calculate the gradient, the efficiency of the two SPSA gradient methods is about the same. The performance from EnKF is shown by the

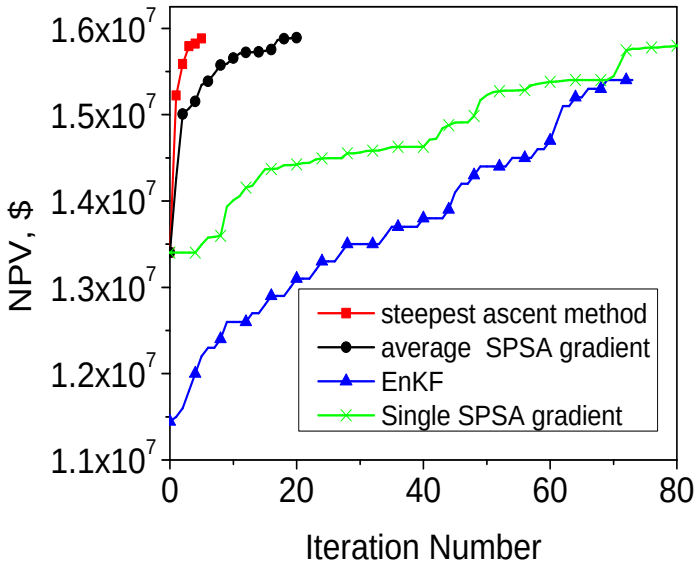


Figure 3.16: NPV as a function of iteration number for case 1.

curve with triangles. Since there are N_e realizations of BHP of producers, the NPV plotted in the Fig. 3.16 is the highest among the N_e realizations. Note that in this method the average gradient calculation (Eq. 2.36) requires N_e simulation runs (one for each realization), therefore, the number of simulation runs for each iteration is equal to N_e plus the number of trials for the line search. With an initial highest NPV of $\$1.14 \times 10^7$, the EnKF method can increase the NPV to $\$1.54 \times 10^7$ in 70 iterations, the lowest NPV among all the methods. From Figs. 3.16 and 3.17, we can see that the steepest ascent algorithm will be sufficiently efficient if the adjoint gradient is available.

To show the robustness of the optimization algorithms, we have tested the production optimization procedure with different initial guesses for the steepest ascent method and all of them ended up with the same final controls and the same NPV. Fig. 3.18 shows the NPV versus iteration number of different initial guesses for the steepest ascent method. All the cases converge to the same final NPV within 5 iterations, which indicates that steepest ascent method is fairly stable in terms of convergence.

Fig. 3.19 shows the final BHP profile obtained from the steepest ascent

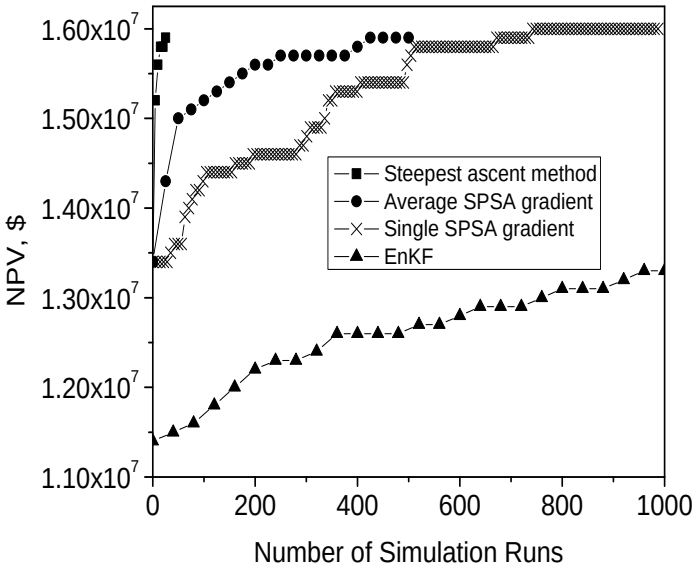


Figure 3.17: NPV as a function of simulation runs for case 1.

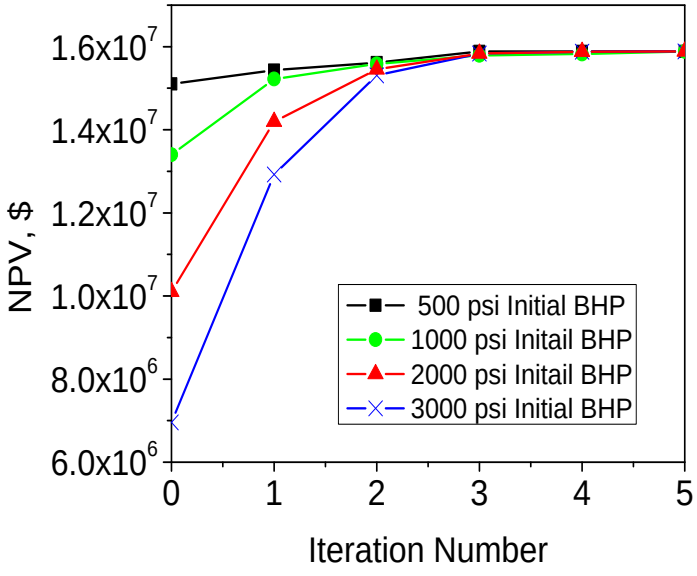


Figure 3.18: Optimization comparison of different initial BHP for steepest ascent method for case 1.

method. The results show a pure "Bang-Bang" solution to this problem. This is not surprising according to the results from Zandvliet et al.(44) as the only constraints on the BHP control are the upper and lower bound constraints. Producers P1 and P4 stay at the lower bound of the BHP specified at early production time and then are equal to the upper bound of the BHP specified at later production time. This is because these two producers are connected to the injector by a high permeability channel. At the early production time, before water breakthrough, BHPs of P1 and P4 stay at their possible lowest value (lower bound of BHP) to produce more oil. The increase in BHP at later production time corresponds to the increase in watercut, as the BHPs increase to the upper bound, the wells are effectively shut-in to stop water production. Producer P2 remains at the lower bound of the BHP specified for the whole reservoir life, since this producer is separated from the injector by a low permeability channel, which acts as a barrier for the water movement. Although there is water breakthrough in Well P3, the BHP remains at its lower BHP bound for the whole reservoir life of 960 days, which may seem unusual. However, a close check on the gradient shows that the elements in the gradient corresponding to the controls at the lower bound are negative and the elements in the gradient corresponding to the controls at the upper bound is positive or zero, so the final control is a local maxima. Note that the well is shut-in automatically whenever the BHP is higher than the gridblock pressure in which it resides, and when the BHP reaches its upper bound, we shut-in the well even if this upper bound is still lower than the gridblock pressure. From a careful inspection of the results, we notice that at the first iteration using the steepest ascent method, all the components of the gradient are negative, except the ones corresponding to the last control step (between day 840 and day 960) of P1 and P4, which have water breakthrough time much earlier than 840 days, are positive. Early iterations will drive these controls of the last control step to the upper bound and all the others to the lower bound. Once the the controls of the last control step get to the upper bound, the ones next to them will be driven to the upper bound. This continues until a local maxima is found. This explains why the steepest ascent method with different initial guesses gave the same NPV at

convergence: all the cases with different initial guesses have the same controls after the first iteration. The above observation seems to coincide with the switching time optimization(44), i.e. optimizing on the time to switch from the lower bound ("on" status in valve setting) to the upper bound ("off" status in valve setting) or vice versa. For this example, we have one switching time per producer and the switching time starts from the end of the reservoir life and then moves backward in time during optimization.

The final BHP controls obtained from the algorithm using an average of ten stochatic gradients of SPSA are shown in Fig. 3.20, which are the same as the ones from the steepest ascent method in Fig. 3.19. This shows that SPSA with an average gradient might be promising for production optimization in the case that the true gradient cannot be readily calculated. We need to note that there are only 32 controls in this production optimization example. The promising results may be due to the fact the problem is so small that the average of ten stochastic gradient of SPSA is sufficiently close to the true gradient. Fig. 3.21 shows the final BHP obtained from a single SPSA gradient. Although the NPV increased to a value close to that from the steepest ascent method as shown in Fig. 3.16, the final BHP does not seem realistic with the nonsmooth behavior for well P2. The "bumpy" behavior is mainly because with a single SPSA gradient, SPSA is similar to a random search direction. Other wells show earlier well shut-in (BHP controls at the upper bound) than that from the steepest ascent method. Fig. 3.22 shows the BHP with the highest NPV from the N_e realizations of the EnKF method. Note that in the final control plots of Figs. 3.19 to 3.22, we plot the BHP at its upper bound as long as the well is shut-in. The actual BHP from the production optimization might be lower than the upper bound since the upper bound of BHP is much higher than the average reservoir pressure.

3.2 Case 2

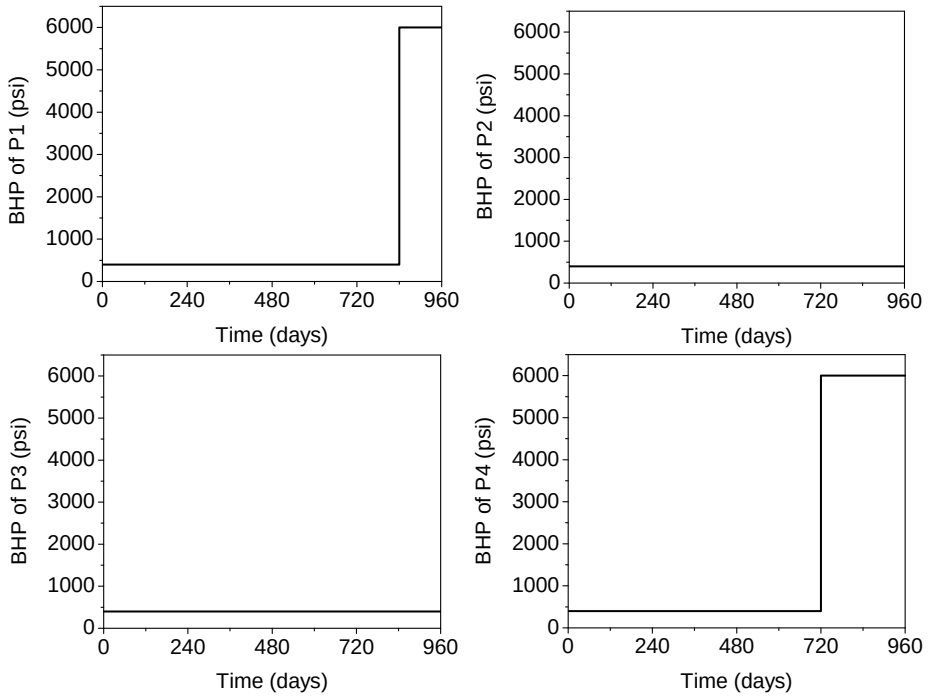


Figure 3.19: Final BHP controls from steepest ascent method for case 1.

3.2.1 Reservoir Description

Similar to case 1, case 2 is also a 2-D reservoir with no flow boundaries. Only water and oil are present in the reservoir. The reservoir consists of $25 \times 25 \times 1$ grid system. There is only one injector (INJ) located at (13,13) and four producers located at P1 (1,1), P2 (25,1), P3 (1,25) and P4 (25,25). The numbers in the parenthesis are the well grid block indices. Here the well locations are fixed and they are not subject to optimization. The porosity of the reservoir is assumed to be homogeneous with a constant value of 0.25 while permeability is heterogeneous with an average value of 55 md. The permeability field is generated using the sequential Gaussian simulation (SGS) assuming a log-normal distribution. The variogram parameters for permeability distribution are shown in Table 3.3. The permeability distribution together with the well locations is shown in Fig. 3.23. Noted that Fig. 3.23 shows the actual permeability distribution, not the log-permeability.

The reservoir properties are listed in Table 3.4. The anticipated total project life is 10 years and the control time step size is set to 6 months, so there are 80 control variables ($N_u = 80$) in the control vector with 20 for each producer. All four producers are under BHP control with upper bound 4000 psi and lower bound 200

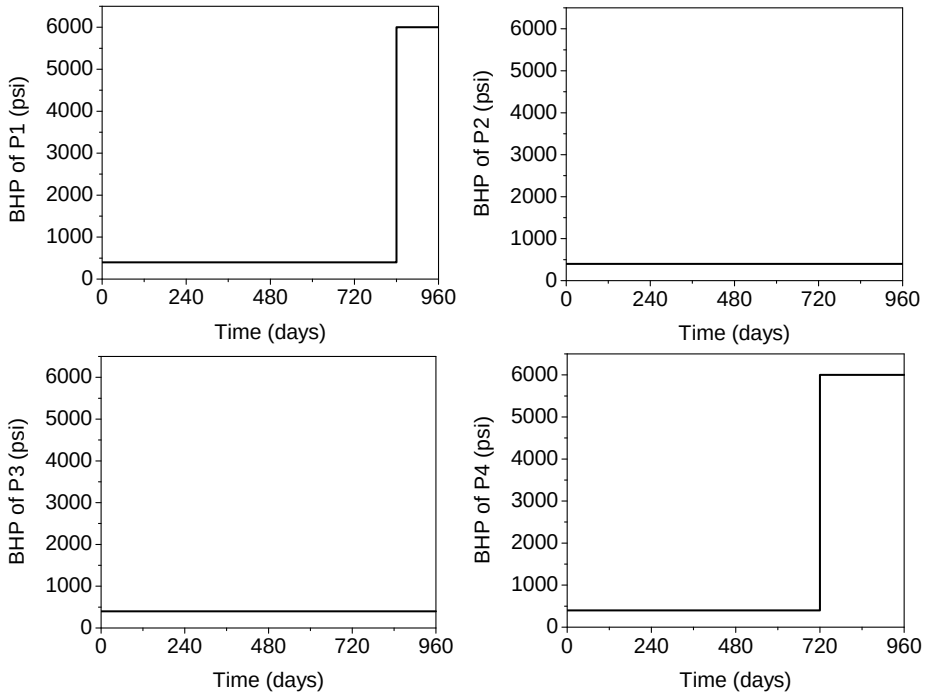


Figure 3.20: Final BHP controls from average SPSA gradient method for case 1.

Table 3.3: Variogram of the permeability distribution of case 2

$(\ln k)_{mean}$	4.0
σ	0.7
α	45
r_1 (ft)	3500.0
r_2 (ft)	500.0

psi. The injection rate is set to a constant of 5000 STB/day. The water and oil relative permeability curve used in the reservoir model is shown in Fig. 3.2.1.

