

Constraint handling for gradient-based optimization of compositional reservoir flow

Drosos Kourounis¹ · Olaf Schenk¹

Received: 8 November 2014 / Accepted: 27 August 2015 / Published online: 14 September 2015
© Springer International Publishing Switzerland 2015

Abstract The development of adjoint procedures for general compositional flow problems is much more challenging than for oil-water problems, due to significantly higher complexity of the underlying physics. The treatment of nondifferentiable constraints, an example of which is a maximum gas rate specification in injection or production wells, when the control variables are well-bottom-hole pressures, poses an additional major challenge. A new formal approach for handling these constraints is presented and compared against a formal treatment within the optimizer employing constraint lumping and a simpler heuristic treatment in the forward model. The three constraint-handling methods are benchmarked for three example cases of increasing complexity. Moreover, the new approach allows the optimizer to converge to optimal solutions exhibiting higher objective values, since unlike the formal lumping-based methods, where a pressure reduction suggested by the optimizer propagates through the smoothing function to all well rates, it handles constraints individually on a per well and per time step basis. The numerical examples show that the new formal constraint-handling approach allows the optimizer to converge significantly faster than formal lumping-based techniques independently of the initial guess used for the optimization.

Keywords Adjoint formulation · Gradient-based optimization · Production optimization · Recovery optimization · Compositional reservoir simulation · Discrete adjoint · Continuous adjoint · Automatic differentiation · Nonlinear constraints · General constraints

1 Introduction

The optimization of time-varying well settings, like for example, the bottom-hole pressures (BHPs), injection rates, or production rates, is an important aspect of optimal reservoir management. When the simulator source code is accessible, a gradient-based optimization method, in which the gradient is computed using an adjoint formulation, is often the method of choice since it is generally the most efficient. Although not considered here, derivative-free methods can also be applied for production optimization problems—see [13] for discussion and examples.

Adjoint gradient-based optimization techniques have been used in a reservoir simulation setting both for history matching (see, e.g., [6, 8, 20, 25, 28]) and for production optimization. Much of the early work on their use for optimization of oil recovery was performed by Ramirez and coworkers, who considered the optimization of several different enhanced oil recovery (EOR) processes [22, 23, 26]. In subsequent work, the focus was on gradient-based optimization (and in some cases on the optimization of “smart wells”) for water flooding [1, 4, 28, 29, 32]. Recent studies have addressed the implementation of adjoint-based procedures into general purpose simulators, the treatment of general constraints, and regularization and other numerical issues [12, 16, 21, 27]. Refer to [17] for a more complete overview of adjoint-based optimization methods.

✉ Drosos Kourounis
drosos.kourounis@usi.ch
Olaf Schenk
olaf.schenk@usi.ch

¹ Institute of Computational Science, Faculty of Informatics, Università della Svizzera Italiana, Via Giuseppe Buffi 13, 6904 Lugano, Switzerland

Although much of the early (1980s) work noted above focused on the application of adjoint procedures for EOR problems, there has not been much work on the use of adjoint techniques for large-scale (practical) compositional reservoir simulation problems. This is likely due to the complexity entailed in implementing adjoint procedures into a general purpose compositional reservoir simulator and to the challenging computational problems that must be solved to perform the optimizations. Compositional simulation is inherently more challenging than black-oil simulation because of the need to perform phase-equilibrium (flash) calculations for all grid blocks at every iteration of every time step. Adjoint formulations are challenging to code because they require analytical derivatives of many variables, and the increased complexity of compositional simulators renders these derivatives much more cumbersome to calculate than in the case of a black-oil simulator.

Recently, an adjoint treatment for multicomponent oil-gas compositional systems was presented in [18]. The formulation included an extensive discussion on engineering constraints that should usually be taken into account in realistic scenarios. These constraints appear either as bounds (box constraints) on the control variables or as inequality constraints on nonlinear functions of the controls and states of the underlying PDEs. Two treatments were proposed for the nondifferentiable constraints: a formal treatment within the optimizer performing lumping for all wells and time steps, and a heuristic approach, where bound constraints are treated in the optimization and nondifferentiable constraints are satisfied in the forward model. The investigation showed that although standard lumping techniques perform well for simple academic problems, they fail to obtain optimal solutions better than the reference for realistic problems. That result motivated further developments of formal constraint-handling techniques. In the present work, we introduce a new formal treatment for the nondifferentiable constraints where lumping is avoided to allow for a more realistic discretization of the nonlinear constraints. The performance of the new approach is compared to the ones introduced in [18] for several different examples of increased complexity.

