Comput Geosci (2017) 21:1385–1402
DOI 10.1007/s10596-017-9634-3

ORIGINAL PAPER

Gradient-based production optimization with simulation-based economic constraints

Oleg Volkov[1] · **Mathias C. Bellout**[2]

Received: 17 October 2016 / Accepted: 26 February 2017 / Published online: 15 March 2017

Abstract In reservoir management, production optimization is performed using gradient-based algorithms that commonly rely on an adjoint formulation to efficiently compute control gradients. Often, however, economic constraints are implicitly embedded within the optimization procedure through well performance limits enforced at each reservoir simulation time-step. These limits effectively restrict the operational capabilities of the wells, e.g., they stop or shut down production depending on a predetermined profitability threshold for the well. Various studies indicate that the accuracy of the gradient and, by consequence, the performance of the optimization algorithm suffer from this type of heuristic constraint enforcement. In this paper, an analytical framework is developed to study the effects of enforcing simulator-based economic constraints when performing gradient-based production optimization that relies on derivatives obtained through an adjoint formulation. The framework attributes the loss in control gradient sensitivity to non-differentiable unscheduled changes in the well model equations. The discontinuous nature of these changes leads to inconsistencies within the adjoint gradient formulation. These inconsistencies, in turn, reduce gradient quality and subsequently decrease algorithmic performance. Based on the developed framework, we devise an efficient simulator-based mode of constraint enforcement that yields gradients with fewer consistency errors. In this implementation, the well model equations that violate constraints are removed from the governing system right after the violation occurs and are not reinserted until the next well status update. The constraint enforcement modes are further coupled with a strategy that improves the selection of initial controls for subsequent iterations of the optimization procedure. After a given simulation, the resulting combination of open and shut-in periods generates a status update schedule, or shut-in history. The shut-in history of the current optimal solution is saved and used in subsequent optimization iterations to make the status update a part of the optimal solution. The novel simulation-based constraint implementation, with and without shut-in history, is applied to two production optimization cases where, for a large set of initial guesses, and different model realizations, it retains and improves the performance of the search procedure compared to when using common modes of economic constraint enforcement during production optimization.

Keywords Simulation-based economic constraints · Reservoir simulation · Production optimization · Adjoint gradients · Reservoir management

✉ Oleg Volkov
ovolkov@stanford.edu

Mathias C. Bellout
mathias.bellout@ntnu.no

[1] Department of Energy Resources Engineering, Stanford University, Stanford, CA 94305, USA

[2] Petroleum Cybernetics Group, Department of Geoscience and Petroleum, NTNU, Trondheim, Norway

1 Introduction

Optimization procedures based on mathematical programming techniques serve as important decision-support tools to select optimal petroleum field development concepts and reservoir management strategies. In reservoir management, decision-making workflows rely on gradient-based production optimization procedures to find improved production

strategies, e.g., optimal well control configurations that increase recovery from waterflooding. Commonly, these procedures depend on reservoir simulations to measure cost function performance, and on an adjoint formulation to efficiently compute cost function derivatives with respect to well controls.

However, the production optimization procedure is often additionally loaded with sets of heuristic constraints used to represent the project's economic interface [9]. These rules are imposed to meet necessary field development profitability criteria and are commonly enforced during reservoir simulation. Often, these rules are implemented as well and field production thresholds based on economic parameters enforced at each simulation time-step. Thus, since reservoir simulations are an integral part of gradient-based production optimization, and this type of economic constraints is commonly embedded within these simulations, there is a need for an analytical framework that studies the effects of enforcing these constraints on the overall optimization procedure in general, and on the computation of adjoint-based gradients, in particular. Furthermore, such a framework will help develop specialized strategies to maintain adjoint-gradient accuracy in spite of the heuristic enforcement of simulation-based constraints, and thus, to mitigate likely decreases in algorithmic performance.

From an operational point of view, the enforcement of economic constraints during reservoir simulation serves as an additional control tool to stimulate production and mimic real field changes to production settings. These constraints are commonly enforced within the time-step domain of the reservoir simulator, independent of the production optimization procedure, where well control types, e.g., bottom hole pressure (BHP) and/or phase rates, are frequently used as variables defined over significantly larger control periods. The main function of economic constraint enforcement is to react to uneconomic production scenarios by performing real-time, i.e., in time-step scale, adjustments to the wells. For example, in the case of waterflooding, operating conditions at which production is unprofitable can be determined, and accordingly, a lower limit for the oil production rate or an upper limit on the water cut (fraction of water in the total liquid production) can be enforced based on these settings. For either limit, a standard type of enforcement is to shut the well reactively once the profitability setting is violated during forward simulation.

Similarly, commingled production is often managed by an analog set of rules that operates on either groups of wells, on the entire field, or both. These field and group constraints come in addition to the constraints operating individually on each well. This work, however, only deals with simulator-based constraints that operate on individual wells and enforce restrictions based on economic performance, though the analysis may be extended to include field constraints and other types of performance measures, in future work.

1.1 Economic constraints in adjoint-based production optimization

Adjoint-based gradients have become an indispensable tool for closed-loop reservoir management [19], integrated field development problems [1], multiobjective analysis [13], robust production optimization [4], and for many other multi-phase flow optimization problems (see review in [10]). Adjoint gradients provide fast, inexpensive, yet accurate linear approximations of the objective function and non-linear constraints. These linear approximations are used by non-linear programming algorithms such as sequential quadratic programming (SQP) [8] or methods of moving asymptotes (MMA) [20] to build non-linear approximations that ensure a fast convergence to a local optimal solution. In the context of production optimization, the SQP and MMA algorithms were first applied in [19] and [2], respectively.

In reference to gradient-based methods, few studies have addressed the impact that simulator-based constraints have on the optimization process. In [4], robust long-term optimization results obtained using reactive control were compared to results obtained using constraints handled within a dedicated gradient-projection trust-region method. The reactive control strategy in that work entails the enforcement of simulator-based constraints. However, the strategy presented is performed in a stand-alone manner, i.e., not within, nor coupled to, an optimization procedure. In [4], the authors demonstrated that a gradient-based production optimization procedure outperformed the reactive control strategy. Furthermore, the work in [12] presented results supporting the claim that production optimization using BHP controls with rate constraints implemented at the simulator level yields higher final solution values, than a formal production optimization setup where rate constraints are handled by an SQP algorithm.

The applicability of simulator-based constraints when performing gradient-based production optimization has been assessed at depth in [5] and [14]. These papers underline the difficulties that arise during gradient-based optimization when a particular well is shut-in, or when it changes its control type, in the middle of a control period due to the violation of a constraint. They report a decrease in performance and attribute it to a loss in sensitivity. The authors claim that sensitivity is lost when the gradient term corresponding to a particular time-step vanishes due to the effects of the constraint violation. The sensitivity loss, in turn, limits the capacity of the optimization algorithm to change the corresponding control variable. The authors propose a couple of strategies for implementing simulator-based constraints to remedy the loss in

sensitivity. In [5], the production optimization uses rate control targets. This allows the authors to implement a procedure that avoids shutting-in the wells completely, but rather keeps them running at an insignificantly small rate to avoid losing sensitivity. In [14], a proactive gradient-based method is proposed which treats the constrained quantities as controllable parameters. In this approach, the optimization procedure involves progressively activating constraints, with each new constraint resulting in a significant improvement in the objective.

In this paper, we argue that the missing gradient terms and subsequent loss of sensitivity are fundamentally related to non-differentiable changes in the simulator's flow equations. Thus, the subsequent decrease in performance is mainly attributed to the non-differentiable unscheduled changes in the well model equations that occur when simulator-based constraints are enforced. These discontinuities introduce inconsistencies in the adjoint formulation which in turn yields inaccurate gradients that reduce the performance of the optimization algorithm. In this work, we develop an analytical framework to assess both the validity and performance of the adjoint gradient formulation when used for gradient-based production optimization with economic constraints. The framework is used to describe typical ways of enforcing these types of constraint and to derive an additional, more efficient mode of enforcement. Two cases are used to compare typical and the novel mode of constraint enforcement during production optimization. Results from this comparison study demonstrate a significant improvement in the convergence of the optimization procedure for a large set of initial guesses and geological realizations.

This paper is organized as follows. Section 2 lays the theoretical background for this work by presenting the general production optimization problem as well as the adjoint-based gradient formulation. In Section 3, we conduct a rigorous analysis of the adjoint-based gradient formulation in which we take into account the non-differentiable changes in the well model equations. Section 4 discusses the conventional ways economic constraints are implemented within reservoir simulation, while Section 5 presents a novel way of dealing with this type of constraints when performing production optimization. Finally, we substantiate the conclusions derived in the theoretical part by conducting numerical experiments on two production optimization cases presented in Section 6. The results of the optimizations are presented and discussed in Section 7, while Section 8 presents some conclusions and recommendations.

2 Background

This section introduces the general production optimization problem and lays the theoretical background for the analysis of the adjoint-based gradient formulation to be performed in the following section. It discusses the importance of continuous differentiability of the objective function when performing gradient-based optimization, and presents the steps involved in the derivation of the adjoint system and gradient. In this and the following sections, we refer to the common presentation of the adjoint formulation in the petroleum literature [3, 19] as the *standard adjoint formulation*.

2.1 General production optimization problem

The production optimization problem is given as

$$\begin{aligned} \text{find maximum of} \quad & J(\mathbf{x}, \mathbf{u}), && \text{(1a)}\\ \text{subject to} \quad & \mathbf{g}_{\text{reserv}}(\mathbf{x}, \mathbf{u}) = 0, && \text{(1b)}\\ & \mathbf{g}_{\text{wells}}(\mathbf{x}, \mathbf{u}) = 0, && \text{(1c)}\\ & x_0 \text{ given}, && \text{(1d)}\\ & \mathbf{u} \in D. && \text{(1e)} \end{aligned}$$

Here, $\mathbf{x}$ are the variables determining the state of the system discretized both in space and time, while x_0 defines the part of the vector $\mathbf{x}$ which corresponds to the initial state $x_0 \overset{\text{def}}{=} \mathbf{x}|_{t=0}$. The vector $\mathbf{u}$ designates the model parameters used as optimization variables, while D is a feasibility region of those variables. $\mathbf{g}_{\text{reserv}}(\mathbf{x}, \mathbf{u})$ and $\mathbf{g}_{\text{wells}}(\mathbf{x}, \mathbf{u})$ are, respectively, the discretized in space and time reservoir and well governing equations for the unknown state variables $\mathbf{x}$. To be precise, in Eq. 1c, we only include equations that are implicit with respect to the well unknowns. Thus, any well equation that can be written in explicit form is satisfied by simple elimination. For example, in the case of a well BHP control, we have

$$\begin{aligned} &\mathbf{g}_{\text{reserv}}(\mathbf{x}, p_{\text{BHP}}) = 0\\ &\underbrace{\mathbf{g}_{\text{wells}} \overset{\text{def}}{=} p_{\text{BHP}} - p_{\text{BHP}}^{\text{target}} = 0}\\ &\qquad\Downarrow\\ &\mathbf{g}_{\text{reserv}}(\mathbf{x}, p_{\text{BHP}}^{\text{target}}) = 0. \end{aligned} \tag{2}$$

The objective function $J(\mathbf{x}, \mathbf{u})$ is the net present value (NPV) given as

$$J(\mathbf{x}, \mathbf{u}) \overset{\text{def}}{=} \sum_{n=1}^{N} \left(\sum_{i=1}^{n_w} \sum_{p=1}^{n_p} C_{p,i}(t_n)\, q_{p,i}(\mathbf{x}, \mathbf{u}) \right) \Delta t_n, \tag{3}$$

where t_n is discretized time steps with time step sizes $\Delta t_n = t_n - t_{n-1}$. $C_{p,i}$ and $q_{p,i}$ denote, respectively, the discounted price and the production/injection phase flow rate of phase p in the i-th well. n_w refers to the total number of wells while n_p is the total number of phases present in the system.