3.2.2 Initial Bottom Hole Pressure

Use the same method as in case 1, we also generate $N_e = 40$ realizations of the initial BHP. Figs 3.25 to 3.28 show the histogram of mean for the realizations of each producer.

Figs 3.29 to 3.32 show the histogram of initial wellbore pressure controls for all four producers.

To show how the initial BHP varies with time, we select the first ten realizations of BHP for each producer P1, P2, P3 and P4 from the N_e realizations and display them in Figs. 3.33 to 3.36.

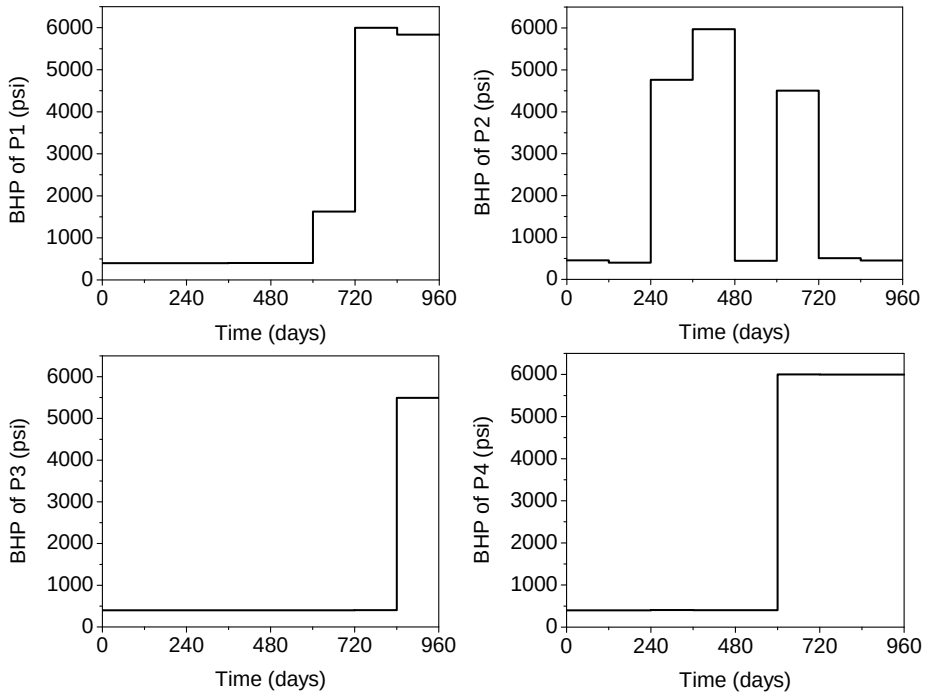


Figure 3.21: Final BHP controls from a single SPSA gradient method for case 1.

3.2.3 Comparison of Optimization Algorithms

Fig. 3.37 shows the net present value as a function of the iteration number for different optimization algorithms. The curve with crosses shows the performance from EnKF. The one plotted in the figure for each iteration is the highest NPV obtained from the N_e realizations. With an initial NPV of $\$1.5 \times 10^8$, the EnKF method is able to increase the NPV to $\$1.64 \times 10^8$ in about 25 iterations. The performance from the steepest ascent algorithm is shown in the curve with squares. The initial BHP is one of the initial realizations from EnKF. At initial conditions for BHPs, net present value is about $\$1.46 \times 10^8$, with the steepest ascent method, the value obtained increases to about $\$1.66 \times 10^8$ in about 5 iterations, slightly larger than that from EnKF. The same initial BHP is used for the SPSA optimization method as in the steepest ascent method. Both SPSA optimization cases give slower convergence compared to EnKF and steepest ascent methods, although the one using the average gradient shows higher NPV at the end of 50 iterations. Fig. 3.37 suggests that the steepest ascent algorithm can achieve the highest NPV in fewest iterations compared to other optimization algorithms.

To investigate the robustness of the steepest ascent method, we start with

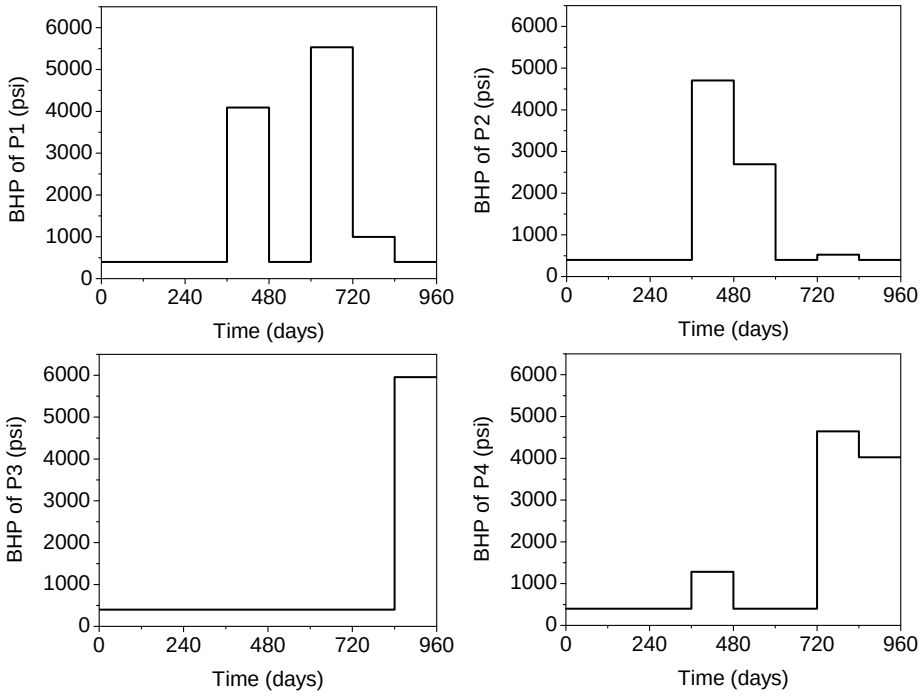


Figure 3.22: Final BHP controls from EnKF for case 1.

different initial guesses to do production optimization. Fig. 3.38 shows the performance of the steepest ascent algorithm with different initial guesses for the BHP. All the cases converge to almost the same final NPV, which indicates that the algorithm is fairly stable in terms of convergence.

Figs. 3.39 shows the optimal BHP profile from the steepest ascent method. As in case 1, the final control variables have the "bang-bang" behavior. Before water breaks through into a producer, it will produce the most oil when the BHP is at its lowest value possible. After water breakthrough, we need to increase the BHP as the water cut increases. From the results of steepest ascent for P1, the BHP is always at its lower bound (upper left panel of Fig. 3.39), since there is low permeability barrier between the injector and this producer (Fig. 3.23) and no water has broken through into this well. The optimal controls for other wells (P2, P3 and P4) reaches the lower bound at early time and upper bound at the late time, because all these wells have water breakthrough at late time. BHP at the upper bound tries to shut-in these wells to lower the cost caused by high water production. At what point the well should be shut-in depends on the water-cut and the contrast between the oil price and the water disposal cost.

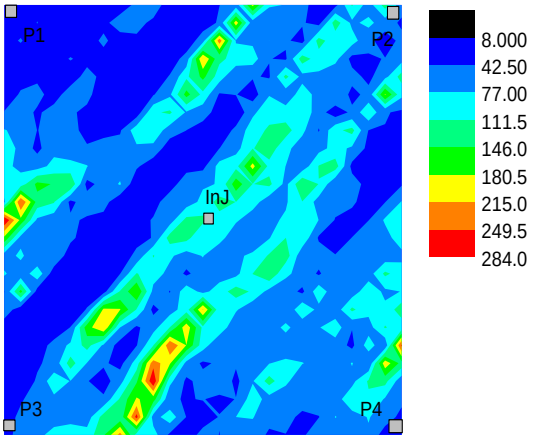


Figure 3.23: Permeability distribution for case 2.

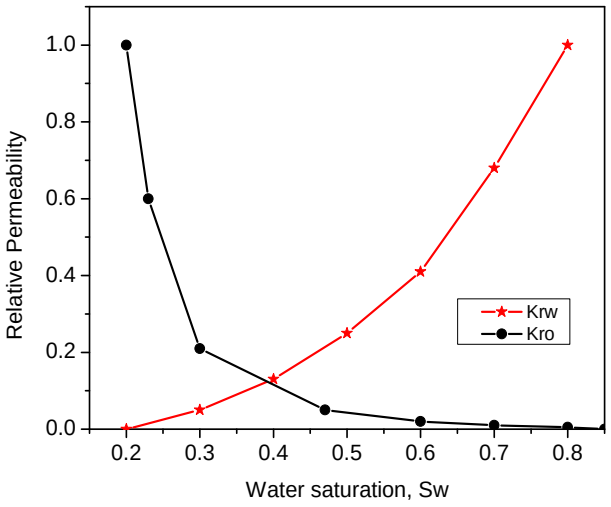


Figure 3.24: Water and oil relative permeability curve for case 2.

The optimal BHP profiles from SPSA are shown respectively in Fig. 3.40 and Fig. 3.41. The final controls from two SPSA methods look "bumpy" although the one with average gradient behaves better than that with a single stochastic gradient. The "bumpy" behavior is mainly because that SPSA is somewhat similar to a random search direction. From Figs. 3.42, we can see that EnKF shows somewhat similar solutions to the steepest ascent method except that the BHP of P3 and P4 did not increase to the upper bound at the later time. For P1, final BHP keeps very close

Table 3.4: Reservoir properties for case 2

Grid block size	118 ft
Thickness	50 ft
ρ_{osc}	56 lb/ft^3
ρ_{wsc}	62.4 lb/ft^3
μ_o	2.4 cp
μ_w	0.96 cp
B_o	0.972
B_w	1.0034 rb/stb
Rock Compressibility	3×10^{-6} psi^{-1}
Porosity	0.25
Top Depth	10000 ft
Residual oil saturation	0.15
Irreducible water saturation	0.2
Initial water saturation	0.2
Initial reservoir pressure	4500 psi
Total production period	10 years
The control step	6 months

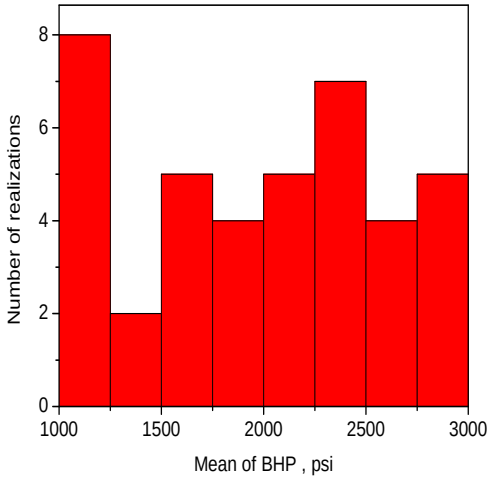


Figure 3.25: P1 initial distribution of mean of BHP for case 2.

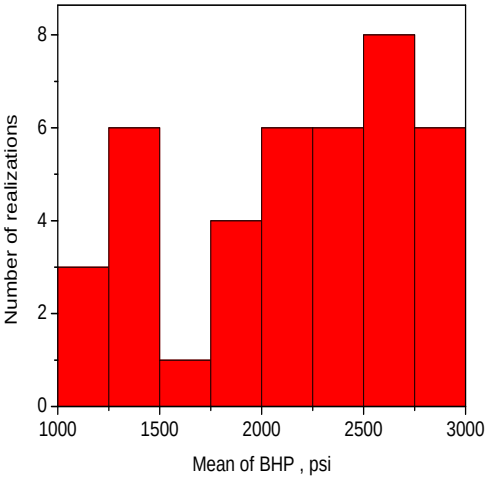


Figure 3.26: P2 initial distribution of mean of BHP for case 2.

to lower bound during the whole production time. Optimal BHP of P2 is close to the lower bound at the early production time and increases to upper bound at the later production time.

3.3 Summary

For two synthetic cases, we compared three optimization algorithms (steepest

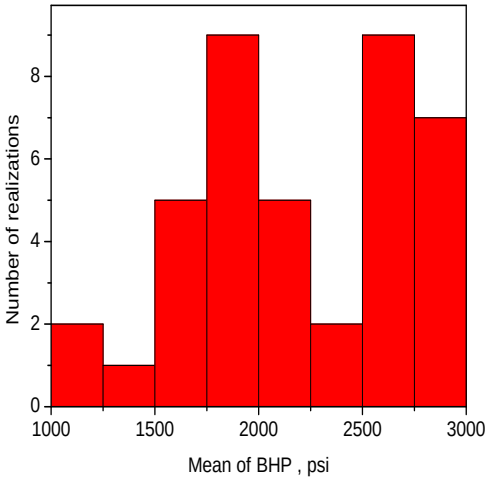


Figure 3.27: P3 initial distribution of mean of BHP for case 2.

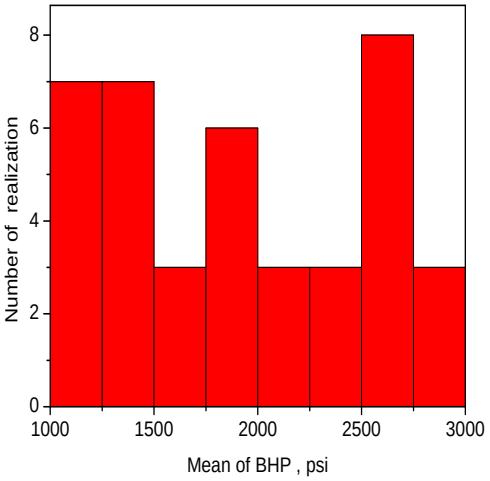


Figure 3.28: P4 initial distribution of mean of BHP for case 2.

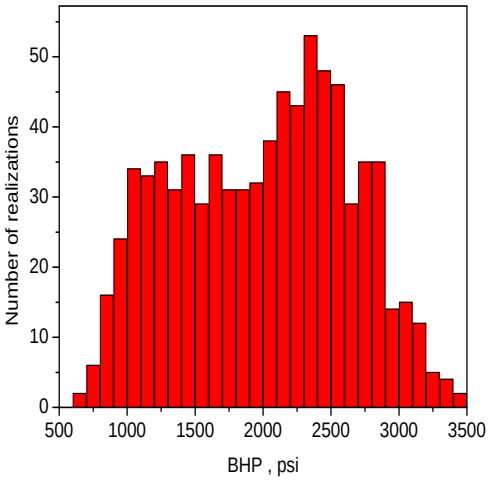


Figure 3.29: Histogram for BHP initial realizations of P1 for case 2.

