IPE3

2022

IPE3 2022

ISBN: 978-604-76-2595-6

NOT FOR SALE

Hanoi University of Mining and Geology

Integrated Petroleum Engineering

IPE3

Hanoi, Octorber 6, 2022

ISNB: 978-604-76-2595-6

TRANSPORT PUBLISHING HOUSE

Available online at www.tapchi.humg.edu.vn

IPE 2022

IPE (2022) 107-114

International Conference on Integrated Petroleum Engineering (IPE-2022)

Projected FR-PR Conjugate Gradient Algorithm with Stochastic Simplex Approximated Gradient (StoSAG) for Efficient Waterflooding Optimization

Quang Nguyen [a], Tran Anh Tong [a,*], Nguyen The Vinh [b], Truong Van Tu [b]

[a]*The University of Tulsa, 800 S. Tucker Drive, Tulsa, OK 74104, USA*
[b]*Hanoi University of Mining and Geology, 18 Vien Street, Hanoi, Vietnam*

Abstract

Solving a large-scale life cycle constrained optimization problem is challenging when adjoint gradient information is not available. In this paper, we present a methodology for the solution of a bound-constrained optimization problem, using the Stochastic Simplex Approximated Gradient (StoSAG) combined with Projected Fletcher – Reeves – Polak – Ribiere (FR – PR) Conjugate Gradient method. Conjugate gradient methods have proven to be among the most useful techniques for solving large linear systems of equations, while could also be adapted to solve nonlinear optimization problems. The first nonlinear conjugate gradient method was introduced by Fletcher and Reeves in the 60s, which had gained huge popularity since then. Over the years, many variants of this original scheme have been proposed, and some have proven their efficiency and have been widely used in practice, including the Polak – Ribiere (PR) and its hybrid version with the original Fletcher–Reeves, known as FR – PR. The key features of these algorithms are that they require no matrix storage and are faster than the traditional steepest descent/ascent methods with a superlinear convergence rate. Our discussion is focused on the application of our algorithmic procedures to waterflooding optimization variables are the well controls and the objective function is the life-cycle net present value (NPV) of production. The results show that the Projected Conjugate Gradient algorithm is more efficient than traditional optimization algorithms, saving up to 20% of iterations to converge optimization problems.

Keyword: Constrained Optimization, Numerical Optimization, Stochastic Gradient, Linear Programming, Field Development

1. Introduction

Life-cycle production optimization is a crucial step in closed-loop reservoir management, defined as the maximization/minimization of a predefined objective function via changing the controls of the wells at different control time steps during a reservoir's lifetime (Brouwer et al., 2004; Jansen et al., 2005; Chen et al., 2016). In cases of interest to us, the objective function is evaluated by the reservoir simulator that generates the solution of the discretized system of nonlinear partial differential equations. Even though there are numerous problems of interest, the specific application considered here refers to the optimization of the well controls problem that maximizes the net present value (NPV), either life-cycle NPV or short-term NPV.

The methods to solve this joint optimization problem include both gradient-based and derivative-free methods. Generally speaking, the latter, derivative-free methods, such as the genetic algorithm (Gen and Cheng, 2000) and particle swarm optimization (Isebor et al., 2014) are very computationally expensive compared to their gradient-

* Corresponding author.
E-mail address: tran-anh-tong@utulsa.edu

based counterparts. Especially when the size of the variables becomes considerable, these derivative-free methods will become computationally infeasible. Hence, gradient-based methods are usually a better approach.

However, when dealing with commercial simulators, typically they either have very limited or no adjoint capability at all, thus stochastic gradients are considered as the best alternative approach. The Stochastic Simplex Approximated Gradient (StoSAG, Fonseca et al. (2016)) has proven to provide a sound alternative for production optimization. In this work, the objective function is evaluated by use of a commercial reservoir simulator to generate a solution. As these simulators have limited to no adjoint capability, a stochastic gradient approach was implemented.

Liu and Reynolds, 2019, introduced a modification of StoSAG using sequential quadratic programming (SQP) filter algorithm for waterflooding optimization with nonlinear constraints. The algorithm needs to determine the Hessian matrix numerically by using the quasi-newton method (BFGS), which results in a slower convergence. Therefore, the objective of this work is to perform a deterministic NPV optimization with a stochastic gradient approach (StoSAG) combined with Projected FR – PR Conjugate Gradient method to make the optimization process more efficient and faster. The results section consists of two examples, which are designed to illustrate the robustness and efficiency of our algorithm.