In the form (1a), the objective function depends on both the state variables $\mathbf{x}$ and the optimization variables $\mathbf{u}$. Furthermore, the optimization problem in Eq. 1 includes the various equality constraints given in Eq. 1b and c. The number of equality constraints is the same as the number of state variables.

Typically, due to a large total number of variables, it is not practical to have the optimization algorithm solve a problem that includes both state and control variables. A significant advantage of the adjoint method is that it uses the equality constraints in Eq. 1b to calculate the variation of the objective function solely with respect to the well controls $\mathbf{u}$. Thus, the adjoint formulation acts in the capacity of the dependence $\mathbf{x} = \mathbf{x}(\mathbf{u})$.

2.2 Objective function differentiability and adjoint gradient formulation

Continuous differentiability of the objective function is essential for the performance of the operating non-linear programming algorithm. This statement is illustrated with the following simplified argument. The mathematical theory of gradients relies heavily on the Riesz representation theorem [18]. This theorem implies that if an objective function $J(\mathbf{u})$ is Fréchet-differentiable with respect to the optimization parameters $\mathbf{u}$, and then, there exists a unique vector ∇J, called the gradient of J, so that

$$\lim_{\varepsilon \to 0} \frac{J(\mathbf{u} + \varepsilon\, \mathbf{u}') - J(\mathbf{u})}{\varepsilon} = \langle \nabla J, \mathbf{u}' \rangle \tag{4}$$

is valid for all admissible perturbations $\mathbf{u}'$. If J is not continuously differentiable, then ∇J may not exist or is non-unique. In both cases, the objective function approximations based on ∇J are inexact, which in turn reduces the performance of the operating non-linear programming method.

In the derivation of the adjoint gradients, we follow the standard procedure of formulating an augmented Lagrangian functional L and then deriving the first optimality conditions based on this functional [15]. For the optimization problem (1), the Lagrangian L has the form

$$L \stackrel{\text{def}}{=} J + \lambda^T \mathbf{g}_{\text{reserv}}(\mathbf{x}, \mathbf{u}) + \mu^T \mathbf{g}_{\text{wells}}(\mathbf{x}, \mathbf{u}). \tag{5}$$

Here, λ and μ are the vectors of Lagrange multipliers with size equal to the number of state equations. By construction, L has the same critical points as J. Therefore, the local optima of J are the same as the local optima of L, located either at a critical point, or on the boundary of the feasibility region D. By definition, the critical point is a vector containing the elements $\{\lambda, \mu, \mathbf{x}, \mathbf{u}\}$ for which the partial derivatives of L with respect to λ, μ, $\mathbf{x}$, and $\mathbf{u}$ are zero, i.e.,

$$\frac{\partial L}{\partial \lambda} = 0 \Rightarrow \text{(1b)}, \tag{6a}$$

$$\frac{\partial L}{\partial \mu} = 0 \Rightarrow \text{(1c)}, \tag{6b}$$

$$\frac{\partial L}{\partial \mathbf{x}} = 0 \Rightarrow \frac{\partial}{\partial \mathbf{x}} \Big(\sum_{n=1}^{N} \sum_{i=1}^{n_w} \sum_{p=1}^{n_p} C_{p,i}(t_n)\, q_{p,i}\, \Delta t_n + \lambda^T \mathbf{g}_{\text{reserv}} + \mu^T \mathbf{g}_{\text{wells}} \Big) = 0, \tag{6c}$$

$$\frac{\partial L}{\partial \mathbf{u}} = 0 \Rightarrow \frac{\partial}{\partial \mathbf{u}} \Big(\sum_{n=1}^{N} \sum_{i=1}^{n_w} \sum_{p=1}^{n_p} C_{p,i}(t_n)\, q_{p,i}\, \Delta t_n + \lambda^T \mathbf{g}_{\text{reserv}} + \mu^T \mathbf{g}_{\text{wells}} \Big) = 0. \tag{6d}$$

As indicated here, Eq. 6a and b are the governing equations satisfied with a state variable solution $\mathbf{x}$ and prescribed $\mathbf{u}$. We next consider Eq. 6c and d to derive the standard formulation of the adjoint system.

2.3 Standard adjoint formulation

Suppose that $q_{p,i}$, $\mathbf{g}_{\text{reserv}}$, and $\mathbf{g}_{\text{wells}}$ are continuously differentiable with respect to $\mathbf{x}$ and $\mathbf{u}$. The Eq. 6c and d can be reformulated in the following simplified form:

$$\sum_{n=1}^{N} \sum_{i=1}^{n_w} \sum_{p=1}^{n_p} C_{p,i}(t_n) \frac{\partial q_{p,i}}{\partial \mathbf{x}} \Delta t_n + \lambda^T \frac{\partial \mathbf{g}_{\text{reserv}}}{\partial \mathbf{x}} + \mu^T \frac{\partial \mathbf{g}_{\text{wells}}}{\partial \mathbf{x}} = 0, \tag{7a}$$

$$\sum_{n=1}^{N} \sum_{i=1}^{n_w} \sum_{p=1}^{n_p} C_{p,i}(t_n) \frac{\partial q_{p,i}}{\partial \mathbf{u}} \Delta t_n + \lambda^T \frac{\partial \mathbf{g}_{\text{reserv}}}{\partial \mathbf{u}} + \mu^T \frac{\partial \mathbf{g}_{\text{wells}}}{\partial \mathbf{u}} = 0. \tag{7b}$$

The equations in Eq. 7a constitute a system that can be solved to obtain the unknown multipliers λ and μ. The solution of this system yields what are called adjoint variables. The equations in Eq. 7b are not solved directly since these equations define a critical point with no guarantee of optimality. Instead, we derive the adjoint-based gradient for the Eq. 4 by using the left-hand-side of Eq. 7b as follows:

$$\nabla J \stackrel{\text{def}}{=} \sum_{n=1}^{N} \sum_{i=1}^{n_w} \sum_{p=1}^{n_p} C_{p,i}(t_n) \frac{\partial q_{p,i}}{\partial \mathbf{u}} \Delta t_n + \lambda^T \frac{\partial \mathbf{g}_{\text{reserv}}}{\partial \mathbf{u}} + \mu^T \frac{\partial \mathbf{g}_{\text{wells}}}{\partial \mathbf{u}}. \tag{8}$$

Further details regarding the derivation of the adjoint-based gradient can be found in [19].

The next section carries out a rigorous mathematical treatment of the discontinuities imposed on the adjoint system when enforcing simulation-based economic constraints. The aim of this analysis is to investigate the mechanism and conditions under which non-differentiable changes, both within production outputs $q_{p,i}$ and/or model equations $\mathbf{g}_{\text{reserv}}$ and $\mathbf{g}_{\text{wells}}$ in Eq. 8, can render the derived adjoint-based gradient inconsistent with respect to the underlying optimization problem.

3 Theoretical analysis of simulator-based constraint enforcement

We first consider the general case in which the expressions $q_{p,i}$, $\mathbf{g}_{\text{reserv}}$, and $\mathbf{g}_{\text{wells}}$ undergo discontinuous changes in time. Because the partial derivatives in Eqs. 7 and 8 are applied to a system already discretized in time, the perturbations of $\mathbf{x}$ and $\mathbf{u}$, which trigger $q_{p,i}$, $\mathbf{g}_{\text{reserv}}$, and $\mathbf{g}_{\text{wells}}$ to change form, are not taken into account in either of these expressions. This is a major reason for why the standard adjoint formulation is inconsistent whenever discontinuous changes occur in the model equations

To study the effects of these inconsistencies, we consider a continuous-in-time (non-discretized) adjoint formulation [3]. This approach, commonly referred to as *optimize-then-discretize*, is less frequently used for gradient-based production optimization since the *discretize-then-optimize* approach is often much simpler to implement within a reservoir simulator. A detailed comparison of continuous-in-time and discrete-in-time adjoint formulations is performed in [12]. In that paper, the authors demonstrate that the adjoint equations resulting from the discretization of a continuous-in-time formulation coincide with those of the discrete-in-time formulation for all time-steps except the last one. This result enables us to apply the continuous-in-time form in the following analysis, though, in accordance with the result, we restrain the formulation to the time interval $0 \leqslant t \leqslant \overline{T}$, where $\overline{T} \stackrel{\text{def}}{=} t_{N-1}$. The continuous-in-time form is preferred because it allows us to use differential calculus to study the effects of enforcing the economic constraints on the adjoint system.

3.1 Derivation of continuous-in-time adjoint formulation

To be clear, here and in the next section, we employ the convention that $\mathbf{g}_{\text{reserv}}$, $\mathbf{g}_{\text{wells}}$, λ, μ, and $\mathbf{x}$ are all discrete-in-space functions, which means they are defined for all t such that $0 \leqslant t \leqslant \overline{T}$. The continuous-in-time form of Eqs. 6c and 6d is as follows:

$$\frac{\partial}{\partial \mathbf{x}}\Big(\int_0^{\overline{T}} \sum_{i=1}^{n_w}\sum_{p=1}^{n_p} C_{p,i}(t)\, q_{p,i} + \lambda^T \mathbf{g}_{\text{reserv}} + \mu^T \mathbf{g}_{\text{wells}}\, dt\Big) = 0, \tag{9a}$$

$$\frac{\partial}{\partial \mathbf{u}}\Big(\int_0^{\overline{T}} \sum_{i=1}^{n_w}\sum_{p=1}^{n_p} C_{p,i}(t)\, q_{p,i} + \lambda^T \mathbf{g}_{\text{reserv}} + \mu^T \mathbf{g}_{\text{wells}}\, dt\Big) = 0. \tag{9b}$$

In [3, 12], the authors apply the Leibniz integral rule to move the partial derivative operator inside the integral sign. However, since the integrand terms in Eq. 9a and b are not necessarily continuous and continuously differentiable for all t, $\mathbf{x}$, and $\mathbf{u}$, we cannot rely on this rule for our general problem. Instead, to perform the interchange between the differentiation and integration operators, in this work, we rely on a set of theorems that offer less restrictive conditions than those imposed by the Leibniz integral rule.

Two particular theorems are relevant for the treatment of Eq. 9a and b. The first theorem [17] provides a sufficient (though not necessary) condition for the operator interchange while using a relaxed requirement of continuity for all t:

Theorem 1 *Let $f(x,t)$ be a real-valued function defined on $X \times \mathbb{R}$, where X is an open interval in $\mathbb{R}$, and assume that for each fixed $x \in X$,*

(i) *$t \mapsto f(x,t)$ is integrable,*
(ii) *$\partial f/\partial x$ exists for almost all t,*
(iii) *there exists an integrable function g, independent of x, such that*

$$\left|\frac{\partial}{\partial x} f(x,t)\right| \leqslant g(t) \quad \text{for almost all } t \text{ and for all } x \in X.$$

Then,

$$\frac{\partial}{\partial x}\int_{\mathbb{R}} f(x,t)\, dt = \int_{\mathbb{R}} \frac{\partial}{\partial x} f(x,t)\, dt.$$

The second theorem (proved in [21]) provides necessary and sufficient conditions that do not require the continuity of f for all $\mathbf{x}$ and $\mathbf{u}$. The theorem uses the *generalized absolutely continuous function in the restricted sense* (ACG) definition for f. This definition is applicable to the functions with piecewise continuous integrands in Eq. 9a and b.