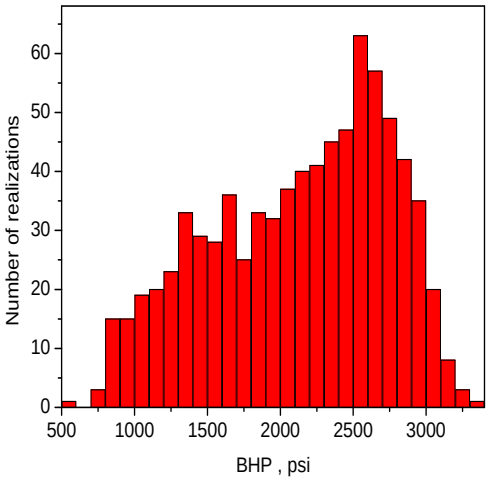


Figure 3.30: Histogram for BHP initial realizations of P2 for case 2.

ascent method, SPSA and EnKF) based on true geology. In comparison with other optimization algorithms, steepest ascent method always gives the best optimization results: highest NPV and smooth BHP profile. The robustness of the steepest ascent method is tested and it proves that steepest ascent method is very stable. Therefore, steepest ascent method will be sufficiently efficient if the adjoint gradient is available. As for SPSA, although SPSA with average gradient works better than SPSA with a single stochastic gradient and gives very satisfactory results in case 1, it seems that it does not work well for case 2 and a rough BHP profile is obtained

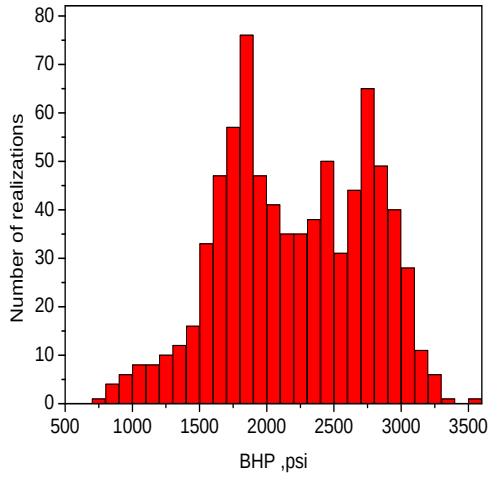


Figure 3.31: Histogram for BHP initial realizations of P3 for case 2.

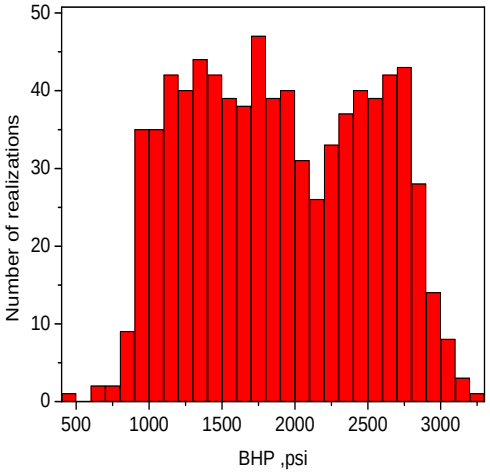


Figure 3.32: Histogram for BHP initial realizations of P4 for case 2.

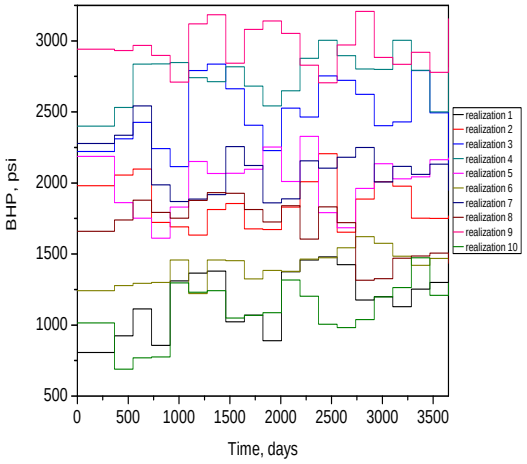


Figure 3.33: Ten realizations of BHP profile for P1 for case 2.

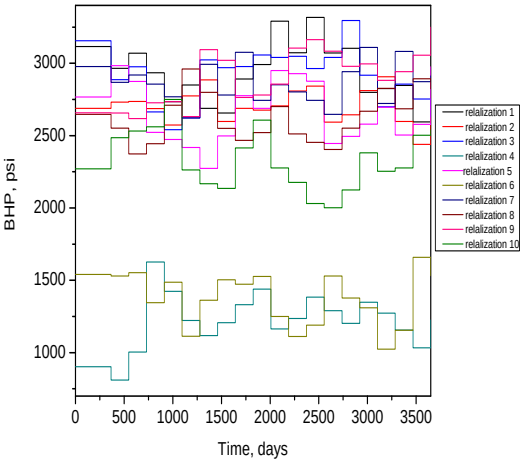


Figure 3.34: Ten realizations of BHP profile for P2 for case 2.

because of its random direction. The convergence speed of both SPSA methods ia much slower than that of steepest ascent method. For EnKF, the average gradient calculation requires N_e simulation runs (one for each realization), therefore, the number of simulation runs for each iteration is equal to N_e plus the number of trials for the line search. It is probably too computationally inefficient for real application. Also in two cases, EnKF does not always give the good results and its convergence is very slow. So later we will focus on using the steepest ascent method for production optimization assuming we will implement the adjoint gradient in the near future.

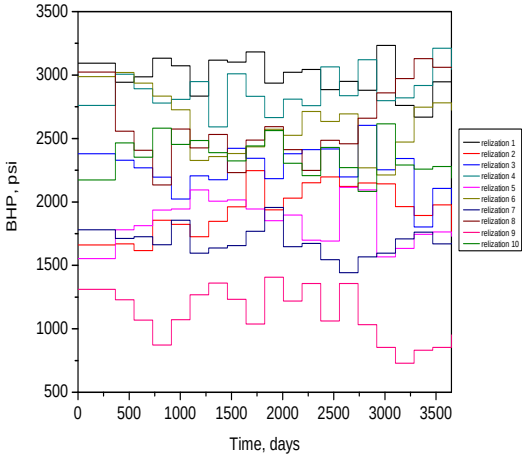


Figure 3.35: Ten realizations of BHP profile for P3 for case 2.

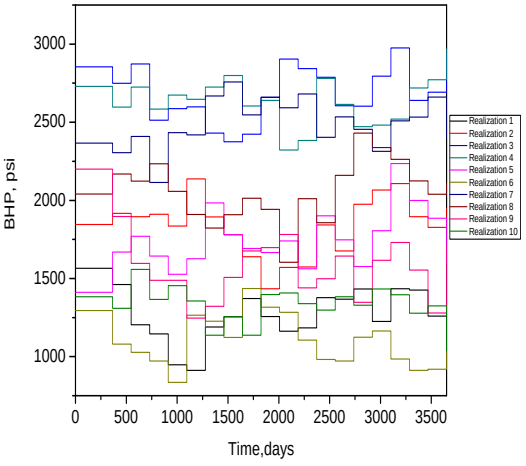


Figure 3.36: Ten realizations of BHP profile for P4 for case 2.

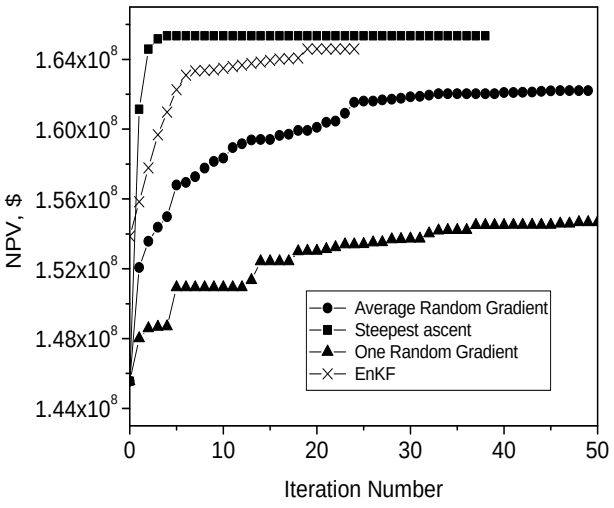


Figure 3.37: Net present value as a function of the iteration number for case 2.

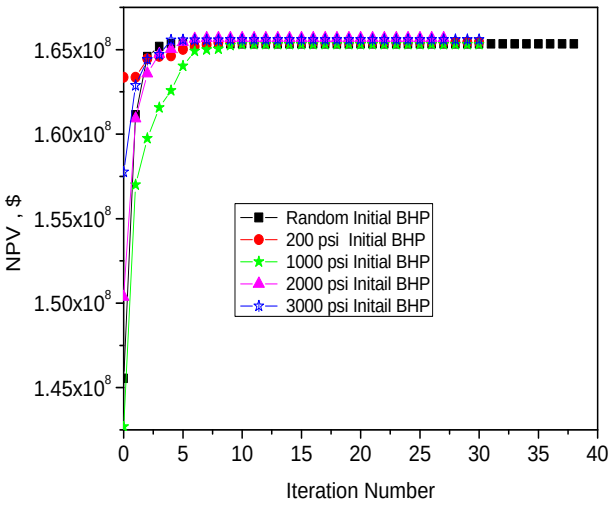


Figure 3.38: Optimization comparison of different initial BHP for steepest ascent method for case 2.

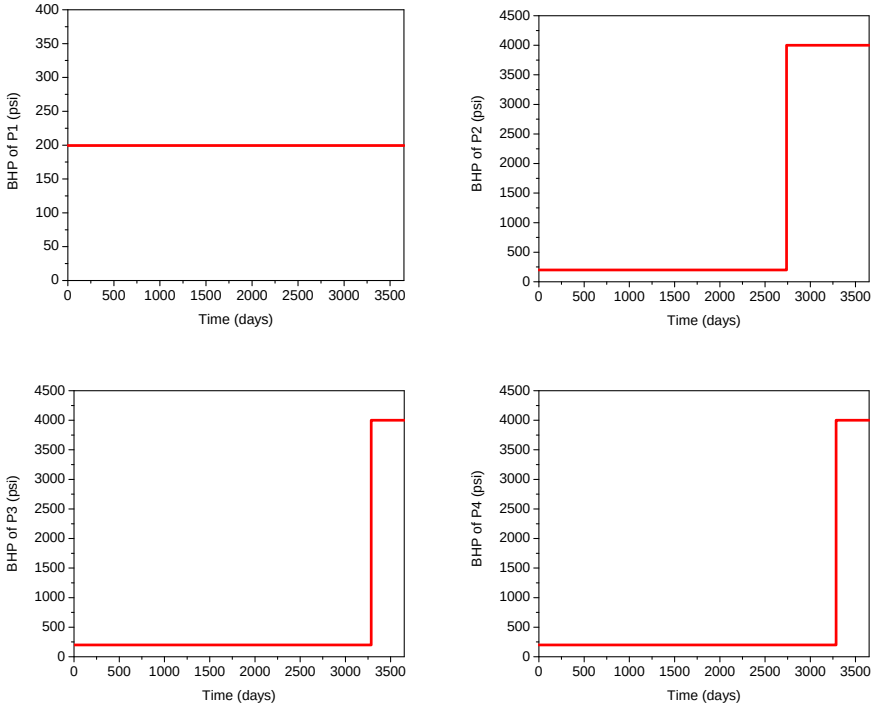


Figure 3.39: Final BHP controls from steepest ascent method for case 2.

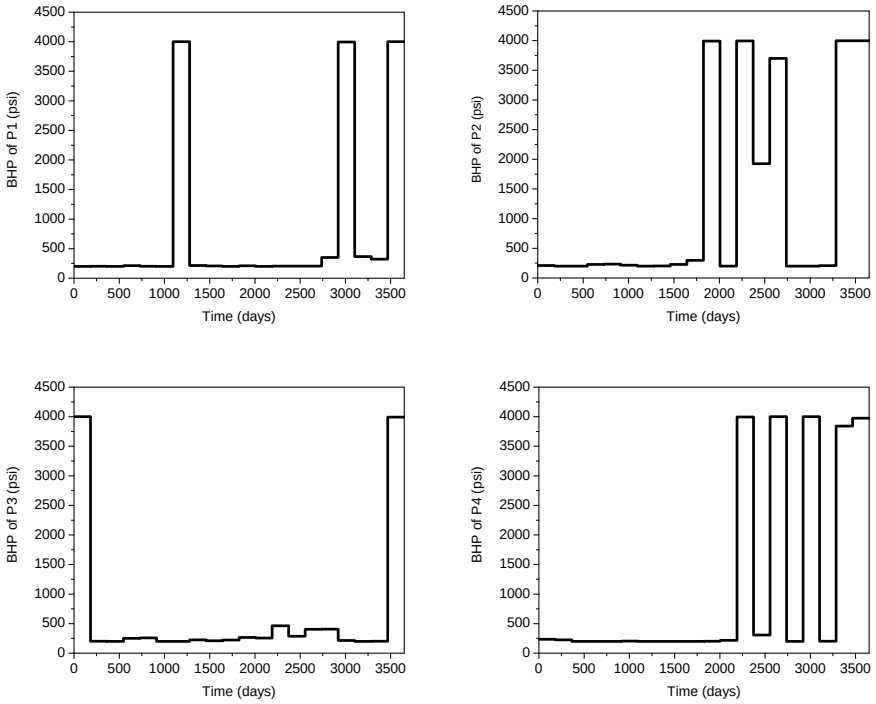


Figure 3.40: Final BHP controls from average SPSA gradient method for case 2.

