SPE 141300

Enhancement of the CPR Preconditioner for Efficient Solution of the Adjoint Equation

Choongyong Han, SPE, Chevron ETC, John Wallis, SPE, Wallis Consulting Inc., Pallav Sarma, SPE, Gary Li, SPE, Mark L. Schrader, SPE, Wen Chen, SPE, Chevron ETC

This paper was prepared for presentation at the SPE Reservoir Simulation Symposium held in The Woodlands, Texas, USA, 21–23 February 2011.

Abstract

It is well known that the adjoint approach is the most efficient approach for gradient calculation, which can be used with gradient-based optimization techniques to solve various optimization problems such as the production optimization problem, history-matching problem, etc. The adjoint equation to be solved in the approach is a linear equation formed with the "transpose" of the Jacobian matrix from a fully implicit reservoir simulator. For a large and/or complex reservoir model, generalized preconditioners often prove impractical for solving the adjoint equation. Preconditioners specialized for reservoir simulation such as Constrained Pressure Residual (CPR) exploit properties of the Jacobian matrix in order to accelerate convergence, so they cannot be directly applied to the adjoint equation. To overcome this challenge, we have developed a new two-stage preconditioner for efficient solution of the adjoint equation (named CPRA).

The CPRA preconditioner has been coupled with an Algebraic Multi Grid (AMG) linear solver and implemented in Chevron's in-house reservoir simulator. The AMG solver is well known for its outstanding capability to solve the pressure equation of complex reservoir models; and solving the linear system with the "transpose" of pressure matrix is one of the two-stages of construction of the CPRA preconditioner.

Through test cases, we have confirmed that the CPRA/AMG solver with GMRES acceleration solves the adjoint equation very efficiently with reasonable number of linear solver iterations. Adjoint simulations to calculate the gradients with the CPRA/AMG solver take about the same time at most as corresponding CPR/AMG forward simulations. Accuracy of the solutions has also been confirmed by verifying the gradients against solutions with a direct solver. A production optimization case study for a real field using the CPRA/AMG solver has further validated its accuracy, efficiency, and the capability to perform long term optimization for large, complex reservoir models at low computational cost.

Introduction

Practical production optimization problems typically involve large, highly complex reservoir models, thousands of unknowns and many nonlinear constraints, which makes the numerical calculation of gradients for the optimization process impractical. Adjoint models can be used for the calculation of gradients very efficiently (Brouwer and Jansen, 2002). Direct coding of the adjoint model is, however, complex and time consuming, and the code is dependent on the forward model in the sense that it must be updated whenever the forward model is modified.

Sarma et al. (2005) proposed a new adjoint construction procedure that makes it relatively easy to construct the adjoint equation and has the additional advantage of making the adjoint simulation code independent of the forward simulation code. For a fully implicit forward model and specific forms of the cost function and nonlinear constraints, all information necessary for the adjoint run is calculated and stored during the forward simulation itself. The adjoint run then requires only the appropriate assembling of this information to calculate gradients of the cost function with respect to control variables at each control step. During the assembling procedure, adjoint equation at each timestep should be solved for Lagrange multipliers using information obtained from forward simulation such as Jacobians.

The adjoint equation at each timestep is a linear equation with "transpose" of Jacobian matrix evaluated at convergence at that timestep. The adjoint equation with transpose of Jacobian matrix, unfortunately, cannot be solved by a linear solver using a preconditioner specially designed and optimized for reservoir simulation such as constrained pressure residual (CPR) (Wallis, 1983; Wallis et al., 1985) or nested factorization method (Appleyard and Cheshire, 1983). This is because the preconditioners exploit properties of the original Jacobian matrix in order to accelerate convergence.

The two-stage CPR preconditioner developed by Wallis (1983) has proven to be a very powerful preconditioner for reservoir simulation, especially when combined with an Algebraic Multi Gird (AMG) linear solver (Cao et al., 2005). The AMG solver has shown its outstanding capability to solve the pressure equation of complex reservoir models in the construction of the CPR preconditioner. Therefore, there arises a strong desire to have the CPR preconditioner/AMG solver working for the adjoint equation as well. To fulfill this demand, a new two-stage preconditioner has been developed for efficient solution of the adjoint equation (named CPRA) and has been coupled with an AMG linear solver in the Chevron's in-house reservoir simulator, CHEARS. The AMG solver is used to solve the linear system with the "transpose" of pressure matrix, which is one of the two-stages of construction of the CPRA preconditioner.

In this paper, mathematical derivation of the CPRA preconditioner will be presented first. Computational performance of the CPRA/AMG solver will then be shown using three test cases. Finally, a production optimization case study for a real field using the CPRA/AMG solver will be presented to further validate its accuracy, efficiency, and the capability to perform long term optimization for a large, complex, reservoir models at low computational cost.

Mathematical Derivation of the CPRA Preconditioner

Definition of the optimization problem

The optimization problem discussed above requires finding a sequence of control vectors u^n (of length m) for $n = 0,1,...,N-1$, where n is the control step index and N is the total number of control steps, to maximize (or minimize) a performance measure $J\left(u^0,...,u^{N-1}\right)$ (Sarma et. al., 2005). The optimization can be described very generally with the following mathematical formulation:

$$
\begin{aligned}
&\max_{u^n}\left[J=\phi\left(x^N\right)+\sum_{n=0}^{N-1}L^n\left(x^{n+1},u^n\right)\right]\forall n\in(0,..,N-1)\\
&\text{subject to:}\\
&g^n(x^{n+1},x^n,u^n)=0 \qquad \forall n\in(0,..,N-1)\\
&x^0=x_0 \qquad \text{(Initial Condition)}\\
&c^n(x^{n+1},u^n)\le 0 \qquad \forall n\in(0,..,N-1)\\
&Au^n\le b \qquad \forall n\in(0,..,N-1)\\
&LB\le u^n\le UB \qquad \forall n\in(0,..,N-1)
\end{aligned}
\tag{1}
$$

Here, x^n refers to the dynamic states of the system, such as pressures, saturations, compositions etc. The cost function J consists of two terms. The first term ϕ is only a function of the dynamic states of the last control step; in a production optimization application it could represent, for example, an abandonment cost. The second term, which is a summation over all control steps, consists of the kernel L^n known as the Lagrangian in control literature (Stengel, 1985). For production optimization, it could include the oil and water rates or some function of the saturations (for sweep efficiency); for history-matching problems, it will represent the history-matching error. Since L^n usually consists of well parameters or quantities that are functions of well parameters, it is written here in a fully implicit form.