Theorem 2 *Let $f(x,t) : X \times \mathbb{R} \mapsto \mathbb{R}$. Suppose that $f(x,t)$ is ACG on X for almost all t in $\mathbb{R}$. Then,*

$$\frac{\partial}{\partial x}\int_{\mathbb{R}} f(x,t)\,dt = \int_{\mathbb{R}} \frac{\partial}{\partial x} f(x,t)\,dt$$

for almost all t in $\mathbb{R}$, if and only if

$$\int_{x_1}^{x_2}\int_{\mathbb{R}} \frac{\partial}{\partial x} f(x,t)\,dt\,dx = \int_{\mathbb{R}}\int_{x_1}^{x_2} \frac{\partial}{\partial x} f(x,t)\,dx\,dt \tag{10}$$

for all $[x_1, x_2] \in X$.

In the following section, we use Theorems 1 and 2 to properly take into account the discontinuities within the adjoint derivation.

3.2 Adjoint formulation subject to discontinuity

In this work, we argue that changes in time fundamentally alter the modeling equations as well as the production profiles in the adjoint gradient expression. These changes manifest themselves as discontinuities in the integrand terms $\lambda^T \mathbf{g}_{\text{reserv}}$, $\mu^T \mathbf{g}_{\text{wells}}$, or $q_{p,i}$, respectively. Discontinuities within the integrand terms appear when the forward simulation reaches either a particular constant point in time $t = \tau$ or when a prescribed condition $c(\mathbf{x},\mathbf{u}) < 0$ is violated. In the following, we analyze the occurrence of these discontinuities within the derivation of the adjoint gradient.

To simplify the analysis, we represent either of these integrand terms using the generic symbol f. We start by describing the simple type of discontinuity that occurs when the simulation reaches a point $t = \tau$ (e.g., the start of a new control period). The expression of f resulting from this type of discontinuity is written as follows:

$$f(\mathbf{x},\mathbf{u}) \overset{\text{def}}{=} \begin{cases} f_1(\mathbf{x},\mathbf{u}) & t < \tau, \\ f_2(\mathbf{x},\mathbf{u}) & t \geqslant \tau, \end{cases} \tag{11}$$

where f_1 and f_2 are continuously differentiable functions of $\mathbf{x}$ and $\mathbf{u}$. It can easily be verified that the conditions of Theorem 1 are satisfied for this type of discontinuity. Thus, the standard adjoint formulation can be applied to this case. Furthermore, by extension, the standard formulation can be applied when having a countable number of discontinuities of this type (each defined using a different time constant τ), which is the result of the piecewise linear parametrization commonly used for production optimization problems.

We now consider the case when a discontinuity in f is triggered by the violation of a condition of type $c(\mathbf{x},\mathbf{u}) < 0$. In this case, we use Theorem 2 to derive the necessary and sufficient conditions to perform the required operator interchange in Eq. 9a and b. Theorem 2 offers less stringent requirements than those imposed by Theorem 1, which allows us to treat the expressions and their derivatives as generalized functions. We define the time at which a constraint violation occurs using the Dirac delta function $\delta(\cdot)$ as

$$\tau(\mathbf{x},\mathbf{u}) = \int_0^{\overline{T}} \delta\big(c(\mathbf{x},\mathbf{u})\big)\, t\, dt. \tag{12}$$

The f that results once a violation $c(\mathbf{x},\mathbf{u}) < 0$ occurs is given as

$$f(\mathbf{x},\mathbf{u}) \overset{\text{def}}{=} \begin{cases} f_1(\mathbf{x},\mathbf{u}) & t < \tau(\mathbf{x},\mathbf{u}), \\ f_2(\mathbf{x},\mathbf{u}) & t \geqslant \tau(\mathbf{x},\mathbf{u}), \end{cases} \tag{13}$$

where again f_1 and f_2 are continuously differentiable with respect to $\mathbf{x}$ and $\mathbf{u}$. For the sake of compactness, we rewrite Eq. 13 using the Heaviside step function $H(\cdot)$. Finally, we take the derivative with respect to $\mathbf{x}$ to obtain

$$\frac{\partial f}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}}\Big(f_1 + H\big(t-\tau\big)\big(f_2 - f_1\big)\Big). \tag{14}$$

Substituting Eq. 14 into the right-hand side of Eq. 10 (Theorem 2) and applying the fundamental theorem of calculus yields

$$\int_0^{\overline{T}}\int_{\mathbf{x}_1}^{\mathbf{x}_2} \frac{\partial f}{\partial \mathbf{x}}\, d\mathbf{x}\, dt = \int_0^{\overline{T}} \Big[f_1 + H\big(t-\tau\big)\big(f_2 - f_1\big)\Big]_{\mathbf{x}_1}^{\mathbf{x}_2} dt, \tag{15}$$

while substituting Eq. 14 into the left-hand side of Eq. 10 and applying the chain rule gives

$$\int_{\mathbf{x}_1}^{\mathbf{x}_2}\int_0^{\overline{T}} \frac{\partial f}{\partial \mathbf{x}}\, dt\, d\mathbf{x} = \int_{\mathbf{x}_1}^{\mathbf{x}_2}\int_0^{\overline{T}} \frac{\partial f_1}{\partial \mathbf{x}} + H\big(t-\tau\big)\frac{\partial (f_2 - f_1)}{\partial \mathbf{x}} - \delta\big(t-\tau\big)\frac{\partial \tau(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}\big(f_2 - f_1\big)\, dt\, d\mathbf{x}. \tag{16}$$

Since f_1 and f_2 are differentiable, the first two expressions in the right-hand side of Eq. 16 can be simplified by interchanging the integrals with respect to t and $\mathbf{x}$ as follows:

$$\begin{aligned} &\int_{\mathbf{x}_1}^{\mathbf{x}_2}\int_0^{\overline{T}} \frac{\partial f_1}{\partial \mathbf{x}} + H\big(t-\tau\big)\frac{\partial (f_2 - f_1)}{\partial \mathbf{x}}\, dt\, d\mathbf{x} = \\ &\int_{\mathbf{x}_1}^{\mathbf{x}_2}\int_0^{\tau(\mathbf{x},\mathbf{u})} \frac{\partial f_1}{\partial \mathbf{x}}\, dt\, d\mathbf{x} + \int_{\mathbf{x}_1}^{\mathbf{x}_2}\int_{\tau(\mathbf{x},\mathbf{u})}^{\overline{T}} \frac{\partial f_2}{\partial \mathbf{x}}\, dt\, d\mathbf{x} = \\ &\int_0^{\overline{T}}\int_{\mathbf{x}_1}^{\chi(t,\mathbf{u})} \frac{\partial f_1}{\partial \mathbf{x}}\, d\mathbf{x}\, dt + \int_0^{\overline{T}}\int_{\chi(t,\mathbf{u})}^{\mathbf{x}_2} \frac{\partial f_2}{\partial \mathbf{x}}\, d\mathbf{x}\, dt = \\ &\int_0^{\overline{T}} \Big[f_1 + H\big(t-\tau\big)\big(f_2 - f_1\big)\Big]_{\mathbf{x}_1}^{\mathbf{x}_2} dt, \end{aligned} \tag{17}$$

where $\chi(t, \mathbf{u})$ is the inverse function of $\tau(\mathbf{x}, \mathbf{u})$ with respect to $\mathbf{x}$, i.e., $\tau(\chi(t, \mathbf{u}), \mathbf{u}) \equiv t$. Whereas the third term on the right-hand side of Eq. 16 is reduced to

$$\int_{\mathbf{x}_1}^{\mathbf{x}_2} \int_{0}^{\overline{T}} \delta(t-\tau) \frac{\partial \tau(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} (f_2 - f_1)\, dt\, d\mathbf{x} = \int_{\mathbf{x}_1}^{\mathbf{x}_2} \left(\frac{\partial \tau(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} (f_2 - f_1) \right)\Big|_{t=\tau(\mathbf{x},\mathbf{u})} d\mathbf{x}. \tag{18}$$

After comparing Eqs. 15 and 17, we conclude that Eqs. 15 and 16 can be equal to each other only if Eq. 18 vanishes. Therefore, a crucial claim is that the necessary and sufficient condition (10) from Theorem 2 holds when either

$$\frac{\partial \tau}{\partial \mathbf{x}} = 0 \qquad \text{or} \qquad f_2 = f_1. \tag{19}$$

This statement remains valid in the presence of a countable number of discontinuities of the same type. Moreover, the same analysis holds for the derivatives with respect to $\mathbf{u}$.

3.3 Application of theorems to discrete-in-time reservoir system

Hereafter, without loss of generality, we consider $\mathbf{g}_{\text{reserv}}$, $\mathbf{g}_{\text{wells}}$, λ, μ, and $\mathbf{x}$ to be discrete-in-space and discrete-in-time functions. If the first option of Eq. 19 is satisfied, i.e., if τ is independent of both $\mathbf{x}$ and $\mathbf{u}$, then the situation is equal to case analyzed earlier using Theorem 1. On the other hand, if c depends on either or both $\mathbf{x}$ and $\mathbf{u}$, then one can find an interval $[\mathbf{x}_1,\ \mathbf{x}_2]$ and a perturbation $\mathbf{x}'$ such that

$$\frac{\partial \tau(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \overset{\text{def}}{=} \lim_{\varepsilon \to 0} \frac{\tau(\mathbf{x} + \varepsilon\, \mathbf{x}', \mathbf{u}) - \tau(\mathbf{x}, \mathbf{u})}{\varepsilon} \tag{20}$$

is non-zero for $\mathbf{x} \in [\mathbf{x}_1,\ \mathbf{x}_2]$.

We now consider the second option of Eq. 19. At this point, a brief comment is necessary regarding the nature of the integrand terms $q_{p,i}$, $\lambda^T \mathbf{g}_{\text{reserv}}$, and $\mu^T \mathbf{g}_{\text{wells}}$: the reservoir-governing equations $\mathbf{g}_{\text{reserv}}$ are based on conservation principles, and are, both by construction and in a physical sense, continuous with respect to $\mathbf{x}$ and $\mathbf{u}$. Wells, on the other hand, may undergo changes in time which can fundamentally modify both their modeling equations $\mathbf{g}_{\text{wells}}$ and their production profiles $q_{p,i}$. The result is that Eqs. 15 and 16 can always be set equal to each other, and thus, the necessary condition (10) can be met for $\lambda^T \mathbf{g}_{\text{reserv}}$, but not for $q_{p,i}$ or $\mu^T \mathbf{g}_{\text{wells}}$.