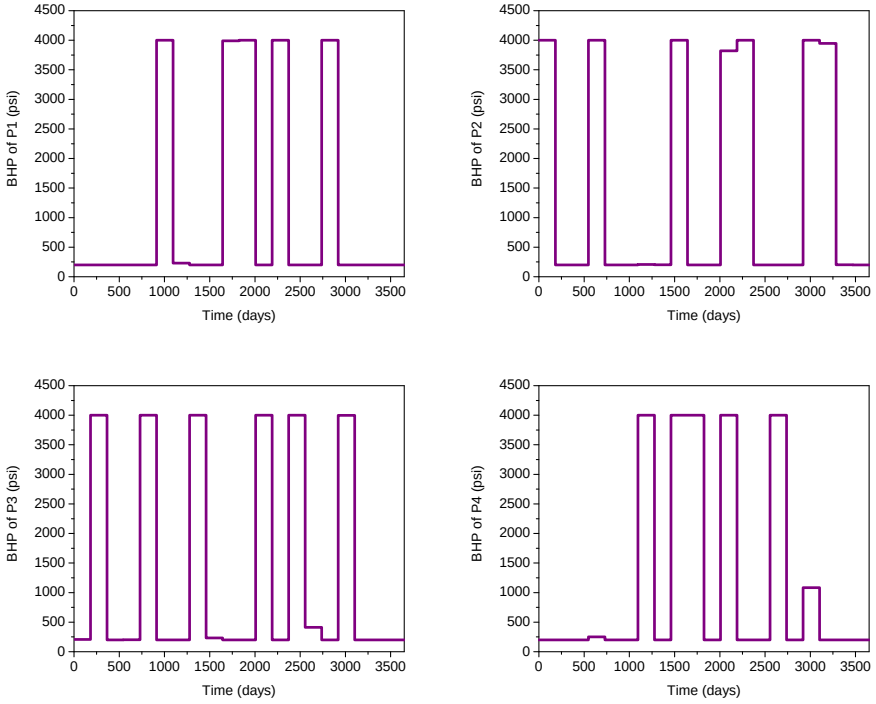


Figure 3.41: Final BHP controls from a single SPSA gradient method for case 2.

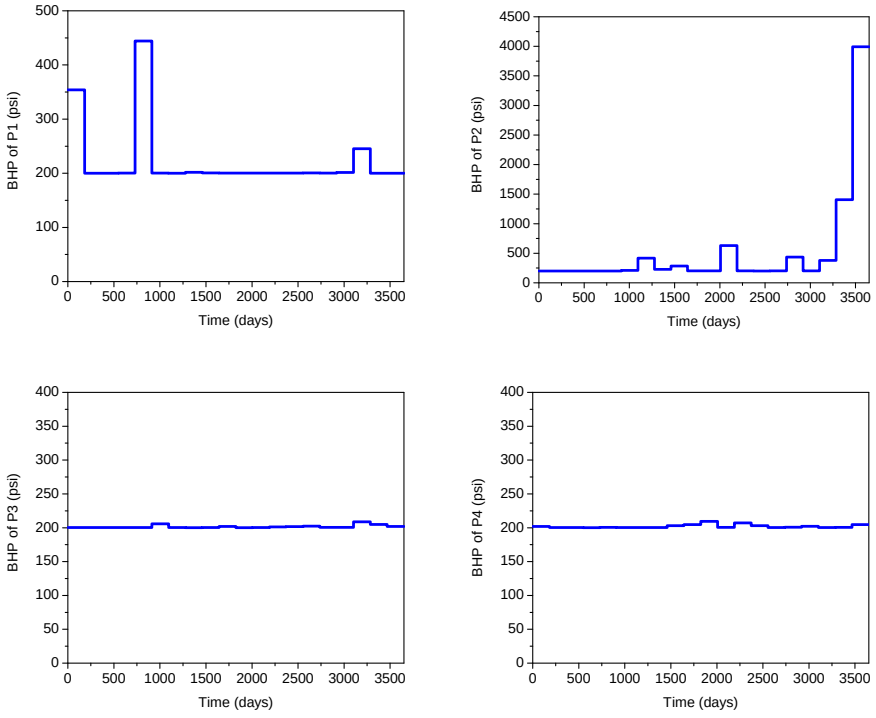


Figure 3.42: Final BHP controls from EnKF method for case 2.

CHAPTER 4

PRODUCTION OPTIMIZATION WITH UNCERTAINTY

In the previous chapter, we considered production optimization with known geological fields: porosity and permeability distributions, and compared three different optimization algorithms (steepest ascent method, SPSA and EnKF). In reality, we do not know the true geology and production optimization must be done based on uncertain geology. In this chapter, we mainly deal with production optimization with geological uncertainty, "closed-loop" reservoir management (CLMR). CLMR involves two steps: production optimization with geological uncertainty and data assimilation (history-matching) to reduce the geological uncertainty as production data become available. This is an iterative process with production optimization and data assimilation alternating through the whole life time of the reservoir. For the data assimilation (history matching) part, we use the ensemble Kalman filter (EnKF) to reduce geological uncertainty by integrating dynamic production data. As for production optimization, steepest ascent method is employed to obtain the optimal control variables.

4.1 Result Analysis

Since adjoint code for gradient calculation is not available, we use the perturbation method to generate gradient for production optimization. If we apply the "closed-loop" reservoir management into large reservoir, this gradient generation will be very time-consuming. Therefore, we only do production optimization with unknown geological fields in the first case of Chapter 3. With Sequential Gaussian Cosimulation, we have generated 90 ensemble members of the porosity and log-permeability fields from the prior geological information for data assimilation with EnKF. The production optimization is done on the central model, which is the

updated model obtained by assimilating measurements without perturbation using the prior mean as its initial realization. For the linear case, the central model is equivalent to the MAP estimate(43). The basic procedure is the following:

1. Optimize the control variables based on the prior mean model (central model) for the whole reservoir life.

2. Generate true data using the final optimum control obtained at the previous data assimilation to the time when there are measurement data. Note that we have measurements every 30 days and the measurement data include oil and water production rates from producer and the BHP from the injector. The synthetic data are generated by adding noise to the true data. The standard deviation for the measured oil and and water rates are 5 STB/day and for the measured BHP of the injector is 10 psi.

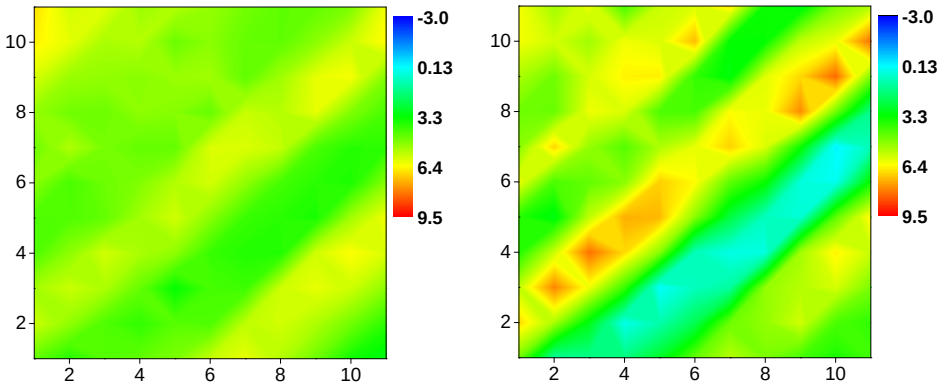
3. Assimilate data with EnKF up to a point t_n that production optimization is requested, which occurs every 120 days.

4. Optimize the control variables based on the updated central model from t_n to the end of the reservoir life.

5. Repeat step 2, 3 and 4 until the end of the reservoir life.

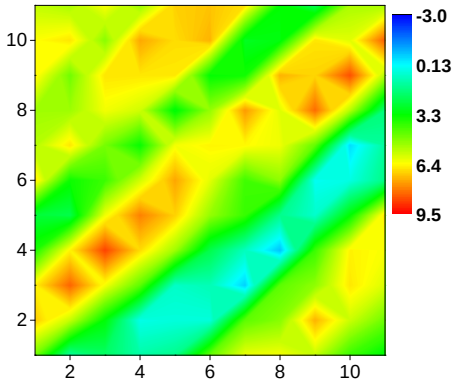
Fig. 4.1 shows the evolution of the average horizontal log-permeability after data assimilation at 60, 120 and 240 and 480. After data assimilation up to day 60, we start seeing the long high permeability channel connecting wells P1 and P3. After day 120, the short high permeability channel between the injector and the producer P3 becomes evident. After day 240, it seems that both high permeability channels get wider than in the truth (Fig. 3.1).

Figs. 4.2 and 4.3 show the average log permeability and average porosity distribution after data assimilation at 960 days. Comparing Fig. 4.2 to the true permeability distribution in Fig. 3.1, we see that the average log-permeability distribution after data assimilation with EnKF captures the main geological features, especially

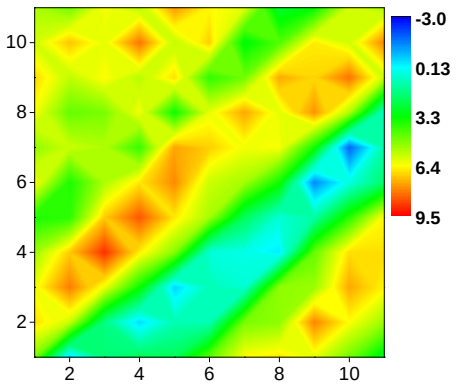


(a) 60 days

(b) 120 days



(c) 240 days



(d) 480 days

Figure 4.1: Evolution of the average log-horizontal permeability during data assimilation.

the long high permeability channel connecting the injector with the two producers (P1 and P4), although the channel from EnKF is a little wider than the true. The short high permeability channel close to producer P3 is much wider than the truth. The high permeability area around producer P2 is more or less shown in the average log permeability distribution, which is similar to the truth case. However, the low permeability channel between the injector and the producer P2 is shifted towards the injector. The average porosity distribution in Fig. 4.3 after data assimilation at 960 days roughly captures the true geological features, but the estimate is poorer than that of the permeability distribution.

Fig. 4.4 shows the ensemble predictions of the oil production rate during data assimilation compared to the truth. In all the similar figures, red curves represent the truth case, blue curves are the central model and grey curves are the ensemble predictions from each step of data assimilation. The central model refers to the

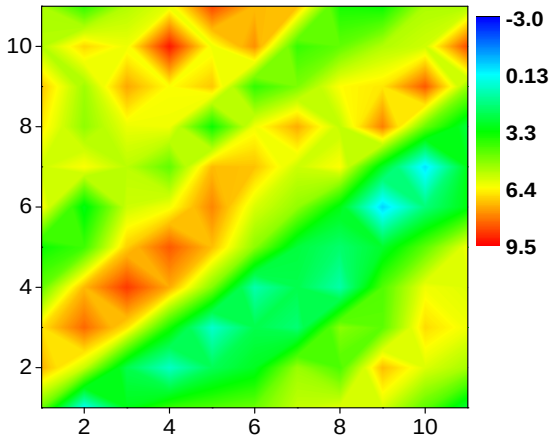


Figure 4.2: Average horizontal log-permeability distribution after data assimilation to 960 days.

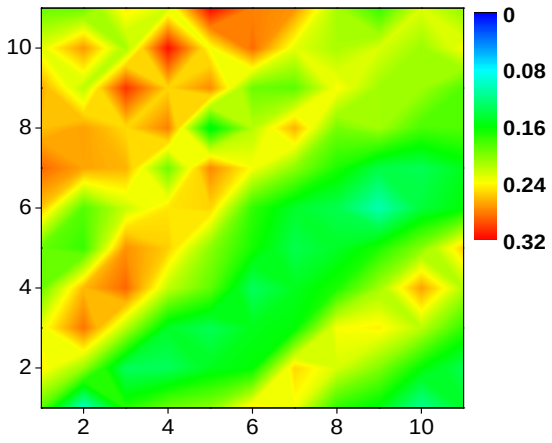


Figure 4.3: Average porosity distribution after data assimilation to 960 days.

ensemble member in the EnKF data assimilation generated by starting with prior mean as the initial model and using the observed data without noise in the assimilation step. It can be seen that the ensemble predictions give large uncertainty during early time data assimilation and the uncertainty band becomes smaller as more data are assimilated. During the course of data assimilation, the truth case is always within the uncertainty band, so the results do not appear to be biased. Fig. 4.5 shows the ensemble predictions of the water production rate during data assimilation compared to the truth. Similar behavior to the oil production rate is observed. Fig. 4.6(a) show the ensemble prediction of the bottomhole pressure compared to the truth. As in the water and oil production rate predictions, there is a large uncertainty at early times and some of the ensemble members even reach the maximum bottomhole pressure specified (10000 psi). After about 240 days, the uncertainty band of the ensemble predictions is so narrow that we see the truth essentially coincides with the ensemble predictions. The average reservoir pressure in Fig. 4.6(b) shows the typical "saw tooth" behavior of sequential data assimilation: the uncertainty band increases during prediction and then gets reduced after data assimilation at late times. Compared to the ensemble predictions of the oil and water production rates of individual well and the BHP of the injections wells, the average pressure pressure shows a slightly bigger uncertainty. Figs. 4.6(c) and

(d) present the cumulative oil production and cumulative water production for the ensemble obtained during data assimilation compared to the truth. The uncertainty band for the cumulative oil production (Fig. 4.6(c)) is much larger than that of the cumulative water production (Fig. 4.6(d)), but the truth case is within the uncertainty band for both results. The large uncertainty band for the cumulative oil production is arises from the fact that the uncertainty in the oil rate (Fig. 4.4) is large than the uncertainty in the water rate (Fig. 4.5) at times prior to 240 days.

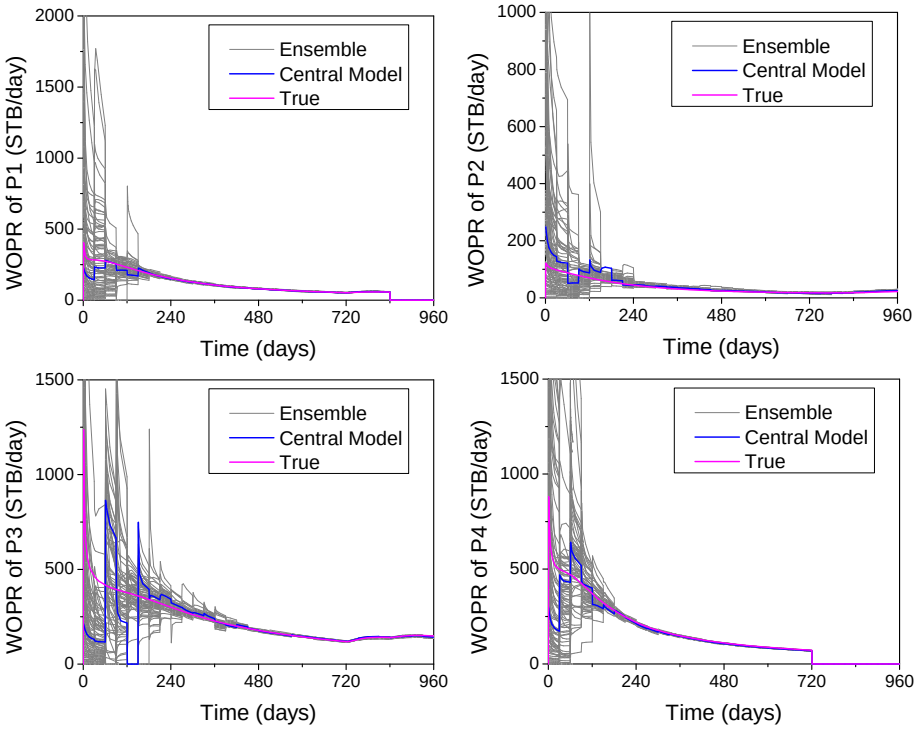


Figure 4.4: Ensemble oil production rate compared to the truth during data assimilation from t=0 to t=960 days.