The paper introduces the underlying PDEs in Section 2 and presents the discrete adjoint formulation and the solution of the adjoint linear systems in Section 3. Heuristic and formal constraint-handling approaches are presented in Section 4 along with the new lumping-free treatment, which is the main contribution of this work. The gradients of the objective function and the Jacobian of the nonlinear constraints are obtained using the discrete adjoint formulation and forwarded to a sequential quadratic programming software package SNOPT described in Section 5. Numerical results demonstrating the capabilities of our optimization procedure, for a series of two- and three-dimensional problems involving different numbers of hydrocarbon components

and wells, are presented in Section 6. Conclusions and suggestions for future work are provided in Section 7.

2 Oil-gas compositional simulation equations

The mass conservation equation for component i , which can exist in any phase j (here $j = o, g$, where o indicates oil and g gas), is given by [5, 33, 34]

$$\frac{\partial}{\partial t} \left(\phi \sum_j x_{ij} \rho_j S_j \right) - \nabla \cdot \left(\sum_j x_{ij} \rho_j \mathbf{K} \frac{k_{rj}}{\mu_j} \nabla \Phi_j \right) + \sum_w \sum_j x_{ij} \rho_j q_j^w = 0, \quad i = 1, \dots, n_c. \quad (2.1)$$

In the first (accumulation) term, t is time, ϕ is porosity, x_{ij} designates the mole fraction of component i in phase j , S_j is saturation, and ρ_j is molar density. In the second (flow) term, \mathbf{K} is the permeability tensor, k_{rj} is the relative permeability to phase j , μ_j is the phase viscosity, and the phase potential Φ_j is given by $\Phi_j = p_j - \rho_j g(D - D_0)$, where p_j is phase pressure, D is depth, D_0 is a reference depth, and g is gravitational acceleration. In the third (source/sink) term, q_j^w indicates the phase flow rate for well w . The treatment of this term will be discussed in Section 4.3. Equation 2.1 is written for each of the n_c components present in the system.

For a mixture of n_c components in two fluid phases (oil and gas), thermodynamic equilibrium can be expressed as

$$f_{io}(p_o, x_{io}) - f_{ig}(p_g, x_{ig}) = 0, \quad (2.2)$$

where $f_{io}(p_o, x_{io})$ is the fugacity of component i in the oil phase and $f_{ig}(p_g, x_{ig})$ is the fugacity of component i in the gas phase (temperature does not appear because the system is assumed to be isothermal). We additionally must satisfy the saturation constraint ($S_o + S_g = 1$) and the component mole fraction constraints

$$\sum_{i=1}^{n_c} x_{io} - 1 = 0, \quad \sum_{i=1}^{n_c} x_{ig} - 1 = 0. \quad (2.3)$$

A capillary pressure relationship also appears in cases with nonzero capillary pressure, though here we neglect capillary pressure so $p_o = p_g$. For the wells, we adopt the standard well model relating the pressure of the well p_w with the well rate of phase j q_j^w through the formula

$$(q_j^w)_l = \left(T_w \frac{k_{rj}}{\mu_j} \right)_l (p_l - p_{w,l}), \quad (2.4)$$

where T_w is the well index (or well transmissibility), $p_{w,l}$ is the wellbore pressure for the well in block l , and p_l is

the well-block pressure. Here, $p_{w,l}$ is related to the BHP through an additional well equation. When the well rate is specified, equations of the form (2.4) can be used to compute $p_{w,l}$. For more details on well models, and on the relationship between $p_{w,l}$ and BHP, see [5].

As discussed by many authors (see, e.g., [5, 10, 33, 36]), the system described above contains a total of only n_c primary equations and primary variables per grid block. These equations and variables are coupled (from block to block), and in a fully implicit method are all computed simultaneously at each Newton iteration. The remaining (secondary) variables can be computed locally (block by block), and thus very efficiently, once the primary variables are determined. Various options exist for the choice of primary variables (see [33] for a discussion). Here, we use the so-called natural variable set, which includes, for each grid block, one pressure unknown, $n_p - 1$ saturation unknowns (where n_p is the number of phases; here $n_p = 2$), and $n_c - n_p$ component mole fraction unknowns.

2.1 Discretization of the underlying PDEs

In our formulation, the governing equations (2.1) are solved fully implicitly, using a backward-Euler time discretization, two-point flux approximation, and single-point upwinding [2]. These treatments are standard in practical reservoir simulation. For the solution of the set of nonlinear equations, we use Newton's method with the solution at the previous time step as the initial guess. A limit on the change of the grid-block saturation and mole fractions over a Newton iteration is applied [38]. The Newton iterations terminate when the maximum relative norm of the residual is less than 10^{-6} (tight convergence criteria are required for the adjoint solution, discussed below). For the solution of the linear system at each Newton iteration, we use GMRES preconditioned by the constrained pressure residual method, as described in [16]. Iteration is terminated when the Euclidean norm of the initial residual has decreased by five orders of magnitude.