Thus, focusing on the $q_{p,i}$ and $\mu^T \mathbf{g}_{\text{wells}}$ terms, below, we list two conclusions for the validity of the standard adjoint formulation when discontinuities are imposed during gradient computation. These conclusions vary depending on whether the discontinuities are predetermined, as is the case for the common constant piecewise-linear well control parametrization, or whether the discontinuities stem from economic constraint enforcement. (This is a non-exhaustive list since discontinuities stemming from other sources, e.g., other types of constraints, may also influence the adjoint system, but this type of situation is beyond the scope of this analysis.)

Conclusion 1 If the well equations $\mathbf{g}_{\text{wells}}$ are modified by a discontinuous change at a predetermined time, then Theorem 1 allows us to interchange the differentiation and integration operators in Eq. 9a and b. In this case, the standard adjoint formulation is valid when computing the gradient.

Conclusion 2 If the discontinuity in the well equations $\mathbf{g}_{\text{wells}}$ is precipitated by the violation of the well performance constraint $c(\mathbf{x}, \mathbf{u}) < 0$, then, according to Theorem 2, the operator interchange cannot be applied. In this case, the standard adjoint formulation is inconsistent due to errors occurring when taking the partial derivatives of both $q_{p,i}$ and $\mathbf{g}_{\text{wells}}$.

Having an analytic framework with sufficient generality to both support Conclusion 1 and 2 is an important contribution from this work. This allows us to describe both the type of control changes that occur within the default control configuration for production optimization, consisting of predetermined control changes, as well as the additional changes that occur whenever simulator-based constraints are violated. As mentioned, it is often the case that this latter type of economic well performance limits is implemented within simulators during production optimization. Finally, using this framework, we can assess the validity and performance of the adjoint gradient formulation under these various control and constraint scenarios.

In the next section, the analytical framework is used to describe common modes of economic constraint enforcement and their consequences. Thereafter, the framework is used to derive an additional *mode of enforcement* with the goal of maintaining adjoint gradient accuracy in spite of the discontinuities created by well performance limits.

4 Conventional modes of economic constraint enforcement

First, we provide several definitions necessary to describe changes in well model equations during simulation. We then categorize the various types of control changes as being either *scheduled* or *unscheduled changes*. These distinctions will be useful to describe both traditional and new constraint-handling approaches, as well as to present optimization cases and analyze results, in subsequent sections. The description of traditional modes of economic constraint

enforcement is provided at the end of this section, while the new constraint-handling approach is described in the next section.

4.1 Definitions to describe well control changes and patterns

We continue to use the terminology developed previously, e.g., the terms $\mathbf{g}_{\text{wells},1}$ and $\mathbf{g}_{\text{wells},2}$, to describe several examples of well model changes that often occur during a typical reservoir simulation whenever these constraints are active. We divide the domain of possible changes into two categories: *scheduled* and *unscheduled changes*.

Scheduled changes This category consists of changes conditioned only to time, e.g., control updates predetermined by a production schedule. In reservoir simulation, scheduled updates cause the well equation to change in four possible combinations:

1. in the case of rate type of control before and after the control update, the well equation changes from

   $$\mathbf{g}_{\text{wells},1} \stackrel{\text{def}}{=} q_{\cdot,i} - q_{\cdot,i}^{\text{target},1} \quad \text{to} \quad \mathbf{g}_{\text{wells},2} \stackrel{\text{def}}{=} q_{\cdot,i} - q_{\cdot,i}^{\text{target},2};$$

2. in the case of rate type of control before and BHP type of control after the control update, the well equation changes from

   $$\mathbf{g}_{\text{wells},1} \stackrel{\text{def}}{=} q_{\cdot,i} - q_{\cdot,i}^{\text{target},1}$$

   to non-existing well equation $\mathbf{g}_{\text{wells},2}$;

3. in the case of BHP type of control changing to rate type of control, the well equation before the change, $\mathbf{g}_{\text{wells},1}$, is non-existing, and after the update becomes

   $$\mathbf{g}_{\text{wells},2} \stackrel{\text{def}}{=} q_{\cdot,i} - q_{\cdot,i}^{\text{target},2};$$

4. finally, for completeness, in the case of a BHP controlled well remaining in BHP type of control mode after the update, the change is between two non-existing well equations $\mathbf{g}_{\text{wells},1}$ and $\mathbf{g}_{\text{wells},2}$.

For any of these scheduled changes, the time τ is constant, and the standard adjoint formulation is valid according to Conclusion 1.

Unscheduled changes This category consists of changes triggered when the setting of the default, i.e., the currently active, well control type violates a specified constraint. The triggering of unscheduled changes may stem from various types of constraint enforcement. One such type, for example, is the enforcement of an upper or lower bound that limits the value of an associated control type, e.g., (*target*)-rate bounds when the well is controlled by BHP setting, and vice versa. Other triggers of such changes may be an economic limit on production, or even the condition that a producer (injector) must have positive (negative) well rates during simulation.

Well control vs. well status updates Finally, we distinguish between two categories of well model updates: a *well control update* and a *well status update*. Well control updates occur at predetermined intervals called control steps or periods, i.e., they are scheduled events during simulation. The well settings prescribed at these updates correspond to the well control optimization variables $\mathbf{u}$. Well status updates, on the other hand, occur both at each control step and in addition, may occur at any time step during simulation. For example, an unscheduled well status update occurs (and is stored) at the time-step an economic limit is enforced.

Thus, defined, well control and well status updates are two categories that are non-exclusive, in that well control updates are a subset of well status updates. These categories will be useful to explain the results from the new economic constraint enforcement algorithm introduced below. Also, they will later be useful to describe additional strategies that use shut-in history to improve the performance of the optimization procedure. Next, we describe two conventional approaches for how to enforce unscheduled changes.

4.2 Traditional modes of economic constraint enforcement

In production optimization, economic constraints may be implemented as non-linear constraints within the optimization algorithm itself (*indirect enforcement*), or as simple checks at each simulation time-step that test whether the constraints are honored or not (*direct enforcement*). These two ways of enforcing constraints are discussed below.

Indirect enforcement In this type of enforcement, the simulation itself is performed without economic limits. However, the defined economic limits are imposed as non-linear constraints within the optimization algorithm which requires the constraints to be honored for all well control values. Details regarding this particular type of enforcement and the procedure of its implementation can be found in [23].

Direct enforcement In this type of enforcement, the simulation includes an automatic shut-in algorithm mimicking the workover action of the reactive control. Below, we distinguish two modes of this algorithm implemented in commercial reservoir simulators [6] referred by the commonly used nomenclature “SHUT” and “STOP”. It should be noted that in this discussion, we only have BHP as the

active well control, and that no associated (target) rate constraints are specified. Cases involving rate control as the active control type will be treated in further work by the authors.

STOP. In this mode, the well is connected to the reservoir throughout the entire simulation. *While solving* Eq. 1b and c, the relevant well production profiles, $c(\mathbf{x}, \mathbf{u})$, are verified for violation of the economic limits *at each time step*.

In fact, this control mode corresponds to the reactive control strategy, i.e., once a violation is detected on a well, the governing equation for that well is replaced by a shut-in well equation. The shut-in equation is implemented, for example, by setting the total fluid volume rate equal to zero (making $\mathbf{g}_{\text{wells},2}$ non-trivial). Under this mode of enforcement, the standard adjoint formulation is non-valid due to errors when taking the partial derivatives of both the $q_{p,i}$ and $\mathbf{g}_{\text{wells}}$ terms (see Conclusion 2).

SHUT. Similar to the previous mode, in this mode, the well is checked *at each time step* and the well is shut-in once a violation occurs. However, in this mode, the shut-in procedure is accompanied by totally removing the well equation from the reservoir system until the simulation finishes. Thus, for the remaining of the simulation, after the control update is implemented, $\mathbf{g}_{\text{wells},2}$ is non-existent, i.e., there is no governing equation for the well. Moreover, $\mathbf{g}_{\text{wells},1}$ is also non-existent since the well is originally controlled by BHP (see the problem definition in Eq. 1 and the associated derivation in Eq. 2). Obviously, non-existing $\mathbf{g}_{\text{wells}}$ is not taken into account in Eqs. 11–18. Thus, whenever $\mathbf{g}_{\text{wells}}$ is removed from the system, whether the standard adjoint formulation is consistent or not depends only on the partial derivatives of $q_{p,i}$.

Based on this discussion, the SHUT mode, when compared to the STOP mode, is shown to contain less errors that lead to inconsistencies in the standard adjoint formulation. However, the SHUT mode lacks the capacity of the reactive control to reopen the well if the economic constraints are satisfied at later simulation times. This observation leads us to develop a novel mode of constraint enforcement which combines the advantages of the two conventional modes.

5 HALT—a novel mode of constraint enforcement

The nomenclature "HALT" and its associated algorithm refer to a new, direct mode of enforcement proposed in this paper. This mode operates similarly to SHUT, except that the isolation of the well from the reservoir does not last until the simulation ends, but only for a small period of time. The actual duration of the shut-in period depends on when the next *well model update* occurs. Below, we describe the concept of the shut-in pattern which determines all well model updates during simulation. Furthermore, we use the concept of the shut-in pattern to provide a detailed description of the HALT algorithm.

5.1 Shut-in pattern

The number and timing of well model updates is defined by the update schedule of the well. This schedule is determined before simulation, and it is either set from the default well configuration, or from well status updates that have been stored from a preceding simulation run. In this work, we refer to the collection of all well status updates during a simulation as the *shut-in pattern* for that simulation. Mathematically, the shut-in pattern is defined with two sets of values: $\{\theta_1, \ldots\}$, which represent the timing of the updates, and $\{\pi_1, \ldots\}$, which are the status of the wells after the updates. With the definition of a shut-in pattern, we can offer a concise description of the HALT mode of enforcement: in the HALT mode, the shut-in period lasts until the first subsequent point in the shut-in pattern. Algorithm 1 describes the manner in which the shut-in pattern is built and modified during simulation. It also shows in exact terms how the HALT mode differs from the SHUT mode.

Algorithm 1 Simulation in SHUT and HALT modes

```
t ← 0
while t < T do
  for each well do
    if well status = shut then
      remove associated g_wells from Eq. 1c
    else
      include associated g_wells into Eq. 1c
    end If
  end for
  calculate x(t + Δt) solution of Eq. 1b and c
  for each open well do
    if c(x(t + Δt), u) ⩾ 0 then
      find k such that θ_{k−1} < t + Δt ⩽ θ_k
      {θ_k, θ_{k+1}, θ_{k+2}, ...} ← {t + Δt, θ_k, θ_{k+1}, ...}
      if HALT then
        {π_k, π_{k+1}, π_{k+2}, ...} ← {shut, open, π_{k+1}, ...}
      else if SHUT then
        {π_k, π_{k+1}, π_{k+2}, ...} ← {shut, shut, shut, ...}
      end if
    end if
  end for
  t ← t + Δt
end while
```

Here, we discuss two ways the HALT method serves as an efficient approach to deal with wells when performing production optimization subject to simulator-based economic constraints. First, once economic constraints are enforced, the well model equations are removed from the system of equations, thus eliminating error terms in the adjoint gradient calculation (as explained by Conclusion 2). This helps preserve gradient accuracy even though wells are shut-in at arbitrary points within control updates, and maintaining gradient quality assures algorithm performance is not diminished. Second, using the HALT approach in combination with the active use of a current best shut-in pattern (properly defined below), we can further mitigate the effects of economic constraint violation by taking into account solution history. Though well status updates originate from shut-ins caused by previous constraint enforcement, these status update points are also used by HALT to check whether a shut-in well may be reactivated. By having HALT verify wells at any subsequent status update point, we are, in effect, adding extra possible reactivation points in time spans where wells are likely to have been shut-in. This yields improved solutions by constantly reactivating wells within critical time periods. In the following, we describe how shut-in history can be used to improve the convergence of the optimization algorithm.