2. Methodology

This section will cover the basic concepts of the general joint optimization problem and the continuous optimization subproblem, as the first step in the general joint optimization two-stage approach.

2.1. Joint Optimization Problem Statement

In joint optimization problems that involve both well locations (denoted as x) and well controls (denoted as u), our goal is to determine wells' optimal controls that can maximize the net present value (NPV) from the reservoir which is necessary for a reservoir development plan. For a three-phase deterministic reservoir model under waterflooding, the NPV objective function is mathematically stated as:

$$J(u) = \sum_{n=1}^{N_c} \frac{\Delta t_n}{(1+b)^{\frac{t_n}{365}}} \left[\sum_{j=1}^{N_P} \left(c_o \bar{q}_{o,j}^n + c_g \bar{q}_{g,j}^n - c_w \bar{q}_{w,j}^n \right) - \sum_{j=1}^{N_I} \left(c_{wi} \bar{q}_{wi,j}^n \right) \right] \tag{1}$$

where u is a N_u-dimensional column vector containing both production and injection wells' controls i.e., $u = [u_1, \ldots, u_{N_u}]^T = [u^{1,1}, u^{2,1}, \ldots, u^{N_c,1}, \ldots, u^{N_c,N_{well}}]^T$, in which N_c is the number of control steps and N_{well} is the total number of wells in the optimization problem. Typical well controls are the water injection rate of injectors, the production rate of producers, and the bottom hole pressure of wells. The time elapsed at the end of the nth control time step is denoted by t_n, and Δt_n is nth control time step size. N_P and N_I denote the number of production and injection wells drilled in the reservoir, respectively. $\bar{q}_{o,j}^n$ *(STB/D),* $\bar{q}_{w,j}^n$ *(STB/D) and* $\bar{q}_{g,j}^n$ *(Mcf/D)*, denote the average oil, water, and gas production rates at the jth well over the nth time step, whereas $\bar{q}_{wi,j}^n$ (STB/D) is the average water injection rate at the jth injection well over the nth control time step. The oil price is represented as c_o *($/STB);* c_g *($/Mscf)* denotes the gas price; c_w ($/STB) represents the disposal cost of produced water; c_{wi} ($/STB) is the water injection cost and b is the annual discount/interest rate. This work considers the well controls as bound linear constraints which occur whenever the constraints and the operational controls are of the same type. Operational controls exist for several practical reasons. Consider the example of a producer operating under a bottom-hole-pressure controlled schedule, there exists a minimum allowable bottom-hole pressure (BHP) required for seamless production. Likewise, to prevent damage to the formation, an injector operating under a rate-controlled schedule is set to have a maximum injection rate. The general production optimization problem can be mathematically stated as the following:

Maximize $J(u),\ u \in R^{N_u}$ (2.a)

Subject to $u_i^{low} \leq u_i \leq u_i^{up},\ i = 1, 2, \ldots, N_u$ (2.b)

where u_i^{up} *and* u_i^{low} denote the upper bound and lower bound of the ith control variable, respectively. In a more general form:

Maximize $J(u),\ u \in R^{N_u}$ (3.a)

Subject to $Au \leq b$ (3.b)

where A is *a* $2N_u \times N_u$ sparse matrix and b is a $2N_u$-column vector resulted from the conversion of bound constraints into general linear constraints

In this work, the producers are operating under BHP-controlled schedules, while the injectors are operating under injection rate-controlled schedules. The economic parameters: oil price c_o = \$63/STB, gas price c_g = \$0.04/Mscf, water disposal cost c_w = \$5/STB, water injection cost c_{wi} =\$5/STB, and annual discount rate b = 0.1.