The final control variables from production optimization based on the central model is exactly the same as the one we obtained based on the true geology of Fig. 3.19. Although not shown here, the updated permeability and porosity distribution for the central model after 960 days is very much like the average permeability and porosity distribution shown in Figs. 4.2 and 4.3, because all the models are close to each other after data assimilation. It should be noted that every time when we do production optimization, we use the same initial guess (500 psi) instead of the final control from last step of production optimization. The reason for doing this is because once the control goes higher than the grid block pressure at the first step of

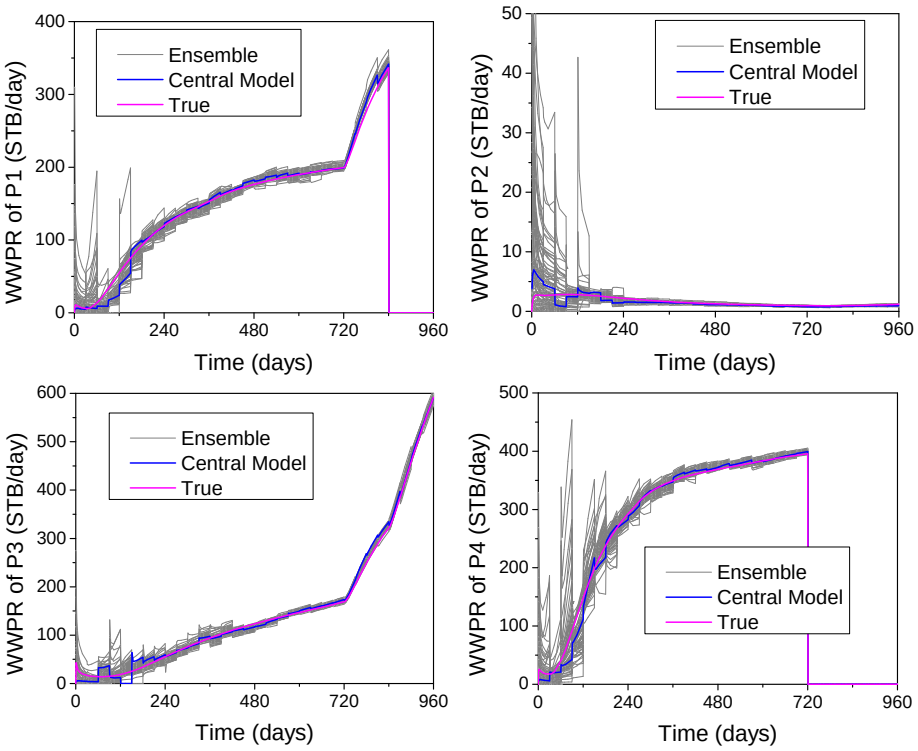
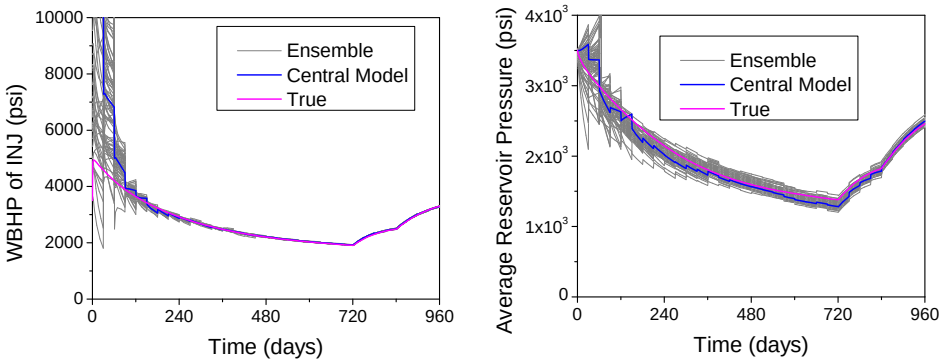


Figure 4.5: Ensemble water production rate compared to the truth during data assimilation from t=0 to t=960 days.

production optimization, the producer will be shut-in due to the fact that the BHP is higher than the grid block pressure. In this case, NPV is never sensitive to the well control and cannot be adjusted during the optimization.

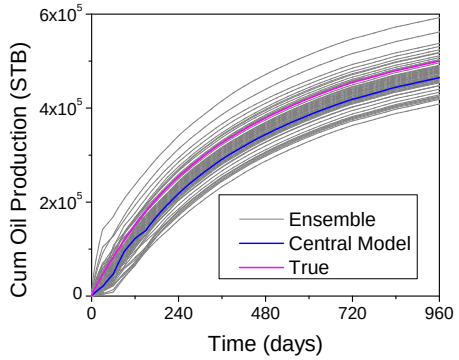
4.2 Nonlinearity

For the closed-loop reservoir management procedure, we also tried to use different initial guesses for the BHP during production optimization to test its stability. Other than 500 psi, we have tried to use an initial BHP of 400 psi which is the lower bound, 1000 psi, 2000 psi and 3000 psi. Surprisingly, different final controls are obtained with different initial guesses. With the initial guess of 400 psi and 500 psi, we obtained the same results as knowing the true geology. However, with an initial guess of 1000 psi, 2000 psi and 3000 psi, we obtained slightly different final controls. Fig. 4.8 shows the final controls when the initial BHP is 1000 psi with geological uncertainty. Compared to the final control obtained with an initial guess of 500 psi, which is the same as that with true geology shown in Fig. 3.19, the only difference is that the well P1 was shut-in one control step earlier when the initial guess for BHP

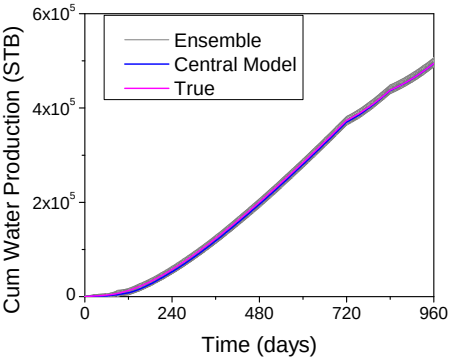


(a) WBHP INJ

(b) FPR



(c) FOPT



(d) FWPT

Figure 4.6: Ensemble prediction compared to the truth during data assimilation from t=0 to t=960 days.

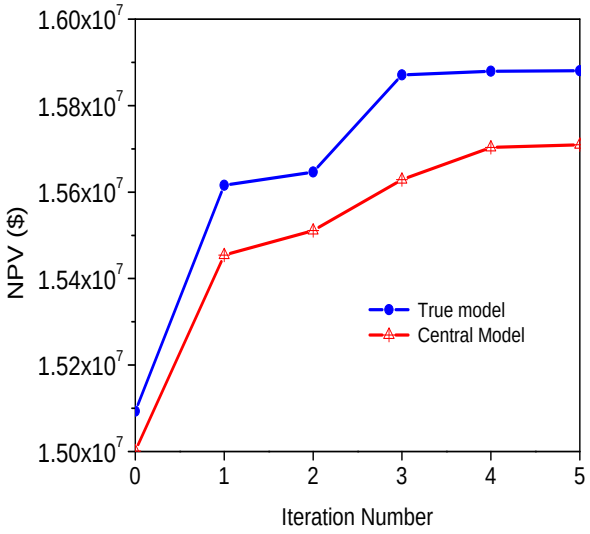


Figure 4.7: Comparison of NPV with iteration number for true geology and updated central model.

is 1000 psi. All the other wells have the same final control in these two cases.

As mentioned earlier, the control shown in Fig. 3.19 is at least a local maxima because all the controls at the lower bound have a negative component in the gradient and all the controls at the upper bound have either zero (well shut-in) or a positive component in the gradient. When we use the final controls from the case with initial BHP 1000 psi and run the simulator with the true geology, we find that the NPV is even higher that the maximum obtained with controls shown in Fig. 3.19. This confirms that the maxima obtained from the steepest ascent method with the true geology is only a local maxima. Since there is only one control that is different between these two cases: one is at its lower bound and one is at its upper bound, we plot the NPV as a function of that control from P1 between the lower bound and upper bound with true geology. The result is shown in Fig. 4.9. Note that the NPV is a nonlinear function of the control variable. When the control is at its lower bound, it has a negative derivative, so it tends to increase NPV by lowering its BHP. As the BHP increase, this derivative decreases and reaches zero at about 1350 psi and then become positive as BHP further increases. The well is shut-in when the BHP reach about 3000 psi, so the NPV becomes flat. A check on the gradients of the control indicates that both solutions are local maxima, but setting this control to the highest value (shut-in) gives the highest NPV as indicated by the results of Fig. 4.9.

As mentioned earlier, there is one switching time per producer and during optimization, the switching time (defined as the time the BHP control switches from lower bound to upper bound) moves backward as a function of iteration for the problem considered in the paper. Here we explore the behavior of the NPV versus the switching time.

Fig. 4.10 shows the NPV versus switching time for all the producers in the vicinity of the final controls obtained with known true geology, i.e. change the switching time of one producer while keeping all the controls of other wells as in Fig. 3.19. The figures show that the highest NPV is obtained at the switching time shown in Fig. 3.19 for P2 (day 960), P3 (day 960) and P4 (day 720), but not

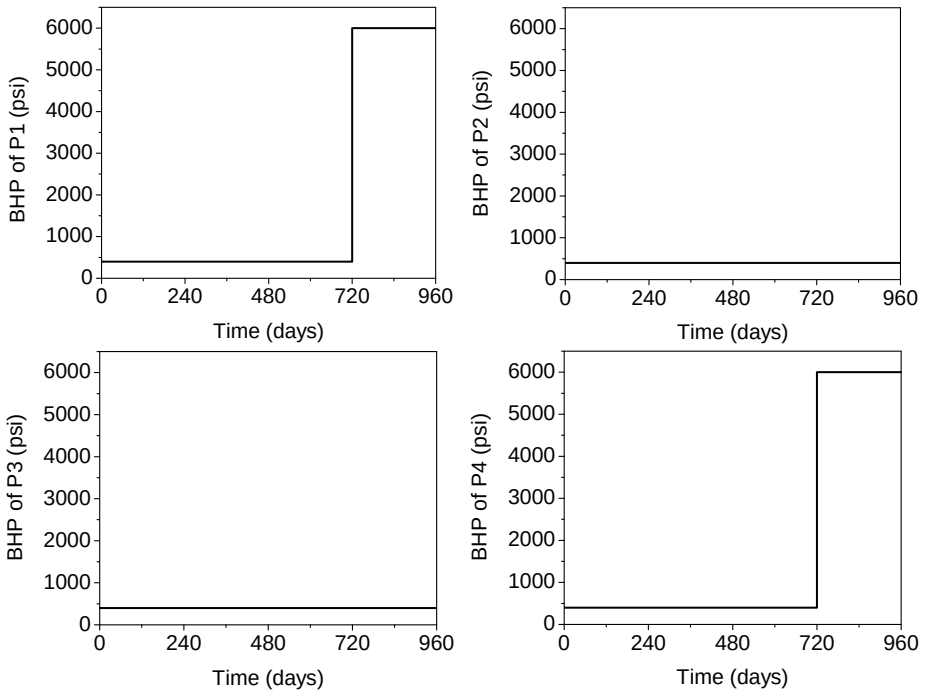


Figure 4.8: Final controls for well P1 with initial BHP= 1000 psi for closed-loop reservoir management.

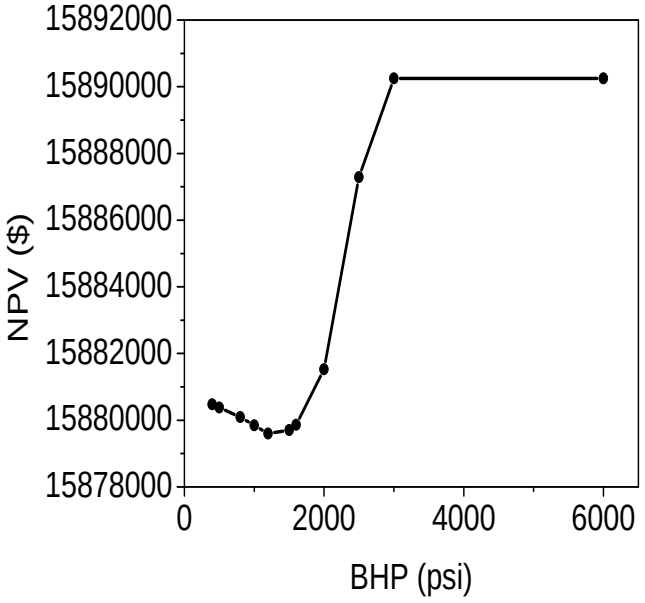


Figure 4.9: NVP as a function of one BHP control for P1.

for P1. Note that the switching time at day 960 refers to keeping the BHP at its lower bound for the whole reservoir life. The highest for P1 is obtained at day 720 instead of day 840 as shown in Fig. 3.19. The final controls in Fig. 3.19 are at a local maxima according to the earlier discussion as we optimize based on the BHP

controls. However, Fig. 4.10 shows that if we optimize based on the switching time, the solution in Fig. 3.19 will not even be a local maxima. Fig. 4.11 shows the NPV versus the switching time for all the producers in the vicinity of the final controls obtained from the closed-loop reservoir management (Fig. 4.8). The highest NPV is obtained at the switching time shown in Fig. 4.8 for P1 (day 720), P2 (day 960) and P4 (day 720), but not for P3. The highest for P3 is obtained at day 840 instead of day 960 as shown in Fig. 4.8. Fig. 4.11 also shows that the final controls in Fig. 4.8 can only be a local maxima when we optimize based on BHP and they will not be a local maxima if we optimize on the switching time.