5.2 Production optimization using shut-in history

In the following, we define two strategies for choosing initial control schedules to enhance the optimization process when using the HALT and SHUT constraint enforcement modes.

Algorithm 2 Optimization with direct enforcement

$\{\theta_1, \ldots\} \leftarrow$ well control update timing
repeat
  $\{\pi_1, \ldots\} \leftarrow$ default well configuration
  simulate with direct enforcement (e.g., **Algorithm 1**)
  calculate the objective function
  calculate the adjoint gradient
  perform an optimization iteration of SQP
until termination criteria are satisfied to given tolerances

During production optimization, multiple reservoir simulations are run sequentially while searching for optimal well controls. In the sequence of simulations launched by the optimization process, the resulting shut-in pattern from one simulation can be used as the initial control schedule in the next simulation. Algorithm 2 shows the common way of performing production optimization with direct constraint enforcement. In this configuration, the shut-in pattern is not preserved from one iteration to another.

Algorithm 3 Optimization with shut-in history

$\{\theta_1, \ldots\} \leftarrow$ well control update timing
best shut-in pattern $\leftarrow$ default well configuration
repeat
  $\{\pi_1, \ldots\} \leftarrow$ best shut-in pattern
  simulate in HALT or SHUT mode (**Algorithm 1**)
  calculate the objective function
  if current best **then**
    best shut-in pattern $\leftarrow \{\pi_1, \ldots\}$
  end if
  calculate the adjoint gradient
  perform an optimization iteration of SQP
until termination criteria are satisfied to given tolerances

In this work, we propose an alternative algorithm (Algorithm 3) that takes explicit advantage of the known shut-in pattern history. This algorithm is hereby defined as the shut-in history strategy. The goal of this strategy is to use the current best shut-in pattern to improve the performance of the next iteration of the optimization procedure. As previously mentioned, the default configuration for well control optimization entails a set of piecewise constant-in-time intervals defined over the production horizon. When we use the shut-in history strategy, we add to the default control updates that stored unscheduled well status updates from the best shut-in pattern so far. This means that when the shut-in history strategy is enabled, the status of a well may be updated at any subinterval of the control updates.

This superposition of control points lets the control vector operate within a solution space with additional degrees of freedom (ultimately determined by the total number of time steps). The thinking behind this strategy is that the shut-in pattern contributes with a heuristic expansion of the default control space based on previous "good" solutions (i.e., solutions that satisfy the imposed constraints), and that this will contribute positively to the overall search performance of the optimization algorithm.

We thus define the two strategies for choosing initial control schedules:

> **Without shut-in history**. Each simulation in the optimization sequence is performed using the default shut-in pattern, i.e., the default control configuration consisting of control updates only (see Algorithm 2).
> **With the shut-in history**. Each simulation in the sequence is performed with the shut-in pattern which so far yields the best objective function value (see Algorithm 3).

In the following section, the different control modes in combination with these strategies are tested for two cases.

6 Case studies

We perform experimental studies on two production cases. The first case is a well-known waterflooding model with synthetic geology and simplified physical properties. This model was first introduced in [22] and has on several occasions been used to benchmark optimization algorithms [11]. The second case is the Brugge benchmark field designed for the SPE Applied Technology Workshop held in Brugge in June 2008. The description of the field as well as complete results of the workshop are summarized in [16].

Production optimization is performed on both cases while enforcing two types of economic constraints. The first constraint restricts production wells from above with respect to their water cut using a limit of 95% (i.e., a workover is activated if the water cut of a producer reaches this limit). Similarly, injector wells are constrained from below with respect to their rate with a limit of 6.28×10^{-3} bbl/day in the first case and a more realistic limit of 62.8 bbl/day in the second case. Choosing a small lower bound of the injection rate in the first case allows us to concentrate on the errors introduced by the economic limits of the producers. These constraints are summarized in Table 1. In both studies, the NPV is computed using water production and injection costs equal to \$3/bbl, while the oil price is set to \$50/bbl. The discount rate is 10%. Reservoir model configurations for the two study cases, as well as simulation and optimization parameters for the experimental runs, are given below.

6.1 Case study 1

The reservoir for the first case has dimensions $60 \times 60 \times 7$ and includes eight vertical water injection wells and four vertical producers completed through all seven layers. This model is supplied with hundred and one realizations of the permeability of which one is designated as true. In this study, we employ the "true" permeability which is shown in Fig. 1 and the randomly selected realizations 18, 40, and 69.

The base case, which is also the default initial guess for the optimization, uses BHP control for both producers and injectors, without any constraints on the associated rates. However, economic constraints do apply for both types of wells (see above). All producers are set with an initial BHP of 259 bars, while all injectors start with a BHP of 261 bars. The physical properties of both fluid and rock for this case are summarized in Table 2. For the optimization of case study 1, we apply a piecewise-linear constant production setup consisting of ten control steps of 180 days each.

Table 1 Economic constraints for cases 1 and 2

Value	Description
95%	Water cut limit (cases 1 and 2)
6.28×10^{-3} bbl/day	Injector rate limit (case 1)
62.8 bbl/day	Injector rate limit (case 2)

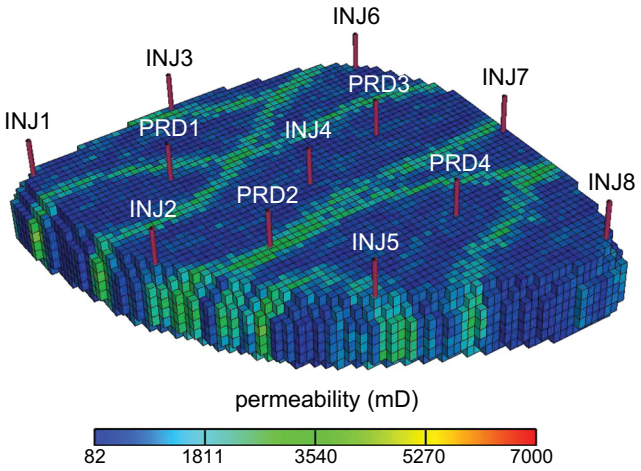

Fig. 1 Model permeability in x- and y-directions

6.2 Case study 2

In the second test case, we use the original dead-oil model and geometry of realization #73 of the stock of the Brugge reservoir model. This realization includes the original corner point grid structure, distribution of active grid blocks, depths, and volumes; NTG, porosity, and permeability distribution; as well as initial pressures and saturations. The grid is shown in Fig. 2. Case study 2 applies the same 30 standard wells: 10 injectors and 20 producers, as the original base case. Also for initial well controls, we use the original base case values. Further details regarding the Brugge model can be found in [16]. The production optimization uses 21 piecewise-linear constant controls with steps of 180 days each.

Table 2 Parameter values for case model 1

Value	Description
20%	Porosity
10^{-5} 1/bar	Rock compressibility
10^{-5} 1/bar	Fluid compressibility
1000 kg/m^3	Fluid density
1cP	Fluid viscosity
0.1	Initial water saturation
0	Rel. perm. endpoints
2	Rel. perm. exponents

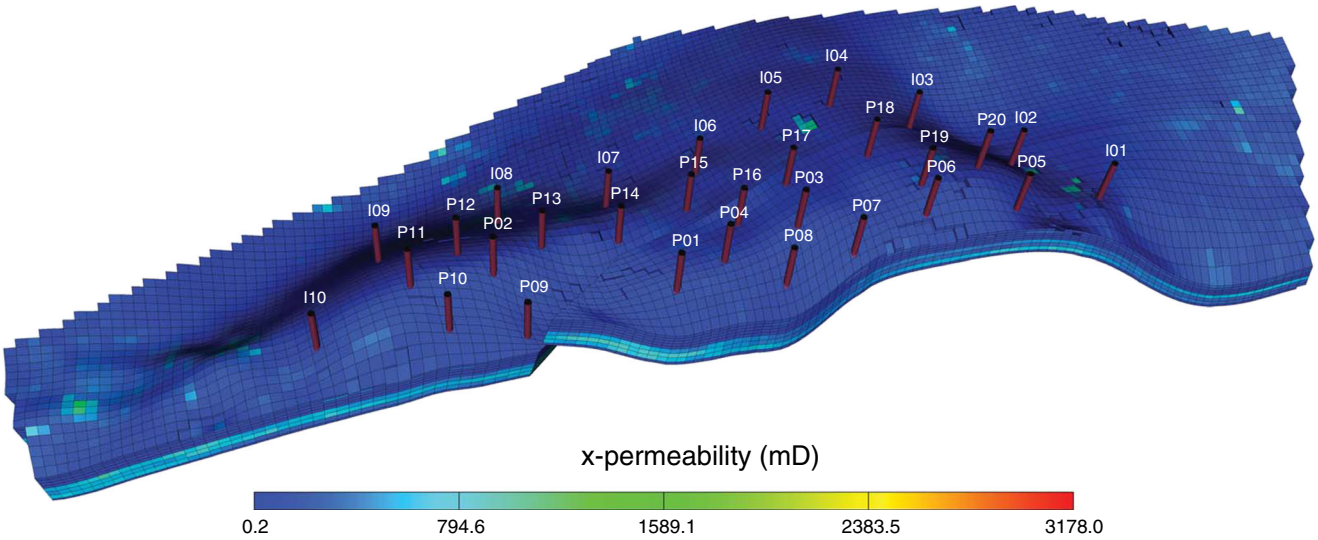

Fig. 2 Model permeability in x-directions

6.3 Simulation and optimization parameters

For simulation, we use the Automatic Differentiation General Purpose Research Simulator (AD-GPRS). This simulator is built on top of the Automatic Differentiation Expression Templates Library in [24, 25]. Since ADGPRS uses adaptive time stepping, the optimization algorithm may meet some inconsistencies if different time stepping schemes are applied for the various runs performed during optimization [23]. To mitigate this bias, the discrete-in-time NPV is evaluated using the ten-point composite Gaussian quadrature formula instead of the commonly used rectangular rule. Further details of this procedure can be found in [23].

We use a non-linear programming algorithm based on the SQP method implemented by the Sparse Nonlinear OPTimizer library (SNOPT) [7]. For both cases, the optimization process is set to terminate once the Karush–Kuhn–Tucker conditions are satisfied to a given tolerance [8]. The SNOPT parameters and termination criteria used for this case are given in Table 3.

Table 3 SNOPT configuration

Value	Description
10^{-7}	Major feasibility tolerance
10^{-4}	Major optimality tolerance
10^{-7}	Minor feasibility tolerance
200	Major iteration limit
20	Major step limit
20	Minor iterations limit

Notice that despite its strengths, the SQP algorithm is a local search algorithm. Therefore, to avoid the bias of possibly getting caught in a poor local solution, in this paper, we perform multiple production optimization runs that each starts from different initial guesses. Details on how these initial guesses are chosen are provided in Section 7.5.

In Sections 7.1–7.4, we use case study 1 with the "true" permeability to test the performance of the suggested modes of enforcement. Furthermore, in Section 7.5, we demonstrate the robustness of the methodology by optimizing over case study 2 and multiple realizations from case study 1.