The set of equations g^n together with the initial condition define the dynamic system, which are basically the reservoir simulation equations for each grid block at each time step. These equations relate the cost function J to the controls u^n through their mutual dependence on the states x^n. The last three equations of Equation (1) refer to the additional constraints for the controls, that is, nonlinear constraints, linear constraints, and bounds on controls. These will not be discussed in this paper, and readers can look at Sarma et al. (2008) for details on these additional constraints.

Adjoint equation

In order to solve the above optimization problem using gradient-based optimization algorithms, the gradient of J with respect to u^n are required. This can be calculated very efficiently using the adjoint method. The adjoint model equations are obtained from the necessary conditions of optimality of the optimization problem defined by Equation (1). These necessary conditions of optimality are obtained from the classical theory of calculus of variations. For a relatively simple treatment of this subject, refer to Stengel (1985). The adjoint equations to be solved in order to calculate the gradient of the objective function with respect to the control variables are as follows (Sarma et al., 2005):

$$\lambda^{Tn} = -\left[\frac{\partial L^{n-1}}{\partial x^n} + \lambda^{T(n+1)}\frac{\partial g^n}{\partial x^n}\right]\left[\frac{\partial g^{n-1}}{\partial x^n}\right]^{-1} \quad \forall n \in (1,\cdots,N-1)$$
$$\lambda^{TN} = -\left[\frac{\partial \Phi}{\partial x^N} + \frac{\partial L^{N-1}}{\partial x^N}\right]\left[\frac{\partial g^{N-1}}{\partial x^N}\right]^{-1} \quad (Final\ \ Condition) \tag{2}$$

Here λ are the Lagrange multipliers and *T* denotes transpose. Because λ^n depends on λ^{n+1}, the Lagrange multipliers for the last timestep must be calculated first. It is for this reason that the adjoint equation is solved backwards in time and the adjoint simulation to get gradients of the cost function with respect to control variables with the Lagrange multipliers is often called the backward simulation. The required gradients of the cost function with respect to the control variables are given as:

$$\frac{\partial J}{\partial u^n} = \frac{\partial J_A}{\partial u^n} = \left[\frac{\partial L^n}{\partial u^n} + \lambda^{T(n+1)}\frac{\partial g^n}{\partial u^n}\right] \quad \forall n \in (0,\cdots,N-1) \tag{3}$$

Here J_A is the augmented cost function (see Sarma et al., 2005). Please note that $\frac{\partial g^{n-1}}{\partial x^n}$ in Equation (2) is actually the Jacobian matrix ($\overline{\overline{J}}$) evaluated at convergence at timestep *n* in the forward reservoir simulation. Therefore, Equation (2) at each timestep can be rewritten in the following simplified form:

$$A^T z = d, \tag{4}$$

where

$$A = \overline{\overline{J}}; z = \lambda$$
$$d = \begin{cases} -\left[\frac{\partial L^{n-1}}{\partial x^n} + \lambda^{T(n+1)}\frac{\partial g^n}{\partial x^n}\right]^T & \forall n \in (1,\cdots,N-1) \\ -\left[\frac{\partial \Phi}{\partial x^N} + \frac{\partial L^{N-1}}{\partial x^N}\right]^T & (Final\ \ Condition) \end{cases} \tag{5}$$

Equation (4) is a linear equation with transpose of Jacobian matrix. Because the Jacobian matrix is transposed, the adjoint equation cannot be solved by a linear solver directly using a preconditioner such as the CPR preconditioner which exploits properties of the original Jacobian matrix in order to accelerate convergence. Because the two-stage CPR preconditioner has proven to be a very powerful preconditioner for reservoir simulation, it has been enhanced to solve the adjoint equation in this study and named the CPRA preconditioner.

The CPR preconditioner

Here a brief description of the CPR preconditioner is provided from which the CPRA preconditioner is derived. Let *A* be the fully implicit reservoir matrix.

$$A = \begin{bmatrix} A_{ss} & A_{sp} \\ A_{ps} & A_{pp} \end{bmatrix} \tag{6}$$

Here subscript *p* refers to pressure and subscript *s* to the remaining unknowns.

To solve a linear equation, $Ay = b$, for the forward reservoir simulation, the True-IMPES (Wallis et al., 2006) pressure reduction is performed using the matrix *N*:

$$N = \begin{bmatrix} I & 0 \\ -Colsum(A_{ps})Colsum^{-1}(A_{ss}) & I \end{bmatrix}, \tag{7}$$

where *Colsum*(*B*) (*B* is any blocked matrix where the sub-matrix blocks are all the same size) is defined to be the block diagonal matrix having as the *i-th* main diagonal block the sum of all the blocks in the *i-th* block column of *B*.
Then,

$$\tilde{A}y = \tilde{b}, \tag{8}$$

where $\tilde{A} = NA$ and $\tilde{b} = Nb$.
The CPR pressure matrix is

$$A_{pp} = C^T \tilde{A} C, \tag{9}$$

where C is the $(n_{cell} \text{ x } n_{eqn})$ x n_{cell} (n_{cell} is the number of cells and n_{eqn} is the number of equation per cell) pressure prolongation matrix.
The CPR preconditioning matrix for $\tilde{A}$ is given by (Wallis, 1983):

$$M_{CPR}^{-1} = \tilde{M}^{-1}[\, I - \tilde{A}C\, A_{pp}^{-1}\, C^T\,] + C\, A_{pp}^{-1}\, C^T\,, \tag{10}$$

where $\tilde{M}$ is an approximate factorization for $\tilde{A}$.