We employ a simple time stepping strategy. The time step size at step $n + 1$ is a multiple of that at n , provided nonlinear convergence was achieved at step n . In this way, the time step can increase until it reaches the maximum allowable value. If the nonlinear solver fails to converge within a prescribed number of Newton iterations, we divide the time step by a fixed constant. This process is repeated until the nonlinear system converges.

3 Adjoint equations for the compositional system

We now present the discrete adjoint equations. Some numerical and coding issues are also discussed.

3.1 Automatic differentiation

Automatic differentiation, or AD, is gaining popularity in the field of scientific computing as a means of facilitating the development and enhancement of large code bases. AD enables, for example, the fast (analytical) determination of Jacobian matrix elements from the code defining the residual vector. The use of AD has allowed the fast construction and assessment of different compositional formulations within the same code [34]. In this work, we take advantage of AD to automate the construction of many of the derivatives required for the adjoint formulation. The AD implementation used in our compositional simulator is the “automatic differentiation expression templates library” (ADETL), developed originally by Younis and Aziz [37].

3.2 Discrete adjoint formulation

Following the fully implicit discretization of the governing equations (using the usual finite volume method, with treatments as noted above), we can express the nonlinear system as

$$\mathbf{g}_n(\mathbf{x}_n, \mathbf{x}_{n-1}, \mathbf{u}_n) = \mathbf{0}, \quad (3.1)$$

where \mathbf{g}_n denotes the fully discretized, both in space and time, set of partial differential equations. Here, $\mathbf{x}_n = \mathbf{x}(t_n)$ and $\mathbf{u}_n = \mathbf{u}(t_n)$ are the states and controls (well settings), respectively, at time step n . The controls \mathbf{u} , as it can be seen from Eq. 2.4 appear in the discrete system (3.1) only at the n th step, since the discretization in time is fully implicit. The corresponding time step size is designated Δt_n . We will use throughout the notation $\partial \mathbf{g}^T / \partial \mathbf{x}$ to denote the matrix $(\partial \mathbf{g} / \partial \mathbf{x})^T$.

We are interested in either maximizing or minimizing an objective function J that is in general a nonlinear function of the states \mathbf{x}_n and the controls \mathbf{u}_n of the forward problem. We assume that J has the following form:

$$J(\mathbf{x}, \mathbf{u}) = \int_{t_0}^{t_N} f(\mathbf{x}(t), \mathbf{u}(t)) dt + \varphi(\mathbf{x}(t_N)), \quad (3.2)$$

where $f(\mathbf{x}(t), \mathbf{u}(t))$ is a nonlinear function varying with time and $\varphi(\mathbf{x}(t_N))$ is a function of only the last state \mathbf{x}_N . After the solution of the forward problem has been obtained, J may be approximated by

$$J \approx \sum_{n=1}^N \Delta t_n f_n(\mathbf{x}_n, \mathbf{u}_n) + \varphi(\mathbf{x}_N). \quad (3.3)$$

Using (3.3), we can state the optimal control problem as

$$\begin{aligned} \text{(NLP)} \quad & \underset{\mathbf{u}}{\text{minimize}} \quad J = \sum_{n=1}^N \Delta t_n f_n(\mathbf{x}_n, \mathbf{u}_n) + \varphi(\mathbf{x}_N) \\ & \text{subject to} \quad \mathbf{g}_n(\mathbf{x}_n, \mathbf{x}_{n-1}, \mathbf{u}_n) = \mathbf{0}, \mathbf{x}_0 = \mathbf{x}(t_0) \end{aligned}$$

In general, a number of linear and nonlinear constraints may need to be included in the nonlinear optimal control problem (NLP). We postpone the discussion of their treatment until Section 4. Now, since $\mathbf{g}_n = \mathbf{0}$, we can introduce the augmented objective function J_A by “adjoining” the governing equations to the original objective function J . The new objective J_A shares the same extrema as J and is defined as

$$J_A = \sum_{n=1}^N \left(\Delta t_n f_n(\mathbf{x}_n, \mathbf{u}_n) + \boldsymbol{\lambda}_n^T \mathbf{g}_n(\mathbf{x}_n, \mathbf{x}_{n-1}, \mathbf{u}_n) \right) + \varphi(\mathbf{x}_N). \quad (3.4)$$

In Eq. 3.4, the vectors $\boldsymbol{\lambda}_n$ are the Lagrange multipliers.