7 Results

7.1 Results for case study 1

Figure 3 shows the optimal control updates (ten control steps) using BHP as initial control and the HALT, SHUT, STOP, and constrained optimization modes of constraint handling. Notice the wells are primarily active during the first five control steps (i.e., up to 900 days). After the fifth control step, the majority of the wells reach their economic limit which signals the end of the economic life of the reservoir. Before this, all solution profiles, except the SHUT mode (Fig. 3b, e), show a significant departure from the initial guess, indicating a considerable exploration of the range defined by the economic limits. On the other hand, as shown in Fig. 3, when optimizing using the SHUT control mode, the optimal solution remains close to the initial controls, which for this case correspond to relatively low initial rates. For this reason, a majority of the wells the optimal solution obtained from the SHUT control mode are not shut-in.

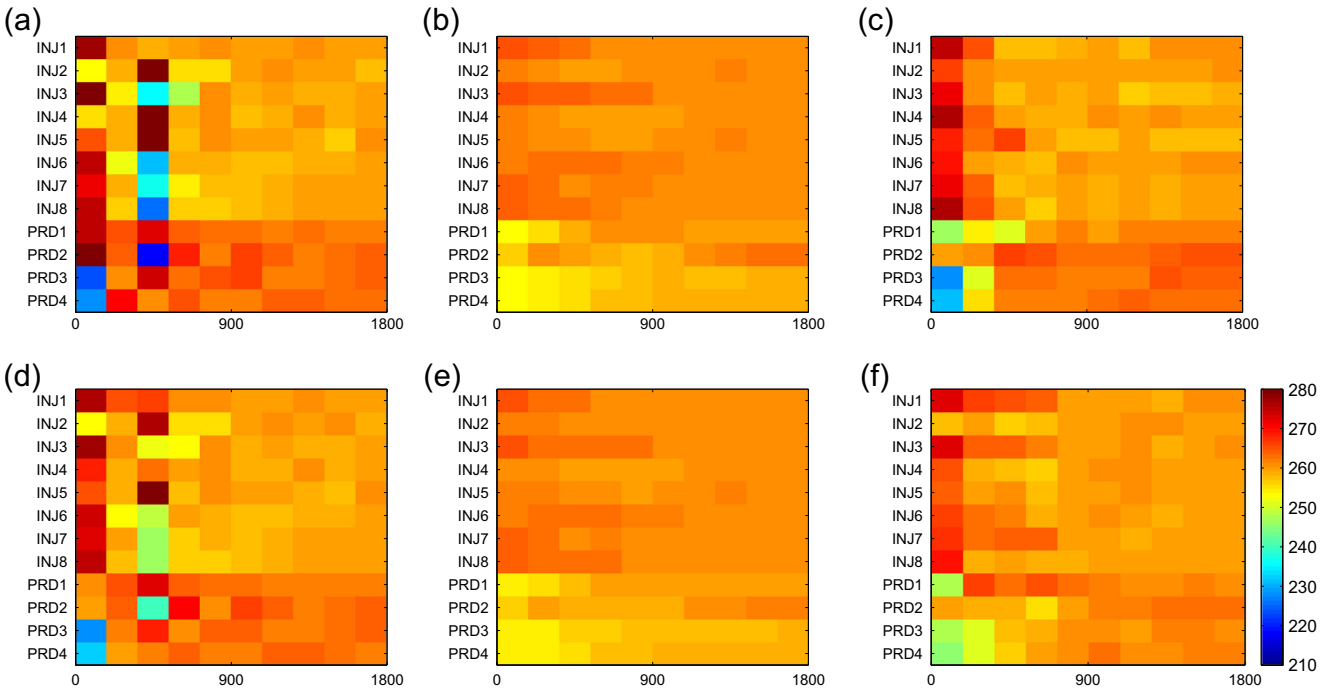

Fig. 3 Optimal solutions of well BHP controls for the modes HALT (**a**, **d**), SHUT (**b**, **e**), STOP (**c**), constrained optimization (**f**), and strategies with shut-in history (**a**, **b**) and without shut-in history (**d**, **e**)

The dissimilarities between the different solutions in terms of well status (open, stopped, or shut) can be explored by comparing the distribution of well statuses in Fig. 4.

Figure 4 presents the simulation time intervals when the wells have status as either open (white), stopped (gray), or shut (black). The distribution of statuses is consistent with

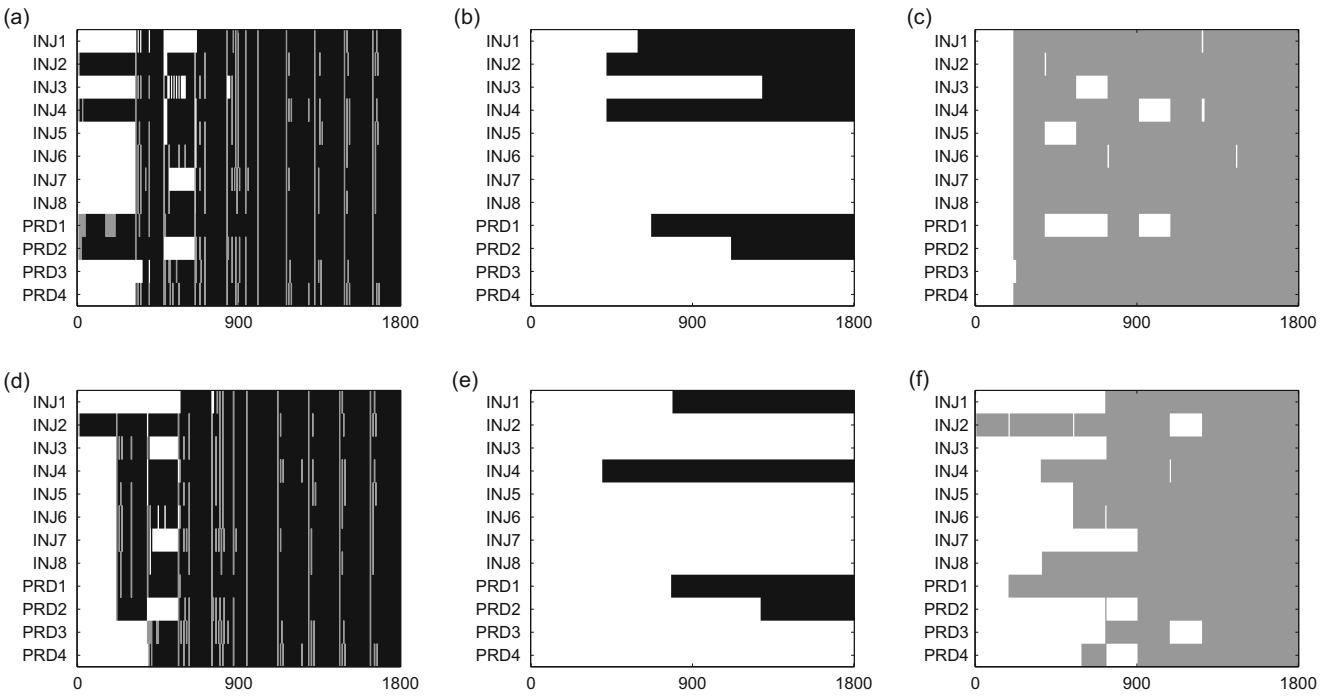

Fig. 4 Status of the wells in the optimal solutions for four enforcement modes: HALT (**a**, **d**), SHUT (**b**,**e**), STOP (**c**), constrained optimization (**f**), and two shut-in strategies: with shut-in history (**a**, **b**) and without shut-in history (**d**, **e**). The *white*, *gray*, and *black* designate respectively the open, stopped, and shut well status

the definition of modes. In the case of the HALT mode (Fig. 4a, d), the shut-in (black) intervals are predominant. Recall that in this mode, the wells are reactivated and verified for violation of the economic limits at every well status update (included predetermined control updates), which means verification is also guided by the currently implemented shut-in pattern history. To reiterate, at the verification time step, if the production performance satisfies the economic limit (e.g., the water cut), then the well is reopened; otherwise, the well is shut for all subsequent time steps until a new verification is performed at the next well status update. In Fig. 4a, d, the verification time steps appear as either gray or white stripes occurring at each status update. The SHUT mode, on the other hand, lacks the capacity to reopen the well. Therefore, the solutions from the SHUT mode given in Fig. 4b, e show only wells that go from open to shut, and that stay shut for the remaining simulation time.

Though the resulting shut-in patterns with and without shut-in history are somewhat similar within the same modes, they yield significantly different objective function values, as shown in Fig. 5. Figure 5 shows the convergence of the NPV as a function of the number of objective function evaluations for the different modes of enforcement, with and without shut-in history. The optimal NPV values for each of these runs are given in Table 4.

To simplify further analysis, we define two NPV threshold values based on the curves in Fig. 5. Each threshold value corresponds to a major bifurcation between two or more curves. The first threshold point, $\widetilde{\text{NPV}}_{\text{econ}} \simeq$ \$162 MM, is the bifurcation point where the curves for the SHUT mode with and without shut-in history, and the curve for constrained optimization, each progress along different paths. The second threshold point, $\widetilde{\text{NPV}}_{\text{stop}} \simeq$ \$174 MM, corresponds to the bifurcation point where the curves for the HALT mode with and without shut-in history, and the curve corresponding to STOP, separate after having

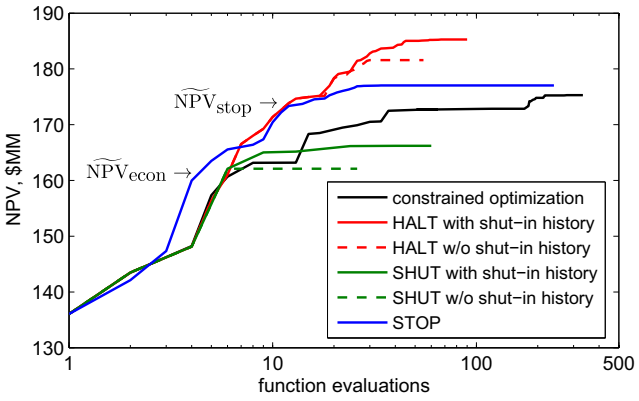

Fig. 5 Convergence of the NPV with respect to the number of objective function evaluations

Table 4 Optimal results

NPV, \$MM	Description
185.269	HALT with shut-in history
181.578	HALT without shut-in history
177.047	STOP
175.299	Constrained optimization
166.208	SHUT with shut-in history
162.095	SHUT without shut-in history

progressed along similar paths for a certain number of function evaluations.

The rate of convergence for all NPV curves before $\widetilde{\text{NPV}}_{\text{econ}}$ is similar for all modes of enforcement, except for the STOP mode (blue curve in Fig. 5). Recall that, when in STOP mode, the testing for whether the economic constraint is satisfied or not occurs at each time-step while solving Eq. 1b and c. Thus, the verification process in STOP is itself part of the search for a viable solution of the non-linear reservoir system. The difference with the other modes is that in STOP, the limits are enforced at an earlier point in the solution procedure, compared to, for example, the SHUT and HALT modes, where the verification process occurs only after the end of the verification time-steps. This observation is confirmed by comparing the results in Fig. 4a, c, d, i.e., where we observe that, in general, the stopped statuses of the wells occur earlier in Fig. 4c compared to when the shut statuses occur in Fig. 4a, d.