Derivation of the CPRA preconditioner

Now the CPR preconditioner is enhanced to solve the adjoint equation (Equation (4)) which is the same as:

$$(\tilde{A})^T\, y = d\,, \tag{11}$$

where $y = \left(N^T\right)^{-1} z$

Here the CPRA preconditioner M_{CPR}^{-T}, is proposed to solve Equation (11) as follows:

$$M_{CPR}^{-T} = [\, I - C(A_{pp}^T)^{-1}\, C^T\, (\tilde{A})^T\,]\, (\hat{M})^{-1} + C(A_{pp}^T)^{-1} C^T\,, \tag{12}$$

where $\hat{M}$ is an approximate factorization of $(\tilde{A})^T$. By rearranging Equation (12), we get:

$$M_{CPR}^{-T} = C(A_{pp}^T)^{-1}\, C^T[\, I - (\tilde{A})^T\, (\hat{M})^{-1}] + (\hat{M})^{-1} \tag{13}$$

Therefore, for construction of the CPRA preconditioner for $(\tilde{A})^T$, the following two stage preconditioning algorithm for residual r ($= d - (\tilde{A})^T\, y$) has been developed:

1) solve $\hat{M}q = r$ for q (first stage: apply ILU(k) to $(\tilde{A})^T$)
2) $s = r - (\tilde{A})^T\, q$
3) $v = C^T\, s$
4) solve $(A_{pp})^T\ p = v$ for p (second stage: single variable solve using AMG)
5) $y = C\, p + q$

Steps 1 to 5 are repeated using GMRES algorithm (Saad and Schultz, 1986) until converged solution y is obtained. With the converged solution y, the solution z to Equation (4) can be obtained by $z = N^T\, y$.

Validation and Performance of the CPRA/AMG Solver

This new CPRA algorithm has been implemented with the AMG solver in the Chevron's in-house reservoir simulator, CHEARS. Following three cases have been tested to confirm the performance of the new algorithm:

Case 1: Synthetic case, 10x10x3 (=300) grids with 264 active grids, BLACKOIL option, 2 water injectors and 1 producer
Case 2: Field P case, 86x100x8 (=68,800) grids with 39,265 active grids, BLACKOIL option, 5 water injectors, 5 producers
Case 3: Field L case, 96x73x150 (=1,051,200) grids with 164,281 active grids, BLACKOIL option, 11 water injectors, 15 producers

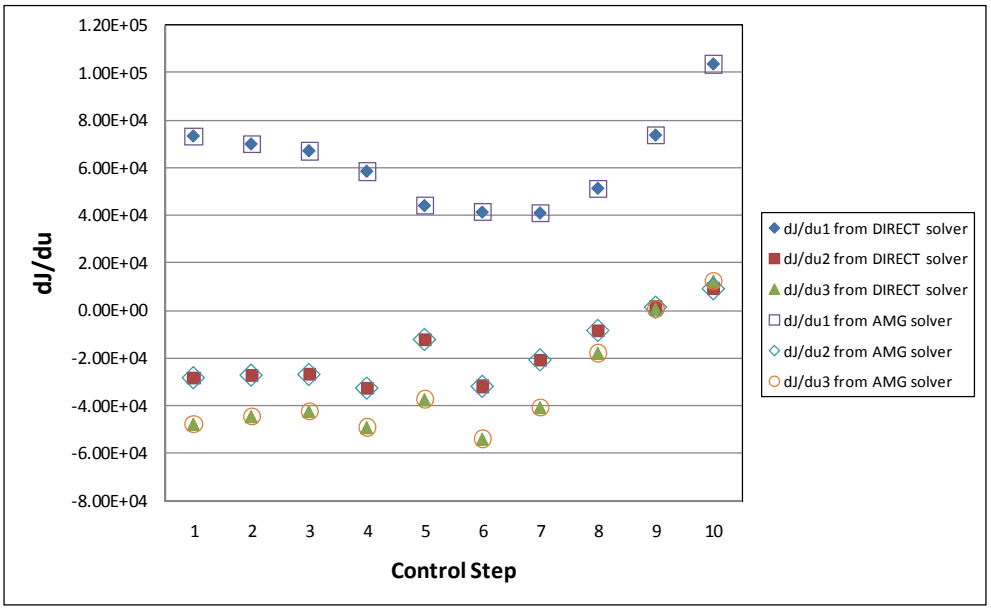

Figure 1 Comparison of dJ/du between DIRECT solver and AMG solver with the CPRA preconditioner