Figs. 4.10 and 4.11 show that the NPV versus the switching time has a concave up shape, while the NPV versus a BHP control shown in Fig. 4.9 has a concave down shape with maxima at the upper and lower bounds. This may be an indication that optimization based the switching time might be easier than that using the BHP controls if the assumption that there is one switching time per well is true.

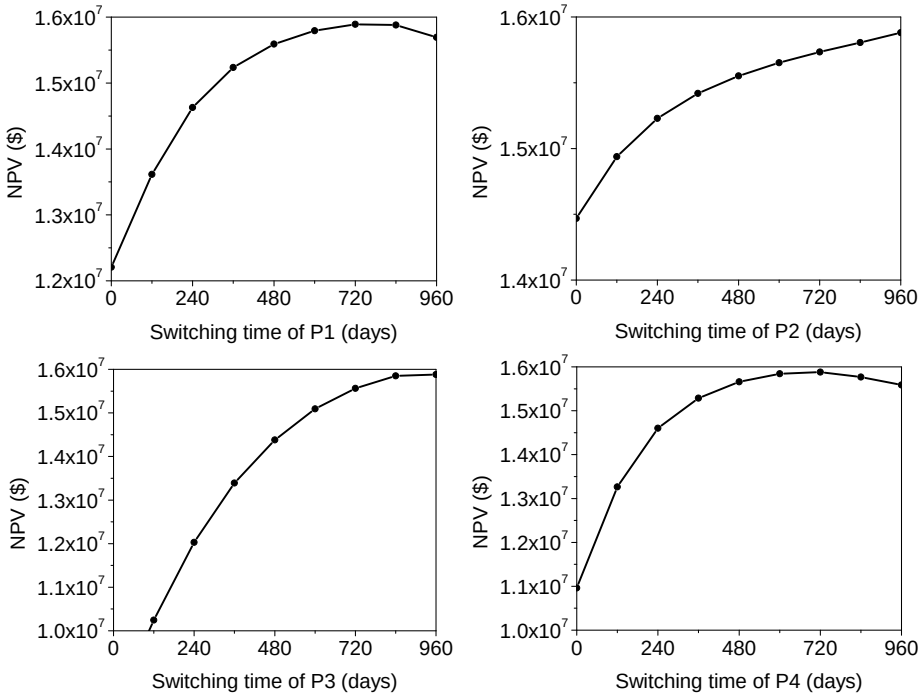


Figure 4.10: NPV versus switching time in the vicinity of final controls with true geology.

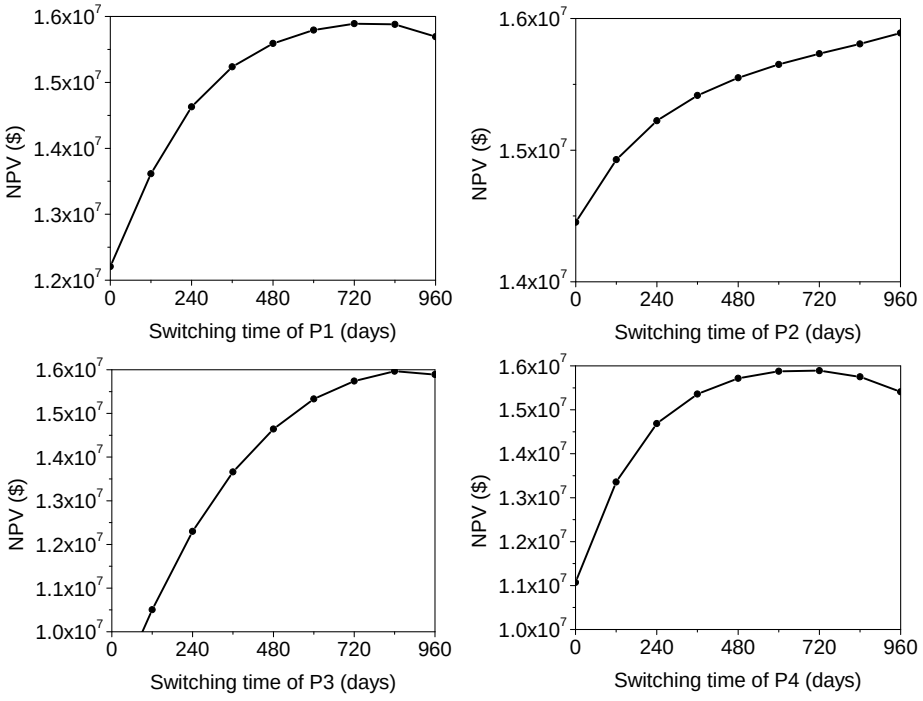


Figure 4.11: NPV versus switching time in the vicinity of final controls of closed-loop reservoir management.

CHAPTER 5

WELL PLACEMENT OPTIMIZATION

In the optimization process, well locations in the reservoir are fixed and known and are not subject to optimization. However, in the process of reservoir development, the first key step is to determine the location of wells we want to drill. We always wish to drill wells at optimal locations so that more hydrocarbons can be extracted at a lower cost. Because well locations in a reservoir simulator are commonly treated as discrete variables, standard implementations of a gradient based optimization algorithm is difficult and not feasible so the optimization for this problem is normally done with a non-gradient based method such as the genetic algorithm. Here, we present a novel idea to convert the problem of optimizing on discrete variables into an optimization problem on continuous variables for the optimal well placement. The basic idea is to initialize the optimization problem by putting a well in every gridblock and then optimize the well rate by maximizing the objective function (NPV). As the cost of "drilling a well" detracts from the traditional expression of NPV for production optimization, when the new objective function considering the cost of well drilling is optimized, some wells will be shutin (eliminated) since well rate becomes zero. For two very simple cases where the problem is to determine the optimal location of a water injection well, We can show this problem formulation yields good results with a single injection well remaining after the optimization process.

Gradient based optimization algorithms, with the gradient of a functional or objective function to be optimized most commonly computed by the adjoint (optimal control) method, have been used in both automatic history matching(7; 5; 6; 22; 31; 38; 39; 25; 16; 35) and production optimization(3; 18; 34; 8; 44; 20). However, to the best of our knowledge, the only other work that uses the gradient directly to

solve the optimal well placement problem is the recent paper of Handels et al.(17)

The novel idea for a gradient based solution of the optimal well placement problem presented here is an alternate to the one considered by Handel et al. As they consider only 2D problems, we will do the same. We consider only a very simple example in which we wish to add one or more water injection wells to a 2D reservoir that contains some producing wells. We initialize the optimization problem by putting an injection well in every gridblock that does not contain a producing well and constraining the problem by specifying a maximum total injection rate that must be allocated among the wells remaining at each iteration of the optimization process. For the objective function, besides the traditional definition of net present value (NPV), a drilling cost is assigned for each injection well so the greater the number of injection wells, the greater the drilling cost, the smaller the objective function (NPV). Decreasing the number of injection wells decreases the drilling costs which by itself results in an increase in the objective function but may also cause a decrease in the objective function due to decreased oil production and increased water production. If an injection well rate is decreased to zero, the injection well is eliminated automatically from the system. Initially, all injection wells inject water at the same rate which is determined by dividing the total allowable injection rate by the number of injection wells. Then we use a steepest ascent algorithm to adjust rates to maximize the objective function over a specified reservoir lifetime. As the optimization proceeds, some well rates are decreased to zero and are removed from the system. For the simple examples considered here, we end up with a single injection well, but it is important to note that this will not always be the optimum solution.

5.1 Problem Formulation

Consider a reservoir containing producing and possibly water injection wells, determine the location of new water injection wells to maximize the net present value (NPV) over a specified reservoir lifetime subject to the condition that the total injection rate (q_t STB/D) is fixed and given the oil revenue per unit volume

(r_o in \$/STB), the production water disposal cost per unit volume (r_w in \$/STB), the cost of drilling an injection well (C_{inj} in \$/well) and the annual interest rate b (discount factor is $1+b$).

The best way to formulate the objective function to include drilling costs is not clear, here we try one simple expression for the objective function including the cost of well drilling, which is given by

$$J(\mathbf{q}_{\text{inj}}) = \sum_{k=1}^{N_t} \left[\sum_{j=1}^{N_{\text{prod}}} \left(\frac{r_o q_{o,j}^k - r_w q_{w,j}^k}{(1+b)^{t^k}} \right) \right] \Delta t^k - \sum_{i=1}^{N_{\text{inj}}} \left(\left[\frac{q_{\text{inj},i}}{q_{\text{inj},i} + 10^{-10}} \right] C_{\text{inj}} \right) \quad (5.1)$$

where N_{inj} is the total possible number of water injection wells, N_t is the total number of reservoir simulator time steps, Δt^k represents the time interval of the kth time step in days, t_k representing the cumulative simulation time in days at the end of the kth time step, and $q_{o,j}^k$ and $q_{w,j}^k$, respectively represent the average oil and water production rates of the jth producer on the kth simulation time step. Here, we consider the simple problem where each water injection rate is fixed over the total simulation time. We let $q_{\text{inj},i}$ denote the injection rate of the ith injection well and let $\mathbf{q}_{\text{inj}}$ denote the column vector which has $q_{\text{inj},i}$ as its ith component. Note because the total water injection rate q_t is fixed, there is no need to include a term in the objective function for the cost of water injection. Now the optimal well placement problem stated above can be stated mathematically. Maximize the functional J defined in Eq. 5.1 subject to the constraint that

$$\sum_{i=1}^{N_{\text{inj}}} q_{inj,i} = q_t \quad (5.2)$$

where the total water injection rate, q_t, is specified.

As stated in the introduction, we initialize the optimization problem by putting a water injection well in every gridblock that does not contain a producing well. Note that the first sum on the right side of Eq. 5.1 represents the traditional term for net present value and the second sum represents the total cost of all injection wells drilled. Also note that each individual term in the second sum on the right

side of Eq. 5.1 is a differentiable function of $q_{\text{inj},i}$, which decays to zero as $q_{\text{inj},i} \to 0$. One concern however is that this decay occurs over a small rate interval, $[010^{-6}]$ and for large value of $q_{\text{inj},i}$, we can maximize the objective function (J) using a gradient based optimization algorithm. Also note that if $q_{\text{inj},i} \geq 10^{-6}$ STB/D, the associated coefficient of C_{inj} in Eq. 5.1 is for all practical purposes equal to unity. When applying a gradient based optimization algorithm to maximize the objective function (J), some of the well injection rates go to zero, effectively eliminating the associated terms from the sum that represents the cost of drilling the injection wells in Eq. 5.1. At early iterations, we expect that the sum representing drilling costs will dominate so that most of the increase in the objective function (J) will be due to eliminating injection wells (setting injection rates to zero). However, at later iterations where only a few injection wells are left, the first term may dominate and if this is the case, it may be possible to increase NPV significantly by redistributing the total water injection rate among injection wells. Because the total rate of water injection is fixed, there must be at least one injection well left at the end of the iteration.

5.2 Optimization Algorithm

As stated above, the optimization problem for well placement has an explicit equality constraint on the total water injection rate, but this also implies that the injection rate of each injection well has to be between 0 and the total injection rate, i.e.,

$$0 \leq q_{\text{inj},i} \leq q_t. \tag{5.3}$$

Because of the total injection rate constraint of Eq. 5.2, there are $N_{\text{inj}}-1$ independent injection rates that can be used as controls to be optimized and one dependent injection rate, which we select arbitrarily and denote by $q_{\text{inj},k}$. From Eq. 5.2, we know that

$$q_{\text{inj},k} = q_t - \sum_{i=1, i\neq k}^{N_{\text{inj}}} q_{\text{inj},i}. \tag{5.4}$$

The gradient of the objective function (J) in Eq. 5.1 with respect to the independent controls (independent injection rates) is a column vector which, for $i \neq k$, has as its

ith entry the total derivative of J with respect to $q_{\text{inj},i}$. By the chain rule, this total derivative is given by

$$\frac{dJ}{dq_{\text{inj},i}} = \frac{\partial J}{\partial q_{\text{inj},i}} + \frac{\partial J}{\partial q_{\text{inj},k}} \frac{\partial q_{\text{inj},k}}{\partial q_{\text{inj},i}}. \tag{5.5}$$

From Eq. 5.4, we can see that

$$\frac{\partial q_{\text{inj},k}}{\partial q_{\text{inj},i}} = -1. \tag{5.6}$$

Using Eq. 5.6 in Eq. 5.5, the total derivative of the objective function (J) with respect to $q_{\text{inj},i}$ for $i \neq k$ is given by

$$\frac{dJ}{dq_{\text{inj},i}} = \frac{\partial J}{\partial q_{\text{inj},i}} - \frac{\partial J}{\partial q_{\text{inj},k}}. \tag{5.7}$$

The gradient of J, which represents the steepest ascent direction, can be expressed as

$$\mathbf{p}^{\ell} \equiv \left(\nabla_{\mathbf{q}_{\text{inj}}} J\right)_{\mathbf{q}_{\text{inj}}^{\ell}} = \left[\frac{dJ}{dq_{\text{inj},1}}, \ldots, \frac{dJ}{dq_{\text{inj},k-1}}, \frac{dJ}{dq_{\text{inj},k}}, \frac{dJ}{dq_{\text{inj},k+1}}, \ldots, \frac{dJ}{dq_{\text{inj},N_{\text{inj}}}}\right]^{T}_{\mathbf{q}_{\text{inj}}^{\ell}}, \tag{5.8}$$

where the superscript ℓ is the iteration index.

The updating equation for the injection rate at the lth iteration is

$$\mathbf{q}_{\text{inj}}^{\ell+\mathbf{1}} = \mathbf{q}_{\text{inj}}^{\ell} + \alpha^{\ell} \mathbf{p}^{\ell}, \tag{5.9}$$

where the scalar α^{ℓ} is the step size in the steepest ascent direction which is discussed later.