For the solutions with NPV greater than $\widetilde{\text{NPV}}_{\text{econ}}$, the economic limit violations are already frequent enough to affect the rate of convergence towards the optimal solution. More specifically, it is clear that the search direction in the optimization variable space is less affected by the economic limits in the HALT and STOP modes than those in the other modes. The worst effect of the economic limits is observed in the case of the SHUT mode, which has a restrictive policy of shutting-in the wells completely until simulation finishes. The result of this constraint-handling technique on the control space is directly related to the argument made in Section 5.2 about how the introduction of additional control points serves as a heuristic way to refine the solution space. In an opposite manner, by shutting-in wells indiscriminately with respect to objective function topology, and leaving them shut for the remainder of the simulation, the SHUT mode is effectively reducing the solution space in a way that, because it is not guided by derivative information of any sort, it is likely to reduce convergence ability and possibly close-off optimal solution areas for the optimization algorithm. These conclusions are supported by the results depicted in Fig. 3b, e, where an apparent fettered search leads to control values that do not depart significantly from the values given in the initial guess.

In the next four sections, we discuss different aspects and implications regarding the optimization results presented so far. First, we analyze how the open/stopped/shut status of the well affects the convergence of the optimization procedure. Next, we discuss whether actively using the well shut-in history is beneficial for the production optimization process. We then compare the direct enforcement strategies against the strategy of constrained optimization. Finally, we study the sensitivity of the results with respect to different initial guesses.

7.2 Comparison of direct modes of enforcement

For STOP and HALT modes, the convergence of the optimization procedure is similar until the NPV reaches $\widetilde{\text{NPV}}_{\text{stop}}$. However, after reaching the $\widetilde{\text{NPV}}_{\text{stop}}$ threshold, the STOP mode ceases to improve. We claim that this is related to the fact that the number of errors that lead to the inconsistency of the standard adjoint formulation is higher in the STOP mode than that in the HALT mode (see explanation in Section 4.2), and not to the usage of the well shut-in history. This claim is supported by examining the results from the HALT and STOP modes that are independent of the shut-in history. Recall that in both the HALT without shut-in history, and the STOP modes, the history of well shut-ins is not retained between separate function evaluations. Examining the results from these two modes, we see that, beyond the $\widetilde{\text{NPV}}_{\text{stop}}$ threshold, the HALT without shut-in history maintains a high rate of convergence compared to STOP, and that it finishes with a better optimal solution (see Table 4). This points to the importance of maintaining gradient accuracy to retain convergence competence.

Finally, we mention that the overall optimization runtime depends not only on the number of function evaluations but also on the runtime of the simulations needed to perform those function evaluations. Using as reference the runtime of a simulation without direct enforcement, $\mathcal{T}$, a simulation using HALT has an average runtime equal to $0.85\,\mathcal{T}$, while a simulation using SHUT takes on average $0.6\,\mathcal{T}$, and a simulation using STOP takes on average $1.2\,\mathcal{T}$. The STOP mode algorithm is, therefore, the most expensive in terms of runtime. This result is supported by the fact that in the STOP mode algorithm the simulator solves additional well equations.

7.3 Role of the shut-in pattern during optimization

Recall that the typical SQP algorithm is set to terminate as soon as one of two criteria is satisfied: either the norm of the gradient falls below a given tolerance, or a maximum number of iterations is reached. Also, note that when computing the gradient norm, we exclude the controls that have reached the boundary of the feasibility region. Using the gradient-based criterion assumes that the gradient is sufficiently accurate which may not be the case here. For this reason, in addition to the two criteria mentioned above, we also terminate the SQP when the update step of the optimization variables reaches the order of machine epsilon. Which of the

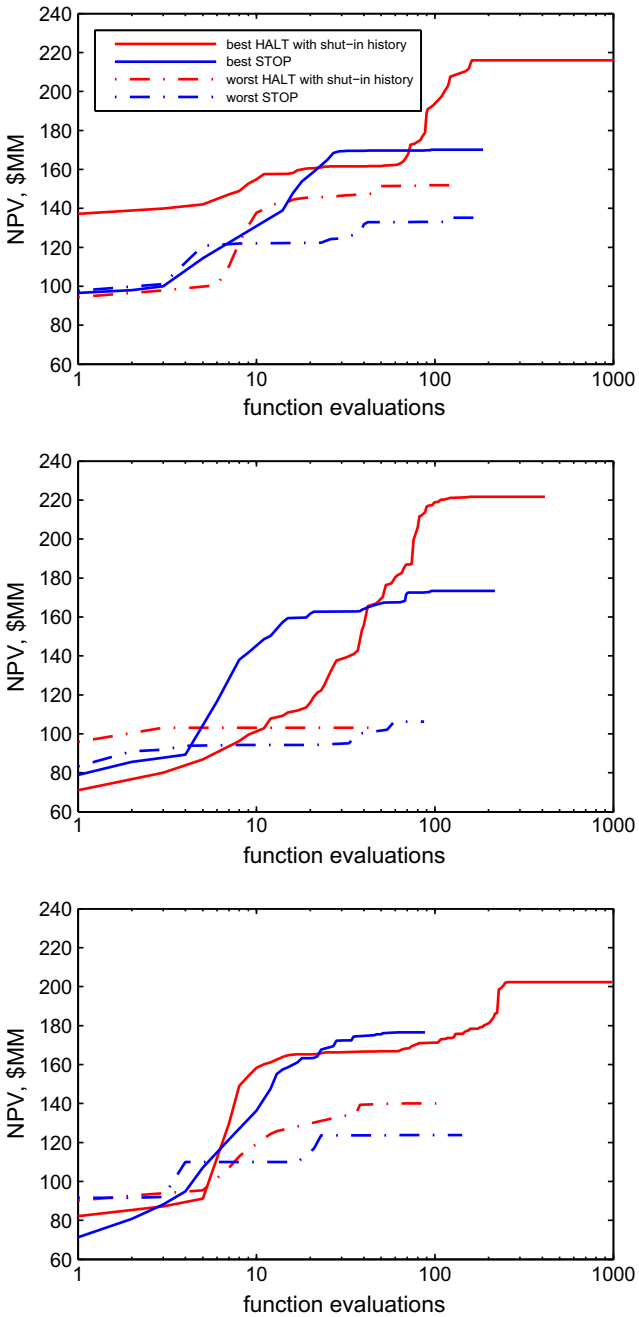

Fig. 6 Convergence of the NPV with respect to the number of objective function evaluations for the best (*solid*) and worst (*dash-dot*) runs from a set of 20 optimization runs launched with different (random) initial guesses. Each optimization is run once with the HALT with shut-in history (*red*) and then with the STOP (*blue*) enforcement mode. Each plot corresponds to a different realization of permeability

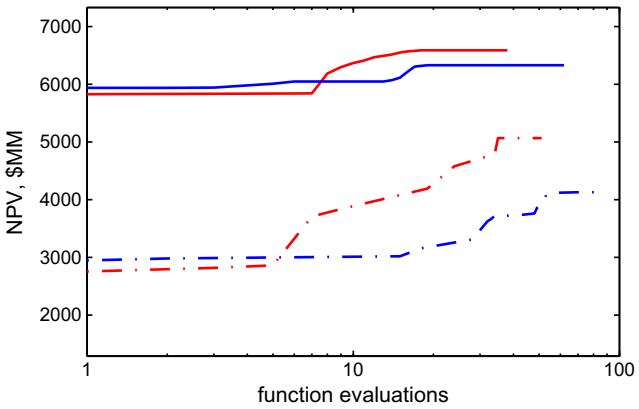

Fig. 7 NPV convergence curve with respect to the number of objective function evaluations for case study 2. The best (*solid*) and worst (*dash-dot*) optimization runs from among ten random initial guesses are shown. For each initial guess, an optimization run using the HALT with shut-in history (*red*) and STOP (*blue*) enforcement modes has been performed

criteria mentioned here that is activated depends in large part on the type of constraint enforcement being implemented. Next, we discuss the role of these criteria in the termination of the different runs in case study 1.

When optimizing the controls of wells subject to possible shut-ins, the gradient-based SQP algorithm may underperform when trying to meet the gradient-based convergence criterion. Because the SQP algorithm acts on the well controls only, it has no direct influence on the well statuses. A consequence is that the algorithm can get caught up in an extended sequence of sub-optimal solutions (resulting from unplanned shut-ins) in which the gradient norm is above the specified tolerance. This happens when all candidate control updates produce shut-in patterns that yield lower NPV values than the current one. The occurrence of shut-ins may thus hinder the algorithm from reaching the tolerance of the gradient-based termination criterion and thus, end up generating unnecessary function evaluations. Such termination can be observed in Fig. 5 in the convergence plots of the HALT (dashed red) and SHUT (dashed green) modes. In these plots, the monotonically increasing part ends with a flat line indicating several minor iterations with sub-optimal shut-in patterns. Thus, in terms of convergence properties, taking into account the shut-in history seems to counteract premature convergence caused by discrepancies between the algorithm update and the actual controls imposed during simulation. As seen in Fig. 5, the strategies that save the optimal shut-in pattern and reapply it in subsequent iterations, i.e., SHUT and HALT, both with shut-in history, continue to improve the optimal solution. Interestingly, this result can be seen in light of the argument made at the end of Section 5.2 that discusses how additional, reused, control points can improve algorithmic performance.

7.4 Economic limits enforced by constrained optimization

The comparison between the results obtained when using constrained optimization (constraints enforced at algorithm level) against the results obtained when using the STOP and HALT modes of enforcement expands the conclusions presented in [12] regarding the constraints used as a control type compared to constraints imposed by economic limits. In addition, we want to mention that this indirect type of enforcement is less accurate than the direct ones because the non-linear constraints implemented at the algorithm level are only an approximation of the economic limits. In fact, the economic limits are approximated twice before being

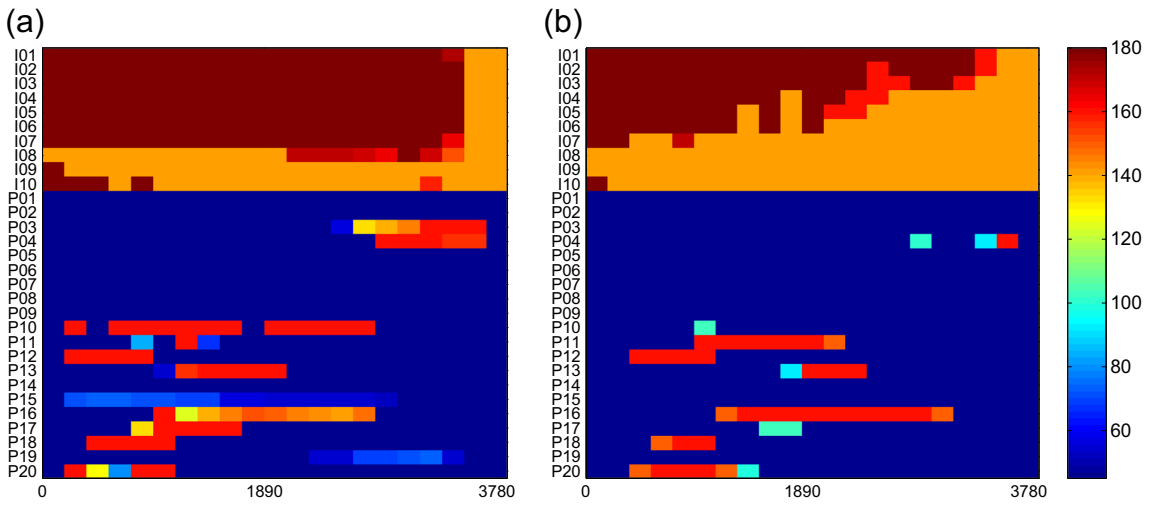

Fig. 8 Optimal solutions of well BHP controls for two enforcement modes: HALT with shut-in history (**a**) and STOP (**b**), obtained with the tenth initial guess for case study 2