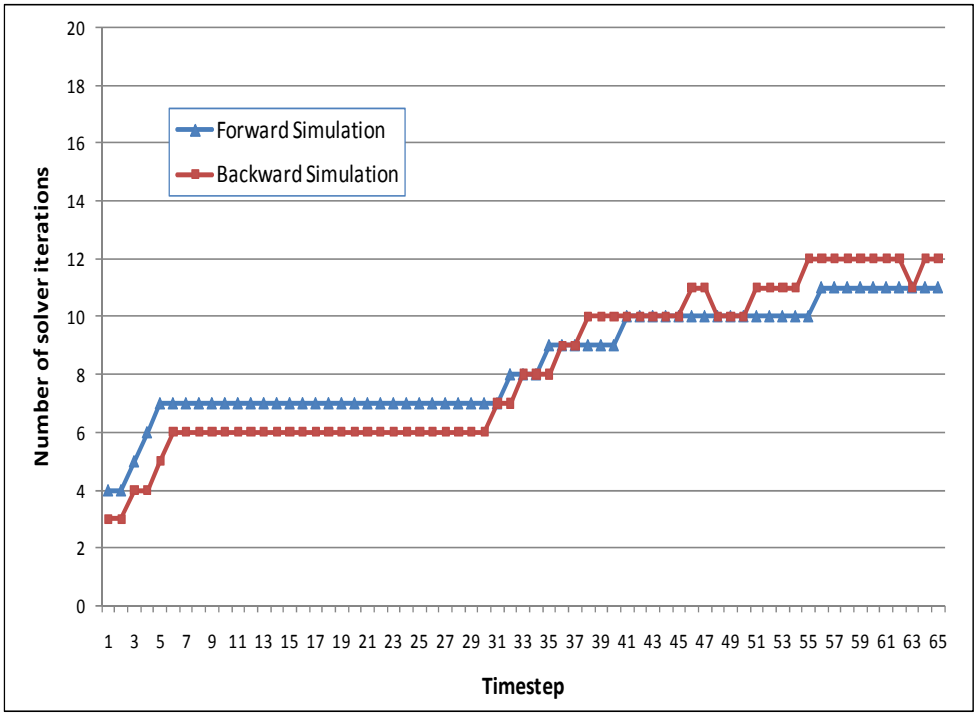

Figure 2 Number of AMG solver iterations with timestep in the forward and backward simulations for Case 1

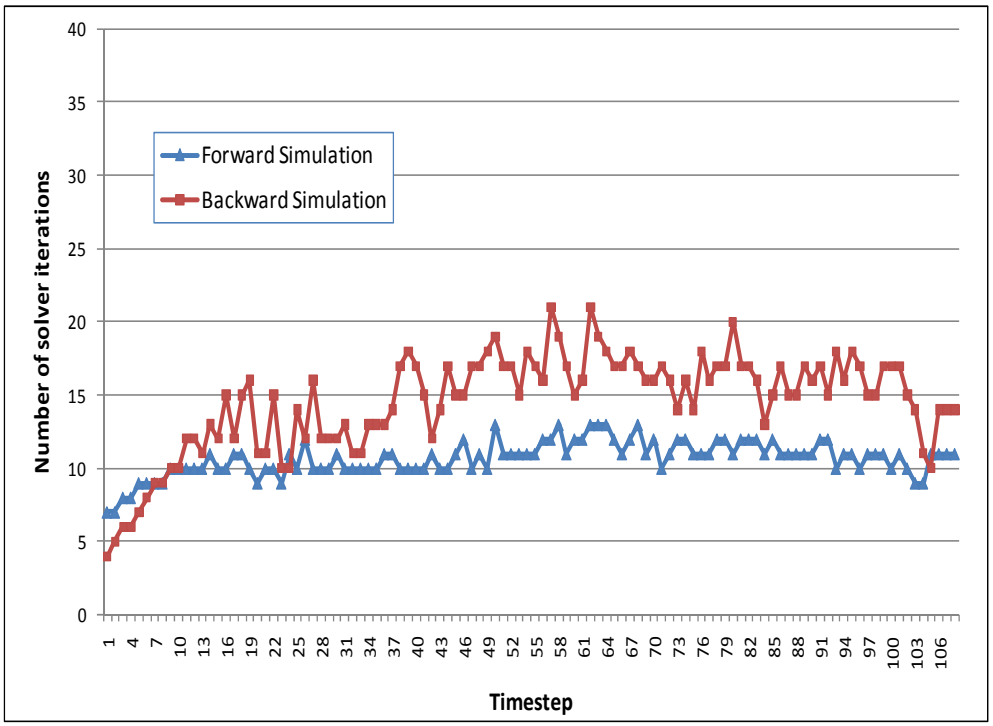

Figure 3 Number of AMG solver iterations with timestep in the forward and backward simulations for Case 2

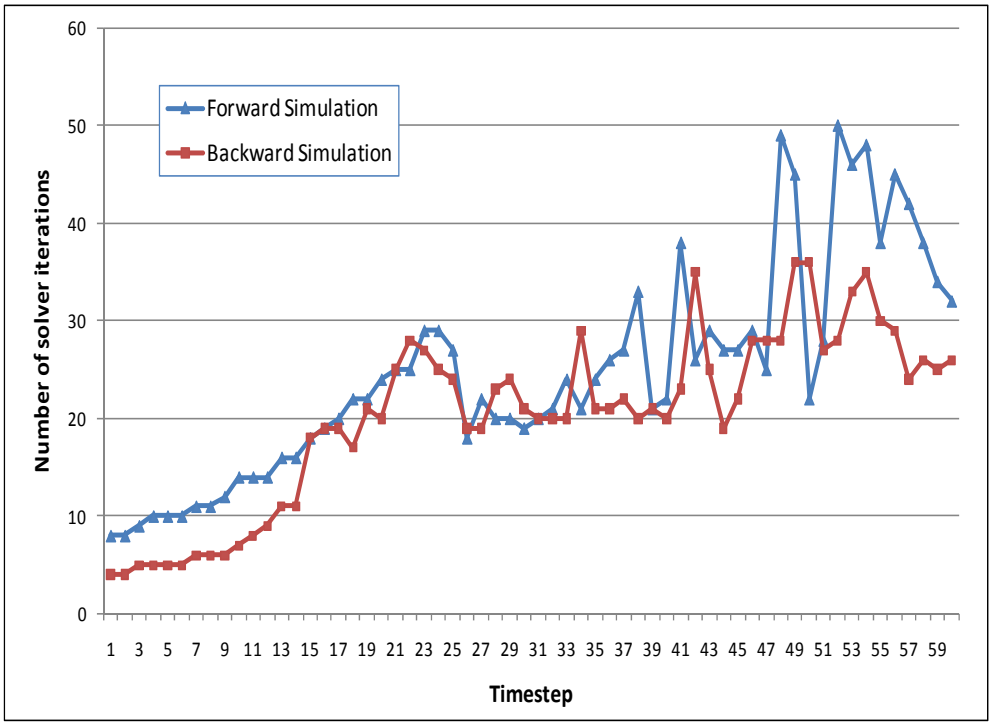

Figure 4 Number of AMG solver iterations with timestep in the forward and backward simulations for Case 3

While testing the new algorithm's performance, it has also been validated that the gradient of the objective function with respect to control variable i ($=dJ/du_i$) from the AMG solver with the CPRA preconditioner is exactly the same as that from direct solver (Figure 1). Case 1 was used for the validation because adjoint equation of the case is relatively easy for the direct solver to solve for its smaller linear system size.