Up to now, we still do not know the total derivative of J with respect to dependent injection rate $q_{\text{inj},k}$. It can be derived by considering the total injection rate constraint. Summation of every component on both sides of Eq. 5.9 gives that

$$\sum_{i=1}^{N_{\text{inj}}} q_{\text{inj},i}^{\ell+1} = \sum_{i=1}^{N_{\text{inj}}} q_{\text{inj},i}^{\ell} + \alpha^{\ell} \sum_{i=1}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}^{\ell}}. \tag{5.10}$$

In Eq. 5.10, considering the total injection rate constraint, $\sum_{i=1}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}^{\ell}}$ and $\sum_{i=1}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}^{\ell+1}}$ are both equal to q_t. In order to satisfy Eq. 5.10, the summation of the components of $\alpha^{\ell}\mathbf{p}^{\ell}$ is equal to zero, it means

$$\alpha^{\ell} \sum_{i=1}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}^{\ell}} = 0. \tag{5.11}$$

Step size α^{ℓ} is not equal to zero, so Eq. 5.11 can be written as

$$\sum_{i=1}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}^{\ell}} = \frac{dJ}{dq_{\text{inj},k}^{\ell}} + \sum_{i=1, i\neq k}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}^{\ell}} = 0. \tag{5.12}$$

From Eq. 5.12, we can see that

$$\frac{dJ}{dq_{\text{inj},k}^{\ell}} = -\sum_{i=1, i\neq k}^{N_{\text{inj}}} \frac{dJ}{dq_{\text{inj},i}\ell}. \tag{5.13}$$

At this point, we have taken care of the total injection rate constraint. The upper and lower bound constraint of Eq. 5.3 can be satisfied by limiting the step size α^{l} at each iteration so that it is not larger than $\alpha_{\text{max}}^{\ell}$ defined by

$$\alpha_{\text{max}}^{\ell} = \min(\alpha_{\text{max},i}^{\ell}), \tag{5.14}$$

where

$$\alpha_{\text{max},i}^{\ell} = \begin{cases} \frac{-q_{\text{inj},i}^{\ell}}{\frac{dJ}{dq_{\text{inj},i}^{\ell}}} & \text{if } \frac{dJ}{dq_{\text{inj},i}^{\ell}} < 0, \\ \\ \frac{q_t - q_{\text{inj},i}^{\ell}}{\frac{dJ}{dq_{\text{inj},i}^{\ell}}} & \text{if } \frac{dJ}{dq_{\text{inj},i}^{\ell}} > 0, \end{cases} \tag{5.15}$$

with the total derivatives in the preceding equation evaluated at $\mathbf{q}_{\text{inj}}^{\ell}$. In the line search algorithm, we use step size $\alpha_{\text{max}}^{\ell}$ determined by Eq. 5.14. If this step size results in an increase in objective function (J), we accept it. Otherwise, we select a new trial step size by cutting the original step size in half until we find a step size which results in an increase in objective function (J). For the examples considered

here, we are always able to select the step size given by $\alpha^{\ell}_{\max}$. Note this step size decreases the rate of one injection well to zero. Unfortunately, the above procedure can only eliminate one injector at a time, which makes the optimization algorithm very inefficient if we start with an injection well in every gridblock that is not penetrated by a producing well, which is the initialization procedure used here. If the injection rate $q^{\ell}_{\text{inj},i}$ is zero, at the next iteration, we first evaluate $dJ/dq_{\text{inj},i}$ at $\mathbf{q}^{\ell}_{\text{inj}}$. If this derivative is negative, then we fix $\frac{dJ}{dq_{\text{inj},i}}$ equal to zero at subsequent iterations which effectively deletes this well from the system. If the derivative we set to zero corresponds to the dependent injection rate $q_{\text{inj},k}$ then we must choose a new dependent injection rate at the next iteration. Also note that if one of the injection rates reaches its upper bound q_t, this is the only well remaining and iteration terminates. Although procedure appears to be robust, it suffers from the disadvantage that only one potential injection well is removed at every iteration.

5.3 Case Study

For well placement optimization, we present only two simple synthetic cases which do no more than serve to illustrate that our idea for well placement optimization is worthy of further investigation. For both examples considered, the cost to drill an injection well (C_{inj}) is \$200,000.

5.3.1 Homogeneous Reservoir

We consider two-phase flow of oil and water in a homogeneous 2D reservoir with four producers located at the corners. The reservoir consists of 15 by 15 grid blocks and grid block system is uniform with $\triangle x = \triangle y = 200$ ft. The thickness of the reservoir is 50 ft. Permeability is isotropic and equal to 500 md. Porosity is constant and equal to 0.25. Each producer operates under bottom hole pressure control with the bottom hole pressures all fixed equal to 400 psi. The total injection rate is $10,000$ STB/D. The total production time is 10 years. The configuration of the reservoir is shown in Fig. 5.1. Because the reservoir is homogeneous and the well locations are symmetric and produced at the same value of flowing bottom hole

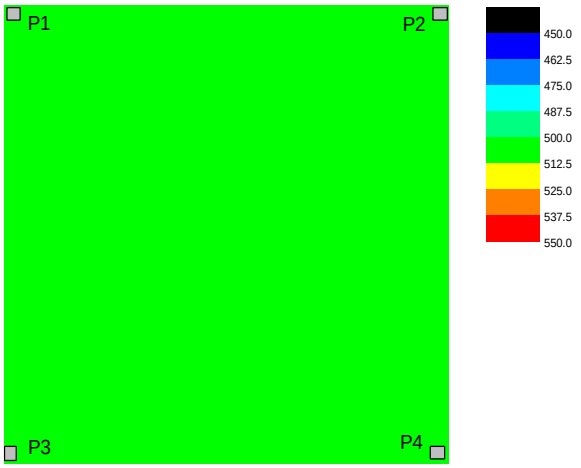


Figure 5.1: Initial well placement and permeability distribution for homogeneous reservoir.

pressure, it suffices to do the optimization on one quarter of the reservoir model.

The relative permeability curve used in the reservoir model is shown in Fig. 5.2. The fluid properties are listed in Table 5.1.

We choose this problem because the solution is obvious, namely, if our optimization algorithm works correctly, we should end up with only one water injection well located at the center of the reservoir, and that is exactly what happened. As shown in Fig. 5.3, there is a single injector left at the end of the optimization process and it is at the center of the reservoir in gridblock $(8, 8)$. NPV as a function of the iteration number is shown in Fig. 5.4. As mentioned earlier, the algorithm generally eliminates only one injector per iteration, but up to iteration 57, the increase in NPV per iteration is much greater than the cost to drill one injection well. This indicates that both terms in Eq. 5.1 are contributing to the increase in NPV at each iteration. The large jump in NPV at iteration 7 corresponds to an iteration where a large step size was used resulting in a significant reallocation of total injection rates among injectors even though only one injection well was removed at this iteration. From iteration 60 through iteration 69, no injector was deleted, the small increase in NPV during these iterations is simply due to reallocating total injection rates

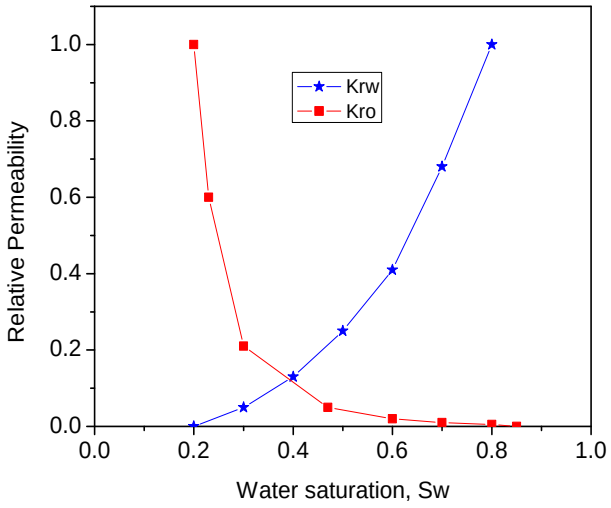


Figure 5.2: Water and oil relative permeability curve .

among the three injectors that remain. At iteration 60, the injection rate at the center injector (gridblock $(8, 8)$) is 4297 STB/D; at iteration 70, the optimization algorithm reduces the number of injection wells to two and at iteration 71, we end up with a single injector in gridblock $(8, 8)$ with the injection rate equal to 10000 STB/day.

5.3.2 Heterogeneous Reservoir

The second example pertains to the placement of injection wells in a heterogeneous reservoir under 2D, two-phase (oil-water) flow. The reservoir contains two producing wells at the locations shown in Fig. 5.5 which also depicts the known permeability distribution. Porosity is constant and equal to 0.25 throughout the reservoir. Note that the producers are in a zone of relatively high permeability. The reservoir consists of 15 by 10 grid blocks and grid block system is uniform with $\triangle x = \triangle y = 200$ ft. The thickness of the reservoir is 50 ft. Again we initialize the optimization problem by putting a water injection well in each of the 148 gridblocks that does not contain a producing well. The initial injection rate of each injector is set equal to $q_t/148$ where the total injection rate $(q_t)$is 5000 STB/day. the total production time is 10 years.

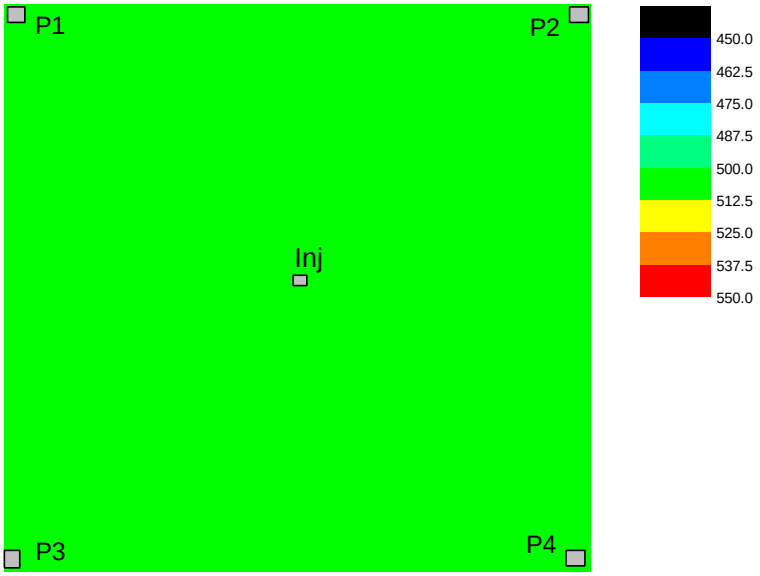


Figure 5.3: Well placement optimization result for homogeneous reservoir.

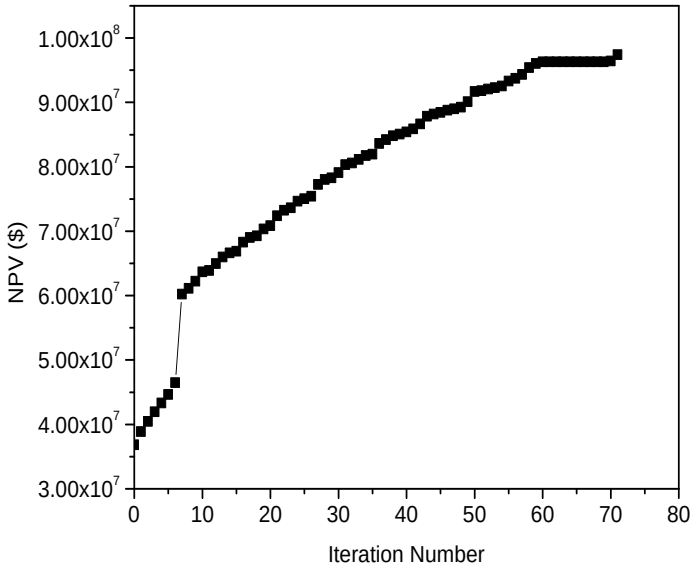


Figure 5.4: Net present value with iteration for homogeneous reservoir.

Table 5.1: Reservoir properties for well placement optimization

Grid block size	200 ft
Thickness	50 ft
ρ_{osc}	56.0 lb/ft^3
ρ_{wsc}	62.4 lb/ft^3
μ_o	2.40 cp
μ_w	0.96cp
B_o	0.9724
B_w	1.0034 rb/stb
Rock Compressibility	4.0×10^{-6} psi^{-1}
Water Compressibility	3.0×10^{-6} psi^{-1}
Oil Compressibility	6.0×10^{-6} psi^{-1}
Top Depth	$10000ft$
Residual oil saturation	0.15
Irreducible water saturation	0.2
Initial water saturation	0.2
Initial reservoir pressure	4000 psi
Total production period	3650 days

The relative permeability curve and the fluid properties used in the heterogeneous reservoir model are same as those used in the homogeneous reservoir.

After optimization, we are again left with a single water injection well at the location shown in Fig. 5.6, but in this case it is not intuitively obvious that a single injection well at the location shown (gridblock $(1, 1)$) is the optimum solution. But this injection well is far from the producers and the preferential permeability direction is almost perpendicular to the flow from the injector to the producers, so the result does seem to be reasonable. Fig. 5.7 shows the net present value as a function of the iteration number. For about the first 60 iterations, the NPV graph can be approximated by a straight line but the slope of this line is about twice the cost of drilling one injection well. However, as only one injection well per iteration is removed, again we conclude that the increase in NPV is due to both reducing the drilling costs and a reallocation of rates among injectors. From iteration 60 to about iteration 120, the slope of the NPV curve is even greater, which indicates that both the reduction in the number of injection wells and the traditional NPV terms (first sum on the right side of Eq. 5.1) are contributing to the increase in NPV but the

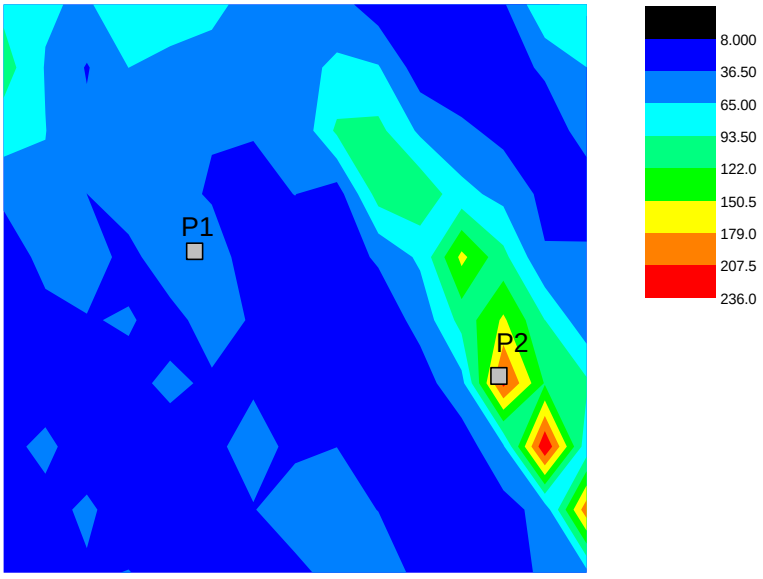


Figure 5.5: Initial well placement and permeability distribution for heterogeneous reservoir.

traditional NPV terms are having a more dominant effect. Note for the last 80 or so iterations the increase in NPV per iteration is relatively small. During this stage of the optimization algorithm, we are not eliminating any injector at each iteration. Instead during most iterations, the numbers of wells are fixed, but the total injection rates are being reallocated among the injectors. For example, at iteration 165, there are four injection wells left, at grid blocks $(1, 1)$, $(1, 10)$, $(11, 10)$ and $(15, 10)$, but the well at gridblock (1,1) has an injection rate equal to 4249 STB/D compared to the total injection rate of 5000 STB/D. The lowest rate is 51 STB/D, but it requires 12 more steepest ascent iterations to eliminate this well.