2.2. The Stochastic Simplex Approximate Gradient (StoSAG)

The algorithm is introduced by Fonseca et al. (2016) as a stochastic gradient approach based on N_p perturbations for robust optimization with geological uncertainty (N_e realizations). In this work, we consider a deterministic optimization without geological uncertainty (i.e. Ne = 1), whose gradient is usually referred to as the "Simplex gradient" in some other literature. Furthermore, at each iteration u_k, with corresponding objective function value $J(u_k)$, each perturbation u_p of the N_p perturbations are assumed to be sampled from the normal distribution with mean u_k:

$$u_p \sim \mathcal{N}(u_k, C_U) \tag{4}$$

where C_U is the spherical covariance with temporal correlation length L:

$$C_U = \begin{cases} \sigma^2 \left(1 - \frac{3}{2}\frac{h}{L} + \frac{1}{2}\left(\frac{h}{L}\right)^3\right), & h \leq L \\ 0, & h > L \end{cases} \tag{5}$$

For each perturbation vector u_p, we can evaluate the corresponding objective function $J(u_p)$. We can then form the corresponding $N_p \times N_u$ perturbation matrix:

$$\Delta U_p = \left[(u_1 - u_k), (u_2 - u_k), \dots, \left(u_{N_p} - u_k\right)\right]^T \tag{6}$$

and the $N_p \times 1$ perturbation vector:

$$\Delta J_p = \left[\left(J(u_1) - J(u_k)\right), \left(J(u_2) - J(u_k)\right), \dots, \left(J\left(u_{N_p}\right) - J(u_k)\right)\right]^T \tag{7}$$

The stochastic gradient could be then computed as:

$$\nabla J(u_k) = \left(\Delta U_p \Delta U_p^T\right)^{-1} \Delta U_p \Delta J_p = \left(\Delta U_p^T\right)^{\dagger} \Delta J_p \tag{8}$$

where in the second equality, the superscript † denotes the Moore-Penrose pseudo-inverse, which could be computed using Singular Value Decomposition (SVD). The second equality is preferred in large-scale problems, but for our purposes, we use the first equality due to its superior computational speed (inverse vs. pseudo-inverse) in small and medium scaled problems. **Algorithm 1** summarizes the overall StoSAG procedures.

Algorithm 1 StoSAG for a continuous deterministic optimization problem

- Preset the number of perturbations N_p, and the perturbation size σ. Generate the block-diagonal covariance matrix C_U from spherical variogram (5)
- Sample the perturbations $u_p \sim \mathcal{N}(u_k, C_U)$. Construct the ($N_p \times N_u$) perturbation matrix ΔU_p (6)
- Construct the ($N_p \times 1$) perturbation vector ΔJ_p as described in (7)

Compute the objective function gradient $\nabla J(u_k)$ using (8).

2.3. Projected Conjugate Gradient (CG)

There are multiple approaches used to solve the bound-constrained continuous optimization subproblem (1) using stochastic gradients. One could perform a logarithmic transform to make the problem (1) unconstrained as follows:

$$w_i = log\left(\frac{u_i - u_i^{low}}{u_i^{up} - u_i}\right) \tag{9}$$

where w_i is the log-transformed variable corresponding to the original well control u_i. The original control u_i could be easily recovered from the inverse logarithmic transform, given by:

$$u_i = \frac{u_i^{up} e^{w_i} - u_i^{low}}{1 + e^{w_i}} \tag{10}$$

However, this method will not result in values at the upper and lower bounds, but rather asymptotic to these limits. Therefore, we could alternatively follow the steepest ascent procedures in the untransformed space with gradient projection as follows:

- Compute the gradient of the objective function $\nabla J(u)$ using the standard StoSAG procedures as described in **Algorithm 1**.
- Perform gradient projection with the steepest ascent to ensure the search direction is feasible within the boxed region created by bound constraints:

$$d_k = (I - A_a^T(A_a A_a^T)^{-1})\nabla J(u_k) \tag{11}$$

where Aa is the matrix whose rows consist of only **active** linear constraints from (2b).

- Inexact line search (backtracking) as the step acceptance criterion:

$$u_{k+1} = u_k + \alpha \frac{d_k}{\|d_k\|_\infty}, \quad \alpha \in (0,1] \quad s.t. \quad J(u_{k+1}) > J(u_k) \tag{12}$$