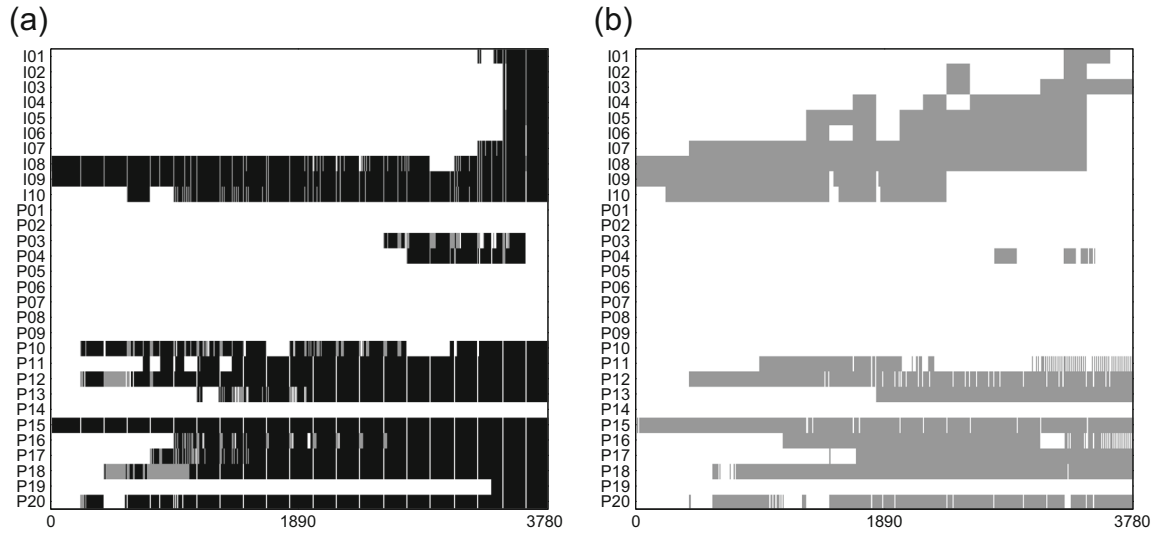

Fig. 9 Status of the wells in the optimal solutions for two enforcement modes: HALT with shut-in history (**a**) and STOP (**b**), obtained with the tenth initial guess for case study 2. The *white*, *gray*, and *black* designate, respectively, the open, stopped, and shut well status

supplied to the non-linear programming solver. They are first approximated when we use the previously run simulation to estimate the relevant performance profile of the well. Then, a second approximation step occurs when we supply SQP with only the first order derivatives of these performance profiles with respect to the well controls. Utilizing only linearized constraints is a limitation of the SQP algorithm.

7.5 Sensitivity analysis

In this section, we study the robustness of these results with respect to the initial guess and the geometrical complexity of the model.

First, we optimize case study 1 with 20 random initial guesses obtained by perturbing the base case with a 5% Gaussian noise. In Fig. 6, we show three plots, each corresponding to a different realization. Each plot shows the NPV function evolution corresponding to the best (solid line) and the worst (dash-dot line) optimization run from among the 20 initial guesses considered. For each pair of runs, we compare the HALT with shut-in history (red) against the STOP (blue) mode. The results confirm that the HALT-based algorithm outperforms the STOP-based one over a wide range of initial guesses. A large variation between the best and worst solutions underlines the high multimodality of the production optimization problem. Because the SQP algorithm is of a local search type, it cannot address the multimodality as efficiently as a global-local hybrid algorithm can. In the future extension of this study, the authors will investigate the performance of hybrid algorithms for production optimization problems with economic limits.

The second sensitivity result is obtained by performing the above analysis to case study 2. Here, we run ten optimization runs with various initial guesses. Nine of those guesses are generated by perturbing the base case with a 5% Gaussian noise, and the tenth guess is assigned with producers/injectors set to the lower/upper bounds of the BHP controls (a "full-blast" configuration).

Figure 7 illustrates the convergence of the best (solid line) and the worst (dash-dot line) optimization runs. The red and blue colors correspond to HALT with shut-in history and STOP modes, respectively. The best function evolution curve (solid line in Fig. 7) is obtained when using the tenth initial guess, i.e., the full-blast configuration for producers/injectors. Although the optimal NPVs vary over a wide range, the HALT mode clearly outperforms the STOP mode. The difference between the two modes is evident when comparing the well BHP controls and well status as shown in Figs. 8 and 9. In these figures, we notice that the injectors remain open for a longer time in the HALT mode than those in the STOP mode. This behavior is similar to the one observed in case study 1 and commented in detail in Section 7.1.

8 Conclusions

The main inference in this work is that a non-valid standard adjoint formulation will yield inaccurate gradients, and that this, in turn, will reduce the performance of a gradient-based algorithm for production optimization. The standard adjoint formulation is subject to inconsistencies whenever discontinuities are introduced within the simulator's

well model equations due to the enforcement of economic constraints.

This type of constraints are commonly enforced within the reservoir simulator when performing gradient-based production optimization. For this reason, in this work, we devise an improved implementation of the simulator-based constraints for which the adjoint gradients have fewer consistency errors. In this new implementation, the well model equations that violate the constraints are removed from the governing system right after the occurrence of the violation and not reinserted until the next well status update. The shut-in periods for these wells last just long enough so as to mitigate the resulting non-differentiability and loss in consistency of the standard adjoint formulation, though without inducing a major loss in the sensitivity of the gradient. Further analysis requires a broad quantitative comparison of the effects of lost sensitivity versus loss of consistency, and is the subject of ongoing work.

Numerical experiments show that the implementation of economic limit enforcement introduced in this paper outperforms commonly used implementations. In this respect, results from the test cases confirm that the proposed implementation speeds up convergence and improves the final optimal solution. Finally, a supplementary strategy that takes into account shut-in history is presented and tested. To further improve convergence, this strategy stores the shut-in pattern corresponding to the best simulation run so far, and applies it as initial control configuration in subsequent iterations of the optimization algorithm. Other possible applications of the strategy are noticed, e.g., the determination of the economic life time of the reservoir, but are deferred to future work by the authors.

Acknowledgments The authors are grateful to the industrial affiliates of the Stanford University Reservoir Simulation Research (SUPRI–B) and Smart Fields Consortia, and the Center for Integrated Operations in the Petroleum Industry, NTNU, for their financial support. We also thank Timur Garipov for useful discussions and suggestions on different aspects of this work.

References

1. Bellout, M.C., Echeverría Ciaurri, D., Durlofsky, L.J., Foss, B., Kleppe, J.: Joint optimization of oil well placement and controls. Comput. Geosci. **16**(4), 1061–1079 (2012)
2. Bukshtynov, V., Volkov, O., Durlofsky, L.J., Aziz, K.: Comprehensive framework for gradient-based optimization in closed-loop reservoir management. Comput. Geosci. **19**(4), 877–897 (2015)
3. Cao, Y., Li, S., Petzold, L., Serban, R.: Adjoint sensitivity analysis for differential-algebraic equations: the adjoint DAE system and its numerical solution. SIAM J. Sci. Comput. **24**(3), 1076–1089 (2003)
4. Chen, C., Li, G., Reynolds, A.: Robust constrained optimization of shortand long-term net present value for closed-loop reservoir management. SPE J. **17**(3), 849–864 (2012)
5. Forouzanfar, F., Rossa, E., Russo, R., Reynolds, A.: Life-cycle production optimization of an oil field with an adjoint-based gradient approach. J. Pet. Sci. Eng. **112**, 351–358 (2013)
6. Geoquest: eclipse technical description Schlumberger (2005)
7. Gill, P., Murray, W., Saunders, M.: User's Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming Stanford University (2008)
8. Gill, P.E., Murray, W., Saunders, M.: SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM J. Optim. **12**(4), 979–1006 (2002)
9. Haldorsen, H.: Choosing between rocks, hard places and a lot more: the economic interface. Norwegian Petroleum Society Special Publications **6**(C), 291–312 (1996)
10. Jansen, J.: Adjoint-based optimization of multi-phase flow through porous media—a review. Comput. Fluids **46**(1), 40–51 (2011)
11. Jansen, J.D., Fonseca, R.M., Kahrobaei, S., Siraj, M.M., Van Essen, G.M., Van den Hof, P.M.J.: The egg model—a geological ensemble for reservoir simulation. Geoscience Data Journal **1**(2)
12. Kourounis, D., Durlofsky, L.J., Jansen, J., Aziz, K.: Adjoint formulation and constraint handling for gradient-based optimization of compositional reservoir flow. Comput. Geosci. **18**(2), 117–137 (2014)
13. Liu, X., Reynolds, A.C.: Gradient-based multi-objective optimization with applications to waterflooding optimization. Comput. Geosci. **20**(3), 677–693 (2016)
14. de Montleau, P., Cominelli, A., Neylon, K., Rowan, D., Pallister, I., Tesaker, O., Nygard, I.: Production optimization under constraints using adjoint gradients. 10th European Conference on the Mathematics of Oil Recovery (A41) (2006)
15. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer, New York (2006)
16. Peters, E., Chen, Y., Leeuwenburgh, O., Oliver, D.: Extended Brugge benchmark case for history matching and water flooding optimization. Comput. Geosci. **50**, 16–24 (2013)
17. Priestley, H.: Introduction to integration. Oxford University Press, Oxford (1997)
18. Riesz, F.: Sur les opérations fonctionnelles linéaires. C. R. Acad. Sci. Paris **149**, 974–977 (1909)
19. Sarma, P., Durlofsky, L.J., Aziz, K., Chen, W.H.: Efficient real-time reservoir management using adjoint-based optimal control and model updating. Comput. Geosci. **10**(1), 3–36 (2006)
20. Svanberg, K.: The method of moving asymptotes - a new method for structural optimization. Int. J. Numer. Methods Eng. **24**(2), 359–373 (1987)
21. Talvila, E.: Necessary and sufficient conditions for differentiating under the integral sign. Am. Math. Mon. **108**(6), 544–848 (2001)
22. Van Essen, G., Zandvliet, M., Van Den Hof, P., Bosgra, O., Jansen, J.: Robust waterflooding optimization of multiple geological scenarios. SPE J. **14**(1), 202–210 (2009)
23. Volkov, O., Voskov, D.: Effect of time stepping strategy on adjoint-based production optimization. Comput. Geosci. **20**(3), 707–722 (2016)
24. Younis, R.: Modern advances in software and solution algorithms for reservoir simulation. PhD Thesis, Stanford University (2011)
25. Zhou, Y.: Parallel general-purpose reservoir simulation with coupled reservoir models and multi-segment wells. PhD Thesis, Stanford University (2012)