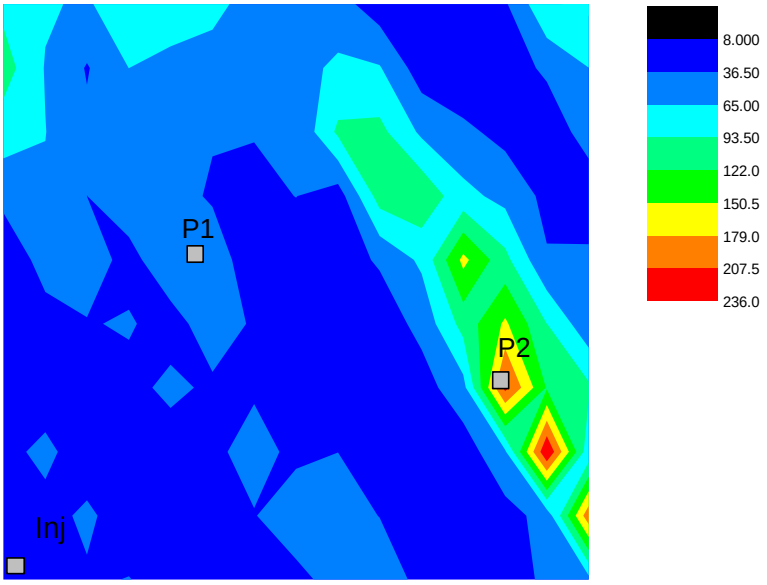


Figure 5.6: Well placement optimization result for heterogeneous reservoir.

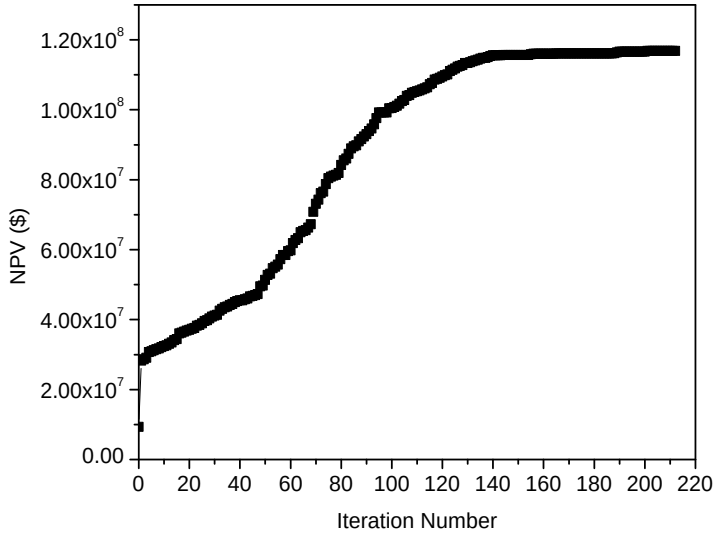


Figure 5.7: Net present value with iteration for heterogeneous reservoir.

CHAPTER 6

CONCLUSIONS

Of the different optimization algorithms compared for production optimization, the steepest ascent algorithm is the most efficient one and it always gives reasonable results. EnKF, when it is treated as an optimization algorithm, requires significantly more reservoir simulator runs and yields a poorer estimate of the optimal controls. SPSA with average stochastic gradient works better than SPSA with a single stochastic gradient, but convergence is slow. The final controls from SPSA using a single stochastic gradient show "bumpy" behavior and is not realistic.

Closed-loop reservoir management with EnKF for data assimilation and the steepest ascent for production optimization based on the central model gives reasonable results for a small test example. The updated permeability and porosity fields capture the geological features of the true fields. The final control is similar to that obtained assuming known geology. Production optimization is a nonlinear problem, at least for the cases considered here. Local maxima are obtained when the controls are at their upper and lower bounds.

As a recommendation for future work, when we optimize the BHP controls, we need consider the constraints on the flow rates of producers. We should try to keep a total injection rate more or less equal to the total production rate. In the closed-loop reservoir management (CLRM), other controls (flow rates of producers, Valve settings and etc) can also be used for production optimization.

The novel idea of converting discrete well placement optimization problems to continuous optimization problems is presented and has been validated by two simple synthetic cases. Compared to traditional non-gradient based optimization algorithms, we apply the gradient based optimization algorithm in the well placement optimization. However, the optimization algorithm for well placement is not

very efficient and generally only one one injector can be eliminated per iteration. The idea we have presented for optimal well placement is both novel and worthy of further serious investigation. In this regard, the first step is to develop a more robust optimization and efficient algorithm than the one presented here.

BIBLIOGRAPHY

[1] A.H. Alhuthali, D. Oyerinde, and A. Datta-Gupta. Optimal waterflood management using rate control, SPE, 102478. 2006.

[2] H. Asheim. Maximization of water sweep efficiency by controlling production and injection rates, SPE, 18365. 1998.

[3] D.R. Brouwer and J.D. Jansen. Dynamic optmization of water flooding with smart wells using optimial control theory. *SPE Journal*, 9:391–402, 2004.

[4] D.R. Brouwer, J.D. Jansen, S. van der Starre, C.P.J.W. van Kruijsdijk, and C.W.J. Berentsen. Recovery increase through water flooding with smart well technology. 2001.

[5] Guy Chavent. Identification of functional parameters in partial differential equations. In *Identification of Parameters in Distributed Systems*, pages 31–48. American Society of Mechanical Engineers, 1974.

[6] Guy M. Chavent, M. Dupuy, and P. Lemonnier. History matching by use of optimal control theory. *Soc. Petrol. Eng. J.*, 15(1):74–86, 1975.

[7] W. H. Chen, G. R. Gavalas, John H. Seinfeld, and Mel L. Wasserman. A new algorithm for automatic history matching. *Soc. Petrol. Eng. J.*, pages 593–608, 1974.

[8] P. de Montleau, A. Cominelli, K. Neylong, D. Rowan, I. Pallister, O. Tesaker, and I. Nygard. Production optimization under constraints using adjoint gradients. *Proceedings of 10th European Conference on the Mathematics of Oil Recovery*, Amsterdam, 4–7 September 2006.

[9] Yannong Dong and Dean S. Oliver. Quantitative use of 4d seismic data for reservoir description. *SPE Journal*, 10(1):51–65, 2005.

[10] D.R. Brouwer et al. Improved reservoir management through optimal control and continuous model updating, (SPE-90149). In *Proceedings of the 2004 SPE Annual Technical Conference and Exhibition*, 2004.

[11] Zandvliet M.J. et al. Bang-bang control in the reservoir flooding. In *Proceeding of the 10th European Conference on the Mathematical Oil Recovery*, 2006.

[12] Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, 99:10143–10162, 1994.

[13] Baris. G., Horne R.N., Rogerd L., and Rosenzweig. Optimization of well placement in a gulf of mexio waterflooding project. *SPE Reservoir Evaluation and Engineeing*, 5(3):229–236, 2002.

[14] Guohua Gao, Gaoming Li, and A. C. Reynolds. A stochastic algorithm for automatic history matching (SPE-90065). In *2004 SPE Annual Technical Conference and Exhibition*, 2004.

[15] Guohua Gao, Mohammad Zafari, and A. C. Reynolds. Quantifying uncertainty for the PUNQ-S3 problem in a Bayesian setting with RML and EnKF. *SPE Journal*, 11(4):506–51, 2006.

[16] Guohua Gao, Mohammad Zefari, and A. C. Reynolds. The Tengiz field history matching problem revisited (SPE-90896). In *2004 SPE Annual Technical Conference and Exhibition*, 2004.

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

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

[19] Peter K. Kitanidis. Quasi-linear geostatistical theory for inversing. *Water Resour. Res.*, 31(10):2411–2419, 1995.

[20] J. F. B. M. Kraaijevanger, P.J. P. Egberts, J. r. Valstar, and H. w. Buurman. Optimal waterflood design using the adjoint method (SPE-105764). In *SPE Reservoir Simulation Symposium*, page 15, 2007.

[21] Gaoming Li and A. C. Reynolds. An iterative ensemble kalman filter for data assimilation, (SPE-109808). In *Proceedings of the 2007 SPE Annual Technical Conference and Exhibition*, 2007.

[22] Ruijian Li, A. C. Reynolds, and D. S. Oliver. History matching of three-phase flow production data. *SPE J.*, 8(4):328–340, 2003.

[23] Naevdal G.and Johnson L.M., Aanonsen S.I., and Vefring E. H. Reservoir monitoring and continuous model updating using ensemble kalman filter. *SPE Journal*, 10(1):66–70, 2005.

[24] Rolf J. Lorentzen, Aina M. Berg, Geir Naevdal, and Erlend H. Vefring. A new approach for dynamic optimization of waterflooding problems, SPE-99690. In *Proceedings of the 2006 SPE Intellegent Energy Conference and Exhibition*, 2006.

[25] Eliana M. Makhlouf, Wen H. Chen, Mel L. Wasserman, and John H. Seinfeld. A general history matching algorithm for three-phase, three-dimensional petroleum reservoirs. *SPE Advanced Technology Series*, 1(2):83–91, 1993.

[26] G. Naevdal, T. Mannseth, and E. H. Vefring. Near-well reservoir monitoring through ensemble Kalman filter (SPE-75235). In *Proceeding of SPE/DOE Improved Oil Recovery Symposium*, 2002.

[27] Nicolas Remy. *Geostatistical Earth Modeling software: User's Manual*. University of Stanford., Stanford, 2004.

[28] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, New York, 1999.

[29] Jude Nwaozo. Dynamic optimization of a water flood reservoir. Master's thesis, University of Oklahoma, Norman, Oklahoma, 2006.

[30] Dean S. Oliver, Nanqun He, and Albert C. Reynolds. Conditioning permeability fields to pressure data. In *European Conference for the Mathematics of Oil Recovery, V*, pages 1–11, 1996.

[31] José R. P. Rodriques. Calculating derivatives for automatic history matching. *Computational Geosciences*, 10:119–136, 2006.

[32] P. Sarma, W.H. Chen, L.J. Durlofsky, and K. Aziz. Production optimization with adjoint models under nonlinear control-state path inequality constraints, SPE 99959. 2006.

[33] P. Sarma, L.J. Durlofsky, and K. Aziz. Implementation of adjoint solution for optimal control of smart wells, SPE 92864. 2005.

[34] Pallav Sarma, Louis J. Durlofsky, Khalid Aziz, and Wen Chen. Efficient real-time reservoir management using adjoint-based optimal control and model updating. *Computational Geosciences*, 10:3–36, 2006.

[35] Pallav Sarma, Louis J. Durlofsky, Khalid Aziz, and Wen H. Chen. A new approach to automatic history matching using kernal pca (SPE-106176). In *2007 SPE Reservoir Simulation Symposium*, 2007.

[36] James C. Spall. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):817–823, 1998.

[37] Kristian Thulin, Gaoming Li, Sigurd Ivar Aanonsen, and Albert C. Reynolds. Estimation of initial fluid contacts by assimilation of production data with EnKF, (SPE-109975). In *Proceedings of the 2007 SPE Annual Technical Conference and Exhibition*, 2007.

[38] M. L. Wasserman, A. S. Emanuel, and J. H. Seinfeld. Practical applications of optimal-control theory to history-matching multiphase simulator models. *Soc. Petrol. Eng. J.*, 15(4):347–355, 1975.

[39] Mel L. Wasserman and A. S. Emanuel. History matching 3-dimensional models using optimal-control theory. *Journal Of Canadian Petroleum Technology*, 15(4):70–77, 1976.

[40] Burak Y., Durlofsky L.J., and Aziz K. Optimization of nonconventional well type, location and trajectory. *SPE Journal*, 8(3):200–210, 2003.

[41] Sudaryanto B.and Yortsos Y.C. Optimization of fluid front dynamics in porous media using rate control.i. equal mobility fluid. 2000.

[42] Sudaryanto B.and Yortsos Y.C. Optimization of displacement in porous media using rate control. 2001.

[43] Mohammad Zafari and Albert C. Reynolds. Assessing the uncertainty in reservoir description and performance predictions with the ensemble Kalman filter, (SPE-95750). In *Proceedings of the 2005 SPE Annual Technical Conference and Exhibition*, 2005.

[44] M.J. Zandvliet, O.H. Bosgra, J.D. Jasen, P.M.J. Van den Hof, and J.F.B.M. Kraaijevvanger. Bang-bang control and sigular arcs in reservoir flooding. *Journal of Petroleum Science and Engineering*, 58:186–200, 2007.

[45] F. Zhang and A. C. Reynolds. Optimization algorithms for automatic history matching of production data. *Proceedings of 8th European Conference on the Mathematics of Oil Recovery*, pages 1–10, 2002.

[46] F. Zhang, J. A. Skjervheim, A. C. Reynolds, and D. S. Oliver. Automatic history matching in a Bayesian framework: Example applications. *SPEREE*, 8(3):214–223, 2005.

[47] Yong Zhao, Gaoming Li, and A. C. Reynolds. Generating facies maps by assimilating production data with the ensemble Kalman filter. In *TUPREP Research Report 24*. The University of Tulsa, 2007.