The maximum or minimum of J_A (and thus J) is achieved when the first variation of J_A is zero ($\delta J_A = 0$). After performing some index shifting, and grouping terms multiplied by the same variation ($\delta \mathbf{x}_n, \delta \mathbf{x}_N, \delta \mathbf{u}_n$), δJ_A can be written as

$$\begin{aligned} \delta J_A = & \left(\frac{\partial \varphi_N}{\partial \mathbf{x}_N} + \Delta t_N \frac{\partial f_N}{\partial \mathbf{x}_N} + \boldsymbol{\lambda}_N^T \frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N} \right) \delta \mathbf{x}_N \\ & + \sum_{n=1}^{N-1} \left(\Delta t_n \frac{\partial f_n}{\partial \mathbf{x}_n} + \boldsymbol{\lambda}_{n+1}^T \frac{\partial \mathbf{g}_{n+1}}{\partial \mathbf{x}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{x}_n} \right) \delta \mathbf{x}_n \\ & + \sum_{n=1}^N \left(\Delta t_n \frac{\partial f_n}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n} \right) \delta \mathbf{u}_n. \end{aligned} \quad (3.5)$$

We can simplify δJ_A by ensuring that $\delta J_A / \delta \mathbf{x}_n = \mathbf{0}^T$ for $n = 1, 2, \dots, N$. We require that the Lagrange multipliers satisfy the following equations:

$$\frac{\partial \mathbf{g}_n^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_n = - \left(\frac{\partial \mathbf{g}_{n+1}^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_{n+1} + \Delta t_n \frac{\partial f_n^T}{\partial \mathbf{x}_n} \right), \quad (3.6)$$

$$\frac{\partial \mathbf{g}_N^T}{\partial \mathbf{x}_N} \boldsymbol{\lambda}_N = - \left(\Delta t_N \frac{\partial f_N^T}{\partial \mathbf{x}_N} + \frac{\partial \varphi_N^T}{\partial \mathbf{x}_N} \right). \quad (3.7)$$

With this choice of the Lagrange multipliers, the total variation becomes

$$\delta J_A = \sum_{n=1}^N \left(\Delta t_n \frac{\partial f_n}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n} \right) \delta \mathbf{u}_n,$$

and the gradient of the objective function with respect to the controls is

$$\frac{\delta J_A}{\delta \mathbf{u}} = \left[\frac{\delta f_1}{\delta \mathbf{u}_1}, \frac{\delta f_2}{\delta \mathbf{u}_2}, \dots, \frac{\delta f_N}{\delta \mathbf{u}_N} \right]. \quad (3.8)$$

The individual entries of $\delta J_A / \delta \mathbf{u}$ are given by

$$\frac{\delta f_n}{\delta \mathbf{u}_n} = \Delta t_n \frac{\partial f_n}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n}, \quad n = 1, 2, \dots, N. \quad (3.9)$$

By driving $\delta J_A / \delta \mathbf{u}$ to zero, we achieve the minimum or maximum of J_A (and thus J). In practice, $\delta J_A / \delta \mathbf{u}$, along with other quantities related to constraints, is provided to a gradient-based optimization algorithm to determine the next estimate for the controls \mathbf{u} .

In optimization problems, the well-control variables do not typically change at each time step in the flow simulation. Rather, they are defined over longer time periods that are referred to as control steps. Time steps are usually small in order to capture flow dynamics, reduce time-discretization error, and facilitate convergence of the Newton iterations. The gradient at the control period m , $\delta f_n / \delta \mathbf{u}_m$, is simply the sum of the gradients $\delta f_n / \delta \mathbf{u}_n$ for all time steps that belong to control period m .

3.3 Solution of adjoint equations

The solution of the linear system of equations that arises when solving (3.6) constitutes the largest computational demand in the adjoint problem. The matrix appearing in this equation at time step n , $\partial \mathbf{g}_n^T / \partial \mathbf{x}_n$, is the transpose of the Jacobian matrix for the converged forward problem, $\partial \mathbf{g}_n / \partial \mathbf{x}_n$. In our implementation, the converged states are written to disk during the solution of the forward problem. These converged states are then read back, during the solution of the adjoint problem, and $\partial \mathbf{g}_n / \partial \mathbf{x}_n$ is reconstructed, along with all other derivatives appearing in Eqs. 3.6, 3.7, and 3.9. This enables the evaluation of the Lagrange multipliers $\boldsymbol{\lambda}_n$ and the gradients $\partial f_n / \partial \mathbf{u}_n$.