However, due to the random nature of stochastic gradients, if the direction is far from ascending, more optimization iterations would be required. As a consequence, it leads to an increase in reservoir simulation calls, which ultimately would accumulate up the overall cost. Therefore, we make a slight modification to the above procedures for the continuous optimization subproblem by replacing the steepest ascent with the conjugate gradient (CG) approach, due to its superiority in terms of a number of iterations needed. At each iteration k, the Fletcher-Reeves (FR) CG weighting factor update is:

$$\beta_k^{FR} = \frac{\|\nabla J(u_k)\|_2^2}{\|\nabla J(u_{k-1})\|_2^2} \tag{13}$$

and the Polak-Ribiere (PR) weighting factor update is:

$$\beta_k^{PR} = \frac{\nabla J(u_k)^T\left(\nabla J(u_k) - \nabla J(u_{k-1})\right)}{\|\nabla J(u_{k-1})\|_2^2} \tag{14}$$

The combined weighting factor update strategy for Fletcher-Reeves-Polak-Ribiere (FR-PR) CG is as follows:

$$\beta_k = \begin{cases} -\beta_k^{FR}, & \beta_k^{PR} < -\beta_k^{FR} \\ \beta_k^{PR}, & |\beta_k^{PR}| \le -\beta_k^{FR} \\ \beta_k^{FR}, & \beta_k^{PR} > \beta_k^{FR} \end{cases} \tag{15}$$

The unprojected FR-PR CG ascent search direction is then defined as:

$$d_k^{FR-PR} = \begin{cases} \nabla J(u_k), & k = 0 \\ \nabla J(u_k) + \beta_k d_{k-1}, & k \geq 1 \end{cases} \quad (16)$$

where d_{k-1} is the actual (projected) search direction used in the last iteration. The projected direction at the current iteration can be computed from the gradient projection equation (11) using d_k^{FR-PR} instead of $\nabla J(u_k)$. The overall resultant algorithm is summarized in **Algorithm 2**.

Algorithm 2 *Projected FR CG for bound-constrained continuous maximization*

1. Preset $n_{grad,max}$, $n_{cuts,max}$, maximum backtracking step size α_{max}, step size reduction factor ρ, and the maximum number of reservoir simulation runs N_{sim}. Specify the number of perturbations N_{pert}, and perturbation size σ for StoSAG.
2. Initialize: iteration index k = 0, and the design vector u_0.
3. Set n_{grad} = 1 and n_{cuts} = 0.
4. Compute the gradient of the objective function $\nabla J(u_k)$ using the standard StoSAG procedures as described in **Algorithm 1**. Go to Step 5.
5. Compute the unprojected Fletcher-Reeves-Polak-Ribiere (FR-PR) ascent direction d_k^{FR-PR} using (13). Go to Step 6.
6. Perform gradient projection with the steepest ascent to ensure the search direction is feasible within the boxed region created by bound constraints:

$$d_k = (I - A_a^T(A_a A_a^T)^{-1}) d_k^{FR-PR}$$

7. Perform inexact line search (backtracking) as described in (10). If backtracking succeeds, go to Step 8. Else if backtracking fails with $n_{cuts} = n_{cuts,max}$ then:
   - If $n_{grad} \leq n_{grad,max}$, go back to Step 4.
   - Else, terminate the algorithm.
8. Check for convergence. If not, set k = k + 1 update $u_k = u_k + 1$, save the following parameters as prior information: $d_{k-1} = d_k$, $\nabla J(u_{k-1}) = \nabla J(u_k)$ and go back to Step 4.

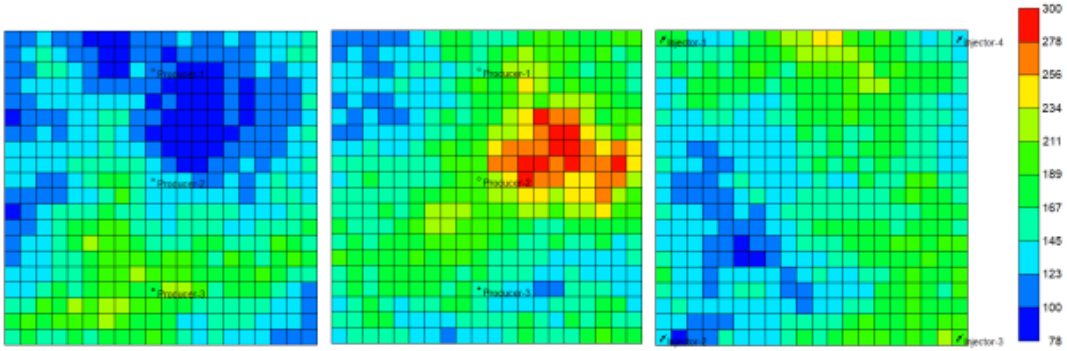
Computational Results

Example 1:

In this example, we will demonstrate the computational superiority of FR-PR CG over the traditional steepest ascent. This case study considers a three-phase reservoir in a 20×20×3 uniform Cartesian grid, under a water-alternating-gas (WAG) injection process. Each grid has a dimension of 100×100×30 ft. There are two wells, 1 injector, and 1 producer, both vertical and fully perforated. The initial pressure of the reservoir is 4400 psi, with the irreducible water saturation and residual oil saturation bringing 0.3 and 0.2, respectively. In order to model the WAG process with CO2 injection, we need to run a compositional model using CMG-GEM simulator.

Table 1. Lower and upper limits of the well controls in Example 1

Variable	Lower Limit	Upper Limit
Producer BHP (psi)	500	2000
Water injection rate (stb/d)	1000	4000
CO2 injection rate (MMscf/d)	5	20



(a) Layer 1 (b) Layer 2 (c) Layer 3

Fig. 1. Permeability distribution of each layer in Example 1.

Table 1 summarizes the well controls in lower and upper bounds. We perform a five-cycle WAG optimization for a total of 3,600 days in their lifetime, in which each cycle consists of a half-cycle water injection period followed by a half-cycle of the CO_2 injection period. Hence, in this example, the number of well controls is 10 × 2 = 20, and the number of well location integers is 2 × 2 = 4. During the gradient calculations using StoSAG procedures, we use the number of perturbations N_p = 10, with the spherical variogram with the temporal correlation length of 5. For the NPV computations, the oil price is set to \$63/STB, the water disposal cost is \$5/STB, and the water injection cost is set to \$5/STB. The gas price is \$0.04/Mscf (excluding CO_2), CO_2 injection cost is \$1,5/Mscf, CO_2 treatment (disposal) cost is \$0.35/Mscf and the annual discount rate is 0:1. All producers are initialized at 1,250 psi, while all injectors are initialized at 2,500 STB/d for water injection rates and 12.5 MMscf/d for CO_2 injection rates.

To visually demonstrate the superiority of projected CG over the traditional steepest ascent, Figure 2 compares the optimization results of these two algorithms using the same number of perturbations (N_p = 10). As projected CG takes four optimization iterations less than the steepest ascent, roughly 40 reservoir simulations are saved.

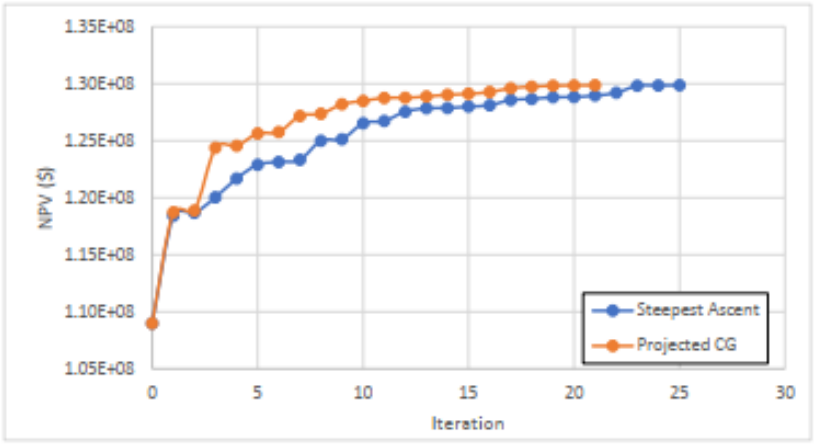


Fig. 2. Comparison between Projected CG and Steepest Ascent, Example 1

Example 2:

In this section, we will demonstrate the application of our algorithm onto a 3D synthetic reservoir modeled in CMG IMEX, shown in Figure 3. The reservoir grid dimension is 20×20×3, with four injectors located at the corners, and one producer located at the center. The injectors are perforated at Layer 3, while the producer is perforated at Layers 1 and 2. The initial reservoir pressure is 4,400 psi. The waterflooding process is simulated for a total lifetime of 3,600 days, divided into 10 equal control time steps (360-day each). Therefore, the total number of well controls is 10×5 = 50. As usual, the injectors are rate-controlled, whereas the producer is BHP-controlled. The upper and lower limits of the design variables are summarized in Table 2.