One way to check the performance of a preconditioner is to check the number of iterations of a linear solver using the preconditioner to solve a linear equation. Figure 2 through Figure 4 show the number of iterations of the AMG solver using

the "original" CPR preconditioner to solve $\tilde{A}y = \tilde{b}$ at each timestep in the forward simulation for each case. The number of iterations of the AMG solver using the CPRA preconditioner to solve $(\tilde{A})^T y = d$ at each timestep in the backward simulation is also shown in the figures. Here $\tilde{A}$ is the Jacobian matrix evaluated at the convergence of Newton iteration at a timestep. The same tolerance parameter values are used to solve both $\tilde{A}y = \tilde{b}$ and $(\tilde{A})^T y = d$ with the AMG solver. Ideally, with the same tolerance, the number of iterations of the AMG solver with the CPRA preconditioner in the backward simulation should not be much different from that with the original CPR preconditioner in the forward simulation. This can be well confirmed in Figure 2 through Figure 4.

Application to a Production Optimization Problem

The gradients obtained from CHEARS is used with a sequential quadratic programming algorithm (SQP) (Rao, 2009) to solve the production optimization problem described by Equation (1). The simulation model used is that of a real reservoir, and is chosen because it is fairly large such that direct and other simple solvers cannot be used for this problem, and this model was also used by Chaudhri et al. (2009) to test ensemble based optimization. Thus these results also allow us to compare adjoint-based optimization to ensemble-based optimization. For details of ensemble-based optimization, readers are referred to Chen et al. (2009).

The reservoir field considered for this study is an off-shore Angola field which is also used for case 3 in the validation and performance of the CPRA/AMG solver section above. The average depth of the reservoir is about 7000 ft and the initial pressure is around 3300 psia at a datum depth of 7250 ft. The bubble point pressure of the reservoir is assumed to be approximately 2195 psia. Thus, the reservoir is initially undersaturated and has no gas cap. The reservoir temperature is 1300F. The field has an average porosity of about 31% and average permeability of 900 md. The oil water contact is located at a depth of 7973 ft. The simulation model has grid dimensions of 96 × 73 × 150. The size of a grid-block in the x-direction is approximately 245 ft and varies between 190 ft to 335 ft in the y-direction. The thickness of an individual layer shows a wide variation between 0.1 ft to 275 ft. There are 15 producers and 11 injectors in the simulation model. The completion interval is different for different wells. The reservoir properties (e.g. porosity and permeability) are assumed to be known without any uncertainty.

Total fluid production (reservoir volume) rate is the primary constraint on the producers whereas the injectors are constrained by the total water injection (reservoir volume) rate. The producers and the injectors are also constrained by the bottom hole pressure (BHP) limits. The producers are constrained by a minimum bottom hole pressure of 2240 psia whereas the injectors are constrained by a maximum bottom hole pressure of 4700 psia. Additionally, the producers and injectors also have group constraints of total fluid production rate and total water injection rate, respectively. The BHP constraints are simple bound constraints and are handled directly by the optimizer. The other constraints are nonlinear constraints and are handled by the simulator itself (Sarma et al., 2008). For this field, we assume that the water injection is started from day 1 and is continued throughout the life of the water flooding project (3960 days). The objective of this study is to optimize the net present value (NPV) of the reservoir over the project life. The control variables are the bottom hole pressures and the control step size for both producers and injectors is considered as 180 days which results in 22 control steps per well for the optimization problem. Therefore, for a total of 26 wells (15 producers and 11 injectors), we need to optimize a set of 572 control variables. Some of the other parameters used in the optimization procedure include oil price of $70/bbl, water disposal cost of $10/bbl, discount rate of 0% per year, and an ensemble of 40 realizations of control variables.

In order to understand the benefit of any optimization process, it is usual to compare the optimization results against a base or reference case. In the case of production optimization, such a base case would be a reasonable production strategy that an engineer would devise given a simulation model and a set of constraints. In general, although an engineer's intuition and knowledge about a reservoir plays an important role in determining optimum or near-optimum well locations, it is very difficult to understand the implications of varying well controls such as rates manually on the optimization process. It is thus usual for engineers to specify constant production/injection rates or BHPs until some detrimental reservoir response such as breakthrough is observed.

For the purpose of this case study also, the base (or reference) case is a constant production/injection rate strategy (unless other constraints are hit). The group constraint of total fluid production rate is equally distributed among all producers. Therefore, in the base case, all the producers are allowed to produce at an equal production rate throughout the water flooding project. In a similar manner, the group constraint of total fluid injection rate is equally distributed among all injectors, allowing each injector to inject water at an equal rate.

Figure 5 shows the increase in NPV (normalized with respect to the maximum NPV obtained) with iterations for SQP/Adjoint with CPRA/AMG solver (green line) and also that from two ensemble-based approaches demonstrated in Chaudhri et al. (2009). The red line is from the original ensemble-based optimization as described in Chen et al. (2009), and the blue line is from the conjugate gradient based ensemble optimization as described in Chaudhri et al. (2009). All methods lead to a significant increase in NPV (about 17%) over the base case. Further, we observe that the SQP/Adjoint reaches close to the maximum NPV after 3 iterations which requires 8 simulations. Both ensemble based methods require on the order of 600 simulations to reach the same level of NPV. This clearly proves the CPRA/AMG solver's accuracy, efficiency, and the capability to perform long term optimization for large, complex reservoir models at low computational cost. This also

demonstrates the benefit of adjoint-based optimization over ensemble optimization for large-scale models for which hundreds of simulations are currently not practically possible. Please note that a direct solver available in CHEARS was not able to solve the linear system of this problem, so more efficient reservoir simulation specific solvers are required for such problems.