For the solution of the linear system in Eq. 3.6, we use GMRES preconditioned by the transpose of the constrained pressure residual preconditioner (CPR), as described in [16]. The underlying physics of reservoir dynamics are of parabolic, due to pressure, and of hyperbolic nature, on every other variable. An efficient preconditioner P should consist of two steps each tailored to the coupled physics of the problem. At first, a pressure matrix A_p is extracted from the Jacobian A , by canceling out derivatives corresponding to variables that would have been treated explicitly, had we assumed an implicit-pressure explicit-saturation (IMPES) formulation; see [16] for details. Then the right-hand side of the original system b after the application of the pressure operator that extracts A_p from A is restricted to the pressure unknowns, through the application of a restriction operator C^T , and the linear system $A_p x_p = C^T b_p$ is solved for the pressure solution x_p prolonged through the application of C to obtain the solution of the first stage of the preconditioning $x_1 = C x_p$. Once the parabolic character of the solution has been captured in x_1 at the first stage, the residual $b_p - A x_1$ can be efficiently preconditioned by a preconditioner tailored for hyperbolic problems in an attempt to capture the hyperbolic component x_2 . Such a preconditioner is the block-ILU0 (BILU0) [14]

although any other preconditioner for hyperbolic problems can be used at this stage. The solution of the two-stage preconditioning process is the sum of the two individual solutions $x = x_1 + x_2$. The two steps are summarized in Eq. 3.10:

$$P^{-1}b = \underbrace{M^{-1}(b - A \underbrace{C A_p^{-1} C^T b}_{\text{residual}})}_{x_2} + \underbrace{C A_p^{-1} C^T b}_{\text{pressure system}}^{x_1}. \quad (3.10)$$

The preconditioner for the transposed system is obtained in a similar way. Transposing expression (3.10), we observed that the two steps required for the solution of the transposed system are now applied in reverse order:

$$P^{-T}b = \underbrace{C A_p^{-T} C^T (b - A \underbrace{M^{-T} b}_{x_1})}_{x_2} + \underbrace{M^{-T} b}_{x_1}. \quad (3.11)$$

The first step now captures the hyperbolic part of the solution and the second step captures its parabolic part. All operators are transposed. We should mention here that for the solution of the adjoints systems, we require very high accuracy to guarantee that residual errors accumulated over hundreds of time steps do not pollute the gradients (which would influence the computed optimum). For this reason, we continue iterating the linear solver until the Euclidean norm of the initial residual has decreased by 10 orders of magnitude. This is significantly higher accuracy than the accuracy required to allow Newton's method to decrease the norm of the residual of each nonlinear system solved at each time step of the forward problem.

4 Nonlinear constraints

Several authors have discussed the incorporation of constraints in the optimal control problem for oil recovery optimization. This includes partial and sometimes heuristic approaches, valid for particular types of constraints [4, 31, 32, 35], and more systematic approaches, valid for a broader range of constraint equations [7, 11, 19, 27, 30]. An important feature in simulations involving highly compressible fluids, which we have in the systems considered here since we inject gas, is the occurrence of transient peaks in the rate in response to changes in well BHP. These transient effects impact simulation results in our models because they occur over time scales that are larger than a time step (this is in contrast to nearly incompressible systems, where this type of transient decays very quickly and is not resolved by the simulator). These transient effects lead to challenges in constrained optimization of compositional systems because

rate constraints can be easily violated. Here, we will present and assess three formal constraint handling approaches and a simpler heuristic procedure for satisfying rate constraints, which can be used in compressible systems. A variety of methods have been proposed to incorporate nonlinear constraints in the adjoint formulation; see, e.g., [9, 17, 18, 28, 30] for detailed discussion.

4.1 Constraint lumping

Many constraints appear as simple bound constraints (e.g., BHP limits in a problem where BHPs are the control variables), but in other cases the constraints are nonlinear since a (nonlinear) simulation is required to evaluate them. Examples are weighted integral quantities of the rates like the total water or gas injected over a period of interest and the net present value among others. Constraints that are described by nondifferentiable functions can be challenging to incorporate. A constraint of this type that appears frequently in production optimization problems is the maximum (or minimum) well flow rate constraint, e.g., for injection constraints, the rate of the injected phase (gas and water) is not allowed to exceed a prescribed upper bound q^+ that is mainly determined by

$$q_{jn}^p \leq q^+, \quad j = 1, \dots, N_w, \quad n = 1, \dots, N, \quad (4.1)$$

where q_{jn}^p is the rate of the injected phase p (gas or water) at well j at time step n and q^+ is a specified maximum rate. To satisfy this constraint, we must guarantee that

$$\max_{j,n} (q_{jn}) \leq q^+, \quad j = 1, \dots, N_w, \quad n = 1, \dots, N. \quad (4.2)$$