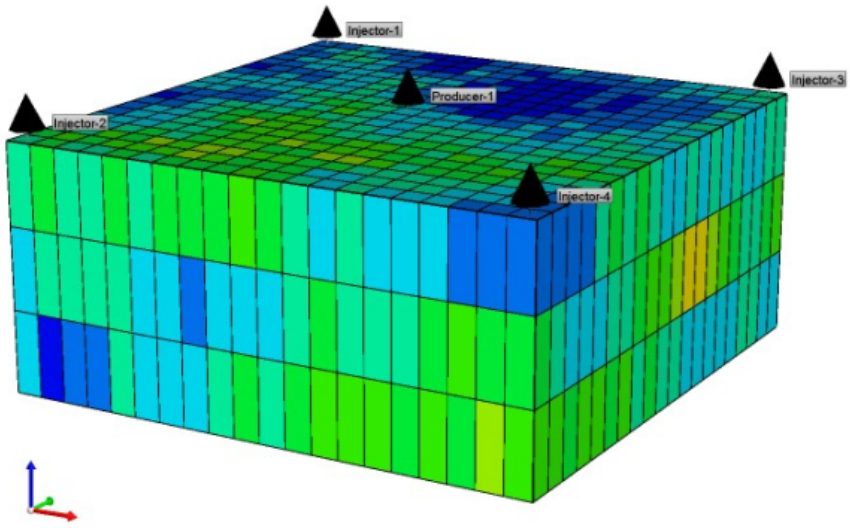
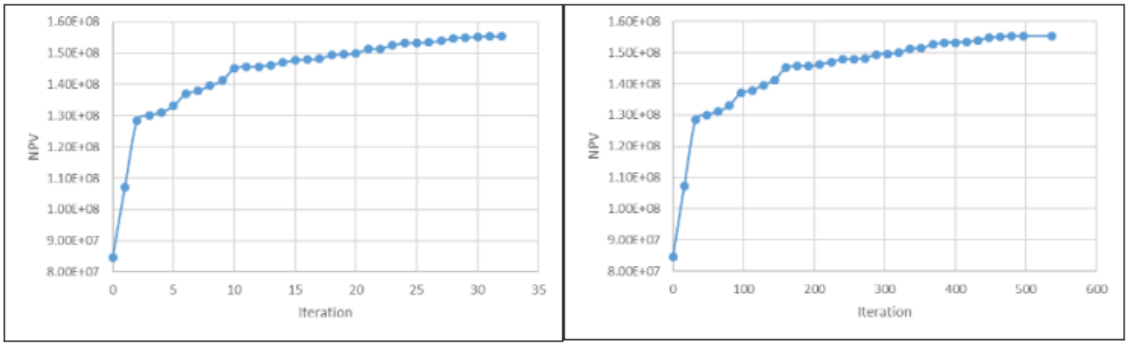


Fig. 3. 3D schematic of the synthetic reservoir.

Table 2. Lower and upper limits of the well controls in Example 2

Variable	Lower Limit	Upper Limit
Producer BHP (psi)	2000	4000
Water injection rate (stb/d)	0	1200