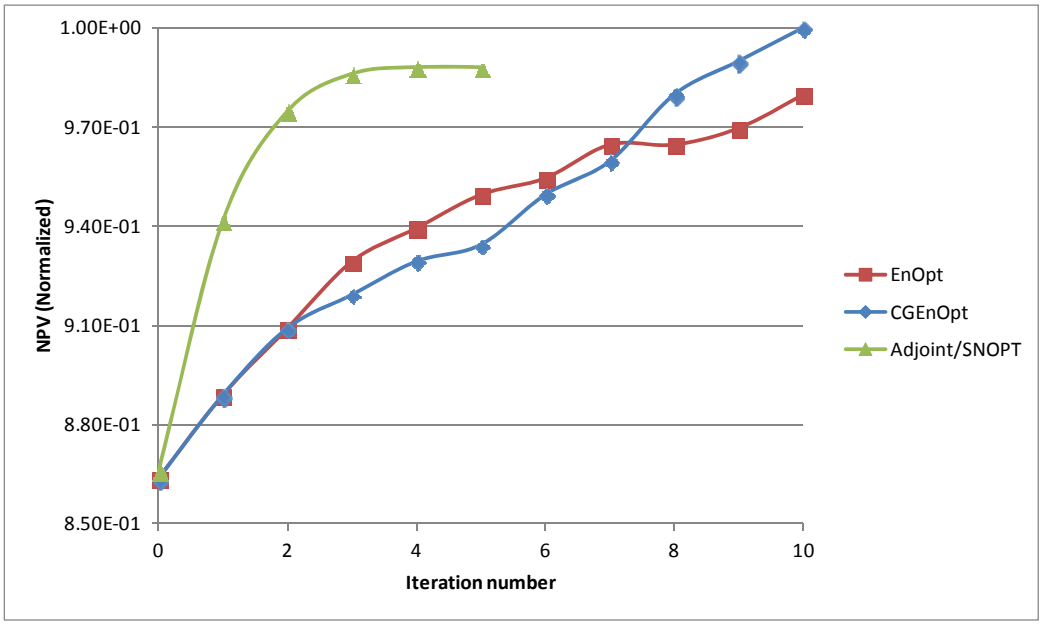

Figure 5 NPV vs. iteration number for ensemble-based optimization (from Chaudhri et al., 2009) and SQP/Adjoint

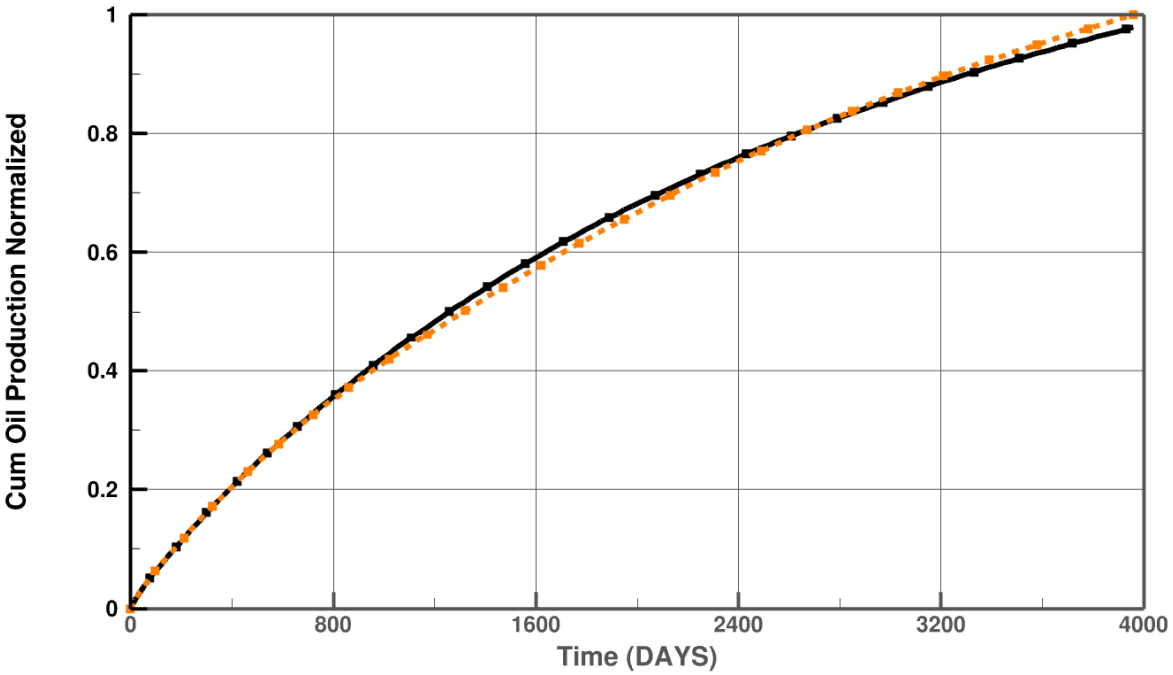

Figure 6 Cumulative oil production for base (black) and optimized case (orange)