However, the max function is not differentiable so it cannot be used to provide gradient information. The approximation of the max function by smooth and differentiable functions is known in the literature as *constraint lumping*. This lumping can be performed on a well-by-well basis, in which case $O(N_w N)$ linear systems must be solved for the evaluation of the Lagrange multipliers, or over the entire model, in which case only $O(N)$ linear systems must be solved [18, 27]. The approximate gradients of the lumped nonlinear constraints are then obtained in the same manner as the gradient of the objective function, i.e., in terms of time step contributions.

Exponential smoothing can be used to approximate the max and min functions as dictated by Bertsekas [3]. For lumping over the entire model, a smooth function c , approximating the max function, is introduced having the form

$$c \approx \alpha \sum_{j=1}^{N_w} \log \sum_{n=1}^N e^{q_{jn}/\alpha}, \quad (4.3)$$

$$\alpha = 0.05q^+. \quad (4.4)$$

The smaller the coefficient multiplying q^+ in Eq. 4.4, the more accurate the approximation of max becomes. The numerical value used here, 0.05, ensures that no overflow occurs in any of the exponential terms in the summation. The approximation of max in Eq. 4.4 is always greater than the maximum of the component well rates, so if this maximum is honored, the true constraint is guaranteed to be satisfied. The derivatives of that function are obtained as described above. For details, see [18].

It is important to recognize that the approach used for constraint lumping can impact the performance of the optimization procedure. Bound constraints on the controls do not require any special treatment as they are readily handled by the optimizer.

4.2 Constraint handling without lumping

As indicated above, operational constraints in reservoir simulation are often inequality constraints specified for input and/or output rates at every time step. In the case no lumping is performed, the constrained optimal control problem (CNLP) reads

$$\begin{aligned} \text{(CNLP) minimize } J &= \sum_{n=1}^N \Delta t_n f_n(\mathbf{x}_n, \mathbf{u}_n) + \varphi(\mathbf{x}_N) \\ \text{subject to } \mathbf{g}_n(\mathbf{x}_n, \mathbf{x}_{n-1}, \mathbf{u}_n) &= \mathbf{0}, \\ \mathbf{x}_0 &= \mathbf{x}(t_0), \\ q_{jn} &\leq q_j^+, \\ j &= 1, \dots, N_w, \quad n = 1, \dots, N, \end{aligned}$$

where q_j^+ is an upper bound specified for each well or group of wells. In this case, the vector of constraints $\mathbf{c}(\mathbf{u})$ in (CNLP) can be of dimension $O(N_w N)$, where N is the number of time steps and N_w is the number of wells. Computing the gradients for each of the entries of $\mathbf{c}(\mathbf{u})$, which is required by SNOPT, will require N adjoint simulations and thus, in total, the solution of $O(N_w N^2)$ linear systems of equations in addition to each forward simulation. This may be computationally intractable for realistically sized problems. Nevertheless, this approach provides significantly higher flexibility to the optimizer in achieving feasible solutions, since the optimizer can manipulate rates for every single well at the particular time steps where feasibility is violated.

4.3 Constraint handling in the simulator

Enforcing feasibility from an infeasible solution, when the constraints are upper and lower bounds on output quantities (e.g., rates when BHPs are specified), can be readily accomplished during the forward simulation. In Eq. 2.4, wells can be controlled by specifying the rate q_j^w instead of the BHP $p_{w,l}$. When well rate is specified, equations of the form

of Eq. 2.4 can be used to compute $p_{w,l}$; see [5] for the details. More precisely, if a maximum gas production rate is specified and a well operating under BHP control violates this maximum, the well can be switched from BHP control to rate control and operated at the maximum rate. A more thorough presentation of the heuristic approach can be found in [18]. We note finally that if rates are used as the control variables, then the rate constraints enter the optimization problem as simple bound constraints, which are easy to satisfy. In this case, however, the BHPs become nonlinear constraints. Our heuristic treatment would then entail the switch from rate control to BHP control if the BHP constraint would otherwise be violated. We did not test the performance of our procedure using rates as the control variables, but this should be considered in future work.