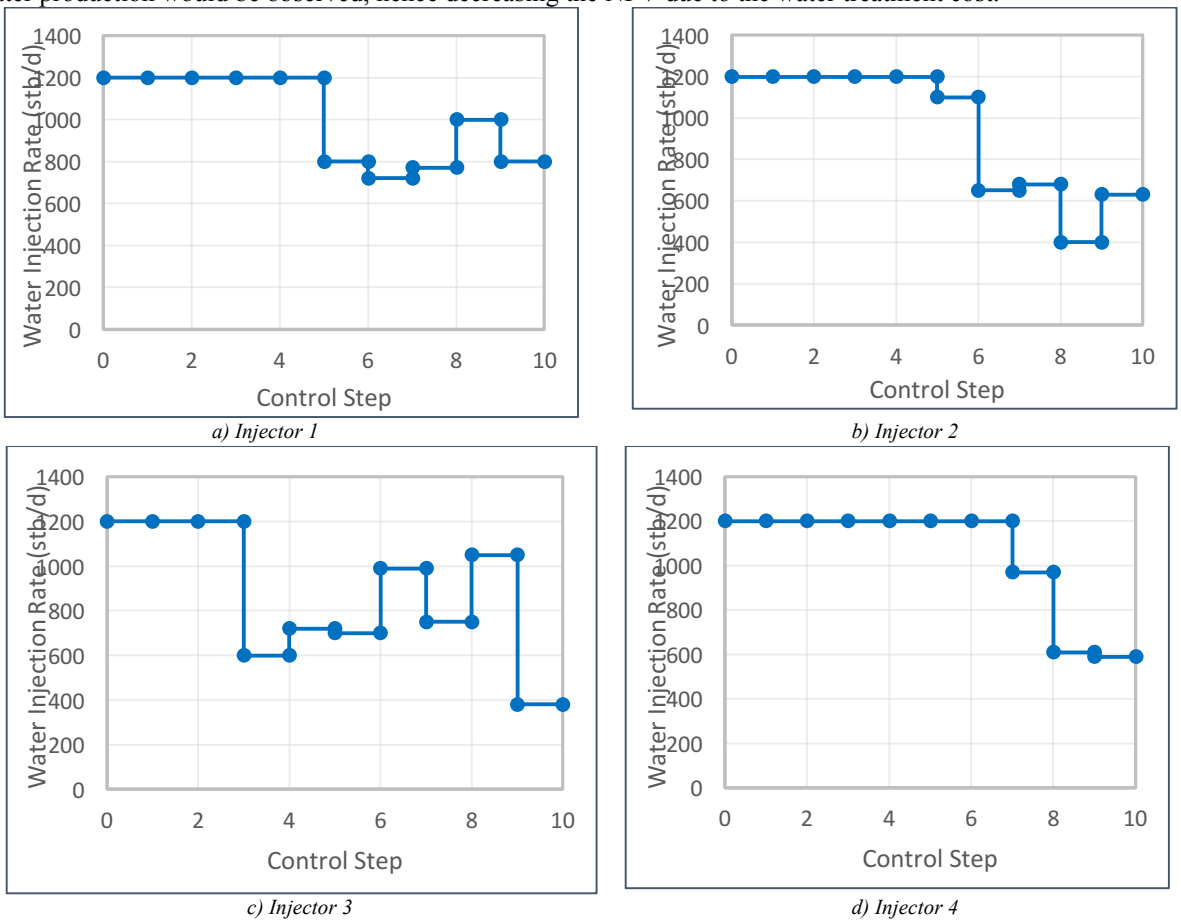
Before optimization, we normalized our design variables to the scale [0,1]. The StoSAG parameters in normalized scale are Np = 15, σ = 0.01, L = 5. In normalized scale, the design variables are initialized to be 0:5, resulting in the corresponding NPV of $\$8.45\times10^7$. Figure 4 shows the optimization results from iteration to iteration. The algorithm converges after 33 optimization runs (or 537 reservoir simulations). The algorithm has improved the NPV to its optimal value of $\$1.55 \times 10^8$.



(a) NPV vs. number of iterations (b) NPV vs. number of simulations

Fig. 4. NPV optimization results.

The optimal well controls are shown in Figure 5 and Figure 6 for injectors and producers, respectively. All four injectors are operating at maximum rates at the early times, and decreasing over time to reduce the water production at the producer. The producer is set to operate at the upper bound throughout the lifetime. This can be explained by the long total lifetime (3,600 days), if the BHP is lowered at any of the control time steps, more water production would be observed, hence decreasing the NPV due to the water treatment cost.