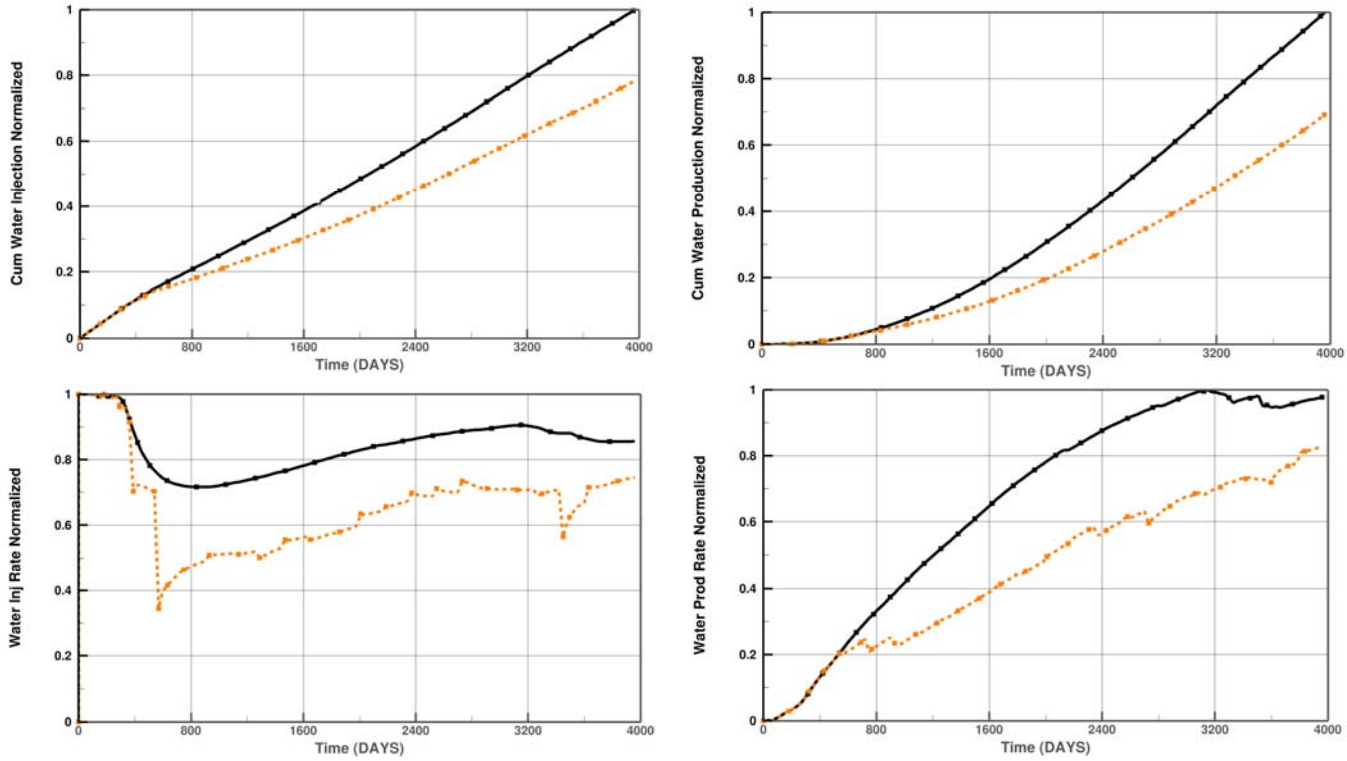

Figure 7 Water injection and production cumulative (top figures) and rates (bottom figures) for base (black) and optimized (orange) cases

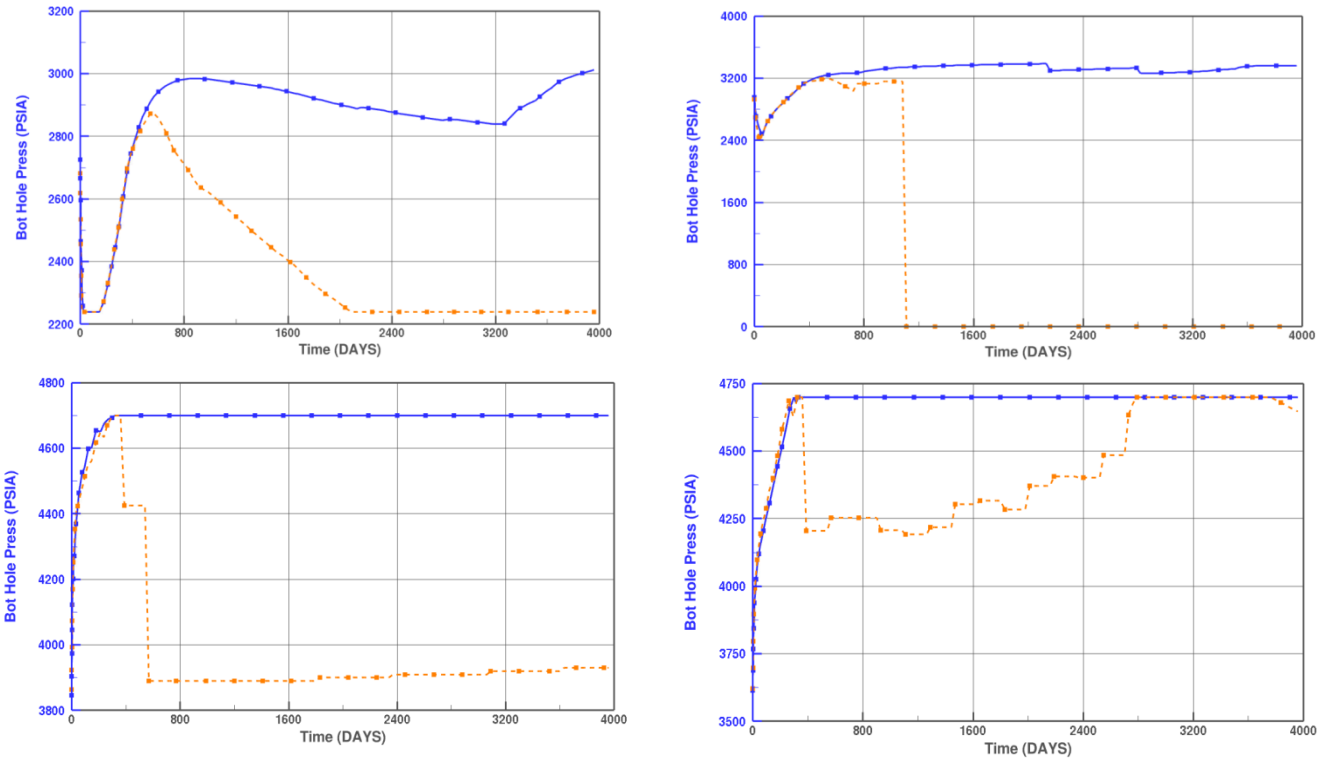

Figure 8 Typical BHP profiles for most producers for base (blue) and optimized (orange) cases (left top), some producers are shut down after optimization (top right), typical injector BHP profiles for most injectors (bottom left), some injectors however continue to inject in late time (bottom right)

Figure 6 shows cumulative oil production (normalized) for base (black) and optimized (orange) cases. We observe that overall, at the end of the optimization period, there is only a slight improvement in cumulative oil production. The significant increase in NPV thus is a result of a significant decrease in water injection and production (normalized) as seen in Figure 7. We see that for about 500 days in the beginning, the water injection and production rates for the optimized case are similar to the base case, and after that, both injection and production decrease significantly, leading to the increase in NPV.