5 Gradient-based optimization and related software

The SNOPT optimizer is used in this work for solving the general nonlinear constrained optimization problem (GNP)

$$\begin{aligned} \text{(GNP) minimize } f(\mathbf{u}) \\ \mathbf{u} \in \mathbb{R}^n \\ \text{subject to } \mathbf{b}_l \leq \begin{pmatrix} \mathbf{u} \\ \mathbf{c}(\mathbf{u}) \end{pmatrix} \leq \mathbf{b}_u, \end{aligned}$$

where $f(\mathbf{u})$ a linear or nonlinear objective function, $\mathbf{c}(\mathbf{u})$ is a vector of nonlinear constraint functions $c_i(\mathbf{u})$ with sparse derivatives, and \mathbf{b}_l and \mathbf{b}_u are vectors of lower and upper bounds. We assume that the nonlinear functions are smooth and that their first derivatives are available.

We note that the adjoint formulation described in this paper may also be used in conjunction with other optimization packages. SNOPT uses a sparse sequential quadratic programming (SQP) algorithm that exploits sparsity in the constraint Jacobian and maintains a limited-memory quasi-Newton approximation to the Hessian of the Lagrangian. The QP subproblems are solved using an inertia-controlling reduced-Hessian active-set method (SQOPT) that allows for variables appearing linearly in the objective and constraint functions.

In summary, the basic structure of an SQP method involves major and minor iterations. The major iterations generate a sequence of iterates \mathbf{u}_k that converge to the optimal solution \mathbf{u}^* . At each major iterate, a QP subproblem is solved to generate a search direction towards the next iterate \mathbf{u}_{k+1} . Solving such a subproblem is itself an iterative procedure, and the minor iterations of an SQP method are the iterations of the QP method. SNOPT requires first-order derivatives of the nonlinear objective and constraint functions with respect to the control variables, which are provided by our adjoint procedure. See [18] for a concise

overview of the underlying theory. A much more in-depth description can be found in [15].

6 Numerical results

The optimization framework introduced in the previous sections will be applied for optimizing cumulative oil production subject to bound and nonlinear constraints. We will present results for three different cases of increased complexity. The aim of the following benchmarks is to reveal the most robust approach for treating nonlinear constraints with respect to the quality of optimal solutions and speed of convergence.

Gradient-based algorithms are usually trapped in local optima. To allow for a more fair comparison, we start the optimization from nine different initial guesses for the well controls. Each initial guess corresponds to a combination of BHPs from the set $\{p_I^u, p_I^l, p_I^a\}$ for the injectors and from the set $\{p_P^u, p_P^l, p_P^a\}$ for the producers, where p^u , p^l , and p^a designate the upper and lower limits on the initial BHPs, and the average between these limits, respectively. We set $p^l = p_{\text{init}} + 1$ bar for the injectors and $p^u = p_{\text{init}} - 1$ bar for the producers, where p_{init} is the initial reservoir pressure. Note that these “limits” are simply used to prescribe initial guesses for the optimization—they are not related to the actual BHP bound constraints. For clarity, we will refer to each case by the number of the corresponding run, as listed in Table 1. For all examples presented, the permeability is described by a diagonal tensor: $\mathbf{K} = \text{diag}(\mathbf{K}_x, \mathbf{K}_y, \mathbf{K}_z)$.

In the results below, we apply the formal constraint handling approaches introduced in Sections 4.1, 4.2 for maximizing cumulative oil recovery, and we compare them against the treatment introduced in Section 4.3, which we refer to as “heuristic constraint handling”. The heuristic approach is realized in two steps. First, we perform the optimization without including the nonlinear constraints. Bound (linear) constraints are honored during the optimization. Then, after this initial optimization has converged, we

run the forward problem once more using the “optimized” BHPs, but this time the simulator is allowed to switch to rate control when required to satisfy the nonlinear constraints. Thus, the computational effort for this approach is little more than that required for optimizing the bound (and linearly) constrained problem, as just one additional simulation run is performed.

6.1 Example 1: Π obstacle

In the first example, cumulative oil recovery is maximized under CO_2 injection. The two-dimensional (2D) geological model used is depicted in Fig. 1. A Π -shaped practically impermeable region is introduced at the center of a homogeneous reservoir. The model is discretized on a 2D 80×80 grid. The permeability for the red cells is set to 4000 mD, while the permeability for the blue cells that comprise the Π -shaped region is set to 10^{-4} mD. Four injection wells are placed at the corners of the model, and the single production well is located inside the Π -shaped region. The fluid consists of a four component (three hydrocarbon components plus CO_2) mixture, as specified in Table 2. Further details on the reservoir model are provided in Table 3.