a) Injector 1 *b) Injector 2*

c) Injector 3 *d) Injector 4*

Fig. 5. Optimal Injection Schedule

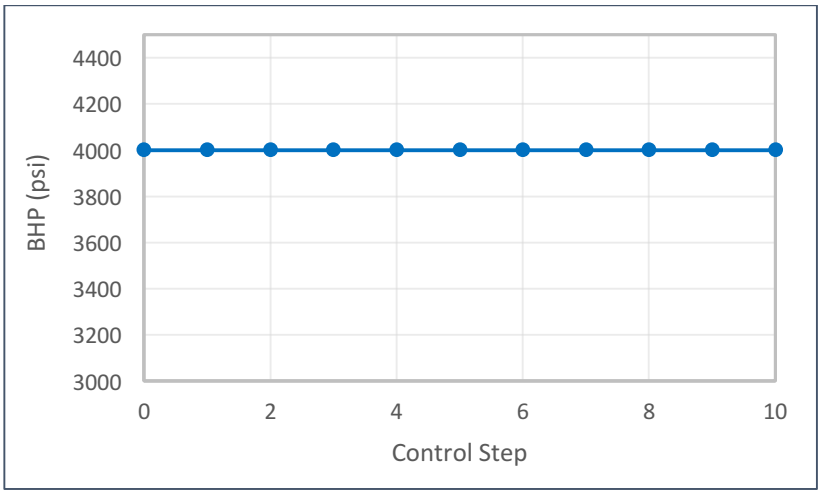


Fig. 6. Optimal Production Schedule of Producer

The water saturation distributions are shown in Figure 7, which compares the initial and final water saturation distribution. It can be seen that, at the final time, most of the reservoir volume is full of water, with a small zone in Layer 1 still containing some oil.

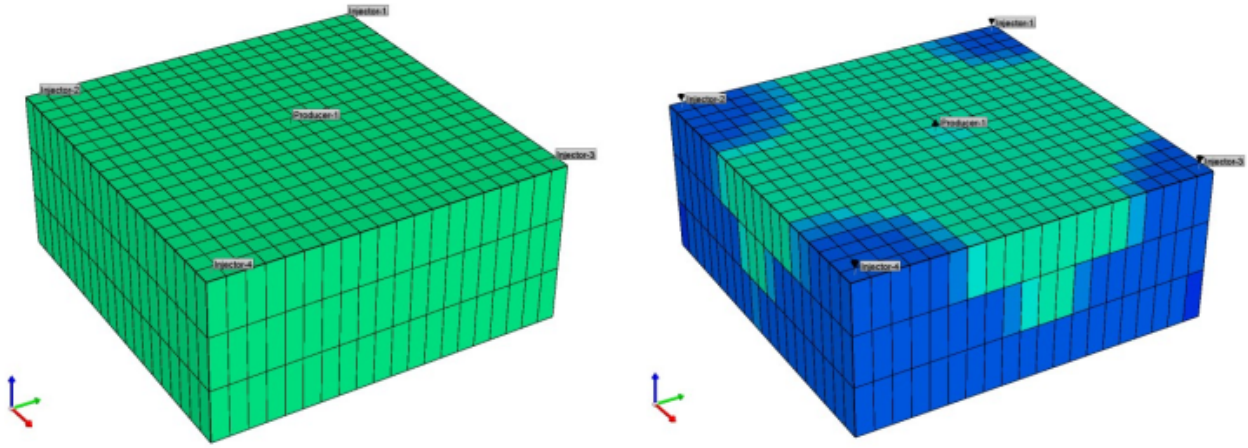


Fig.7. Comparison of water saturation distribution.

3. Conclusions

The projected CG has a better performance than the traditional steepest ascent, and required fewer iterations to converge.

The solution is unique in the simple case (Example 1), but when there are multiple wells presented (Example 2), we would encounter non-unique solutions. Especially if local homogeneity is observed.

References

Brouwer, D. R., G. Nævdal, J. D. Jansen, E. H. Vefring, and C. P. J. W. van Kruijsdijk, Improved reservoir management through optimal control and continuous model updating, in Proceedings of the SPE Annual Technical Conference and Exhibition, Houston, Texas, 26-29 September, SPE 90149, 2004.

Fonseca, R. R. M., B. Chen, J. D. Jansen, and A. Reynolds, A stochastic simplex approximate gradient (StoSAG) for optimization under uncertainty, International Journal for Numerical Methods in Engineering, 2016.

Gen, M. and R. Cheng, Genetic algorithms and engineering optimization, vol. 7, John Wiley & Sons, 2000.

Isebor, O. J., L. J. Durlofsky, and D. Echeverría Ciaurri, A derivative-free methodology with local and global search for the constrained joint optimization of well locations and controls, Computational Geosciences, 18(3), 463–482, 2014.

Liu, Z. and A. C. Reynolds, An SQP-Filter Algorithm with an Improved Stochastic Gradient for Robust Life-Cycle Optimization Problems with Nonlinear Constraints, SPE Reservoir Simulation Conference, Galveston, Texas, USA, 10-11 April, SPE 193925, 2019.