Figure 8 shows BHP profiles for some of the injectors and producers for the base (blue) and optimized (orange) cases. The top left figure shows the BHP profiles typical of most producers. Although it is hard to determine the reason behind the nature of the optimized profiles due to interaction with various constraints, it seems that the optimization is trying to open up the producers slowly over time until minimum BHP is reached. On the other hand, some wells that produce a lot of water are shut down as shown in the top right figure. For most injectors, as seen in the lower left figure, the BHPs hit the lower bound after about 500 days, which is in agreement with the reduction in total injection. Some injectors do, however, continue injecting at the upper BHP bound throughout the optimization period as seen in the lower right figure.

Conclusion

A new two-stage preconditioner, named the CPRA preconditioner, has been developed to solve the adjoint equation with transpose of Jacobian matrix and implemented in the Chevron's in-house reservoir simulator, CHEARS. The AMG solver is used to solve the linear system with the transpose of pressure matrix, which is one of the two-stages of construction of the CPRA preconditioner. Through test cases, it has been confirmed that the CPRA/AMG solver with GMRES acceleration solves the adjoint equation very efficiently with reasonable number of linear solver iterations. Adjoint simulations to calculate the gradients with the CPRA/AMG solver take about the same time at most as corresponding CPR/AMG forward simulations. Accuracy of the solutions has also been confirmed by verifying the gradients against solutions with a direct solver. A production optimization case study for a real field using the CPRA/AMG solver has further validated its accuracy, efficiency, and the capability to perform long term optimization for a large, complex reservoir models at low computational cost. The new algorithm to create CPRA preconditioner and coupling of it with the AMG solver for efficient solution of the adjoint equation are now patent pending.

Acknowledgments

The authors would like to thank Chevron for permission to publish this work.

Nomenclature

c: non linear constraint equation
C: pressure prolongation matrix
g: reservoir simulation equation
J: cost function
J_A: augmented cost function
$\overline{\overline{J}}$ *: Jacobian matrix*
L: Lagrangian
LB: lower bound
M_{CPR}^{-1} *: CPR preconditioning matrix for matrix* $\tilde{A}$
M_{CPR}^{-T} *: CPRA preconditioning matrix for matrix* $\left(\tilde{A}\right)^T$
$\tilde{M}$ *: approximate factorization of matrix* $\tilde{A}$
$\hat{M}$ *: approximate factorization of matrix* $\left(\tilde{A}\right)^T$
n: control step
n_{cell}: number of cells
n_{eqn}: number of equations per cell
N: total number of control steps or matrix used for pressure reduction
r: residual vector
u: control vector
UB: upper bound
x: dynamic states of the system
ϕ *: function of the dynamic states of the last control step*
λ *: Lagrange multiplier*

superscript
T: transpose

subscript
p: pressure
s: unknowns excluding pressure

References

Appleyard, J.R. and Cheshire, I.M., Nested Factorization, paper SPE 12264 presented at the SPE Reservoir Simulation Symposium, San Francisco, 1983.

Brouwer, D.R. and Jansen, J.D., Dynamic Optimization of Water Flooding with Smart Wells using Optimal Control Theory, paper SPE 78278 presented at the SPE 13th European Petroleum Conference, Aberdeen, 2002.

Cao, H., Tchelepi, H.A., Wallis, J., Yardumian, H., Parallel Scalable Unstructured CPR-Type Linear Solver for Reservoir Simulation, paper SPE 96809 presented at the SPE Annual Technical Conference and Exhibition, Dallas, 2005.

Chaudhri, M., Liu, N., and Phale, H. A., An Improved Approach For Ensemble Based Production Optimization: Application To A Field Case, paper SPE 121307 presented at the SPE EUROPEC Conference, Amsterdam, 2009.

Chen, Y., Oliver, D.S., and Zhang, D., Efficient Ensemble-based Closed-loop Production Optimization, *SPE Journal*, 14(4), 634-645, 2009.

Rao, S.S., *Engineering Optimization: Theory and Practice*, John Wiley and Sons Ltd, New York, NY, 2009

Saad, Y. and Schultz, M.H., GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems, *SIAM J. SCI. STAT. COMPUT.*, 7(3), 856-869, 1986.

Sarma, P., Aziz, K., and Durlofsky, L.J., Implementation of Adjoint Solution for Optimal Control of Smart Wells, paper SPE 92864, presented at the SPE Reservoir Simulation Symposium, Houston, 2005.

Sarma, P., Chen, W.H., Durlofsky, L.J., and Aziz, K., Production Optimization with Adjoint Models under Nonlinear Control-State Path Inequality Constraints, paper SPE 99959, SPE Reservoir Evaluation & Engineering, 11(2), 326-339, 2008.

Stengel, R.F., *Optimal Control and Estimation*, Dover Books on Advanced Mathematics, New York, 1985.

Wallis, J.R., Incomplete Gaussian Elimination as a Preconditioning for Generalized Conjugate Gradient Acceleration, paper SPE 12265, presented at the SPE Reservoir Simulation Symposium, San Francisco, 1983.

Wallis, J.R., Kendall, R.P., and Little, T.E., Constrained Residual Acceleration of Conjugate Residual Methods, paper SPE 13536, presented at the SPE Reservoir Simulation Symposium, Dallas, 1985.

Wallis, J.R., Tchelepi, H.A., and Cao, H., Apparatus, Method and System for Improved Reservoir Simulation Using a Multiplicative Overlapping Schwarz Preconditioning for Adaptive Implicit Linear Systems, *U.S. Patent 7516056*, 2006