The well BHPs are constrained to lie between a lower bound of 90 bar and an upper bound of 120 bar. A maximum (per well) gas injection rate of $500 \text{ m}^3/\text{d}$ at reservoir conditions is additionally specified. The total simulation period is set to 256 days, and the well controls are determined at the initial time and for every subsequent 32-day interval. There are thus a total of eight control steps and consequently 40 control parameters.

Two reference solutions are generated. The first is obtained by running the forward simulation with the production wells operating at the minimum BHP (90 bar) and

Table 1 Initial guesses for the optimizations for all cases considered

Run	Initial guess
1	$[p_I^l, p_P^l]$
2	$[p_I^l, p_P^a]$
3	$[p_I^l, p_P^u]$
4	$[p_I^a, p_P^l]$
5	$[p_I^a, p_P^a]$
6	$[p_I^a, p_P^u]$
7	$[p_I^u, p_P^l]$
8	$[p_I^u, p_P^a]$
9	$[p_I^u, p_P^u]$

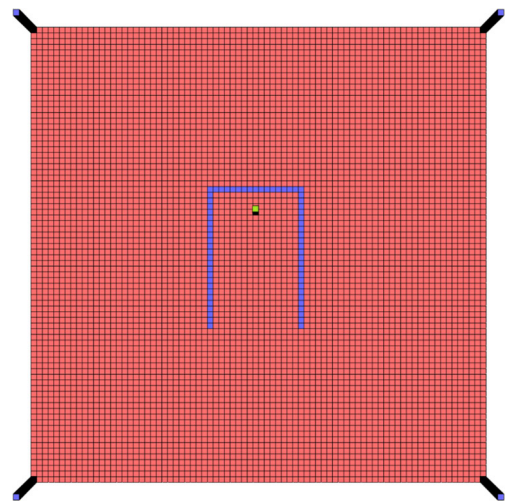


Fig. 1 Injection wells (blue) and production well (green) for example 1. Background shows \mathbf{K}_x ($\mathbf{K}_x = \mathbf{K}_y$)

Table 2 Fluid description for example 1

Component	CO ₂	C ₁	C ₄	C ₁₀
Initial composition (%)	1	20	29	50
Injection composition (%)	100	–	–	–

the injection wells at the maximum BHP (120 bar) and is infeasible since it violates the maximum gas injection rate. Next, we apply the heuristic constraint handling approach described above, with the maximum gas injection rate set to 500 m³/d. This is the well-scheduling approach that is usually applied in real-life fields, and it is usually nearly optimal and does not make any assumptions on the geological properties of the subsurface. The cumulative oil production for these two cases is given in the first row (“Ref.”) of Table 4. The table headings refer to the treatment of the nonlinear constraints—bound constraints are satisfied in all cases.

We next perform optimizations that honor the bound constraints but not the nonlinear constraints. The results for the nine runs, starting from different initial guesses, are presented in Table 4 in the column labeled “Unc.” The best optimum achieved is a cumulative oil production of 190,200 m³,

Table 3 Model parameters for example 1

Parameter	Value	Units
Grid size	80 × 80 × 1	–
Δx	6	m
Δy	6	m
Δz	4	m
Depth	4000	m
Initial pressure	100	bar
Temperature	100	°C
Rock compressibility	7.2 × 10 ^{−5}	1 / bar
Simulation time	256	d
Pressure upper bound	120	bar
Pressure lower bound	90	bar
Residual gas saturation	0	–
Residual oil saturation	0	–
End point rel perm gas	1	–
End point rel perm oil	1	–
Corey exponent gas	2	–
Corey exponent oil	2	–
Well locations [grid block no.]	<i>i</i>	<i>j</i>
Injector 1	1	1
Injector 2	1	80
Injector 3	80	1
Injector 4	80	80
Producer 1	40	48

Table 4 Oil production in 10³ m³ (example 1, 40 control variables) for the optimized objective function without satisfying the nonlinear constraints (“Unc.”), satisfying the nonlinear constraints using the heuristic treatment (“Hrst.”), using the formal lumping-based approach applied to the entire model (“Frm. 1”) and using the lumping-free approach (“Frm. 2”)

Run	Unc.	Hrst.	Frm. 1	Frm. 2
Ref.	163.9	152.2		
1	187.5	156.6	158.2	166.6
2	189.1	162.0	146.2	164.8
3	177.3	149.0	149.2	167.0
4	183.4	150.2	160.6	166.5
5	186.1	152.2	152.9	166.9
6	185.0	158.9	160.2	166.6
7	190.2	162.3	142.5	159.3
8	190.1	163.5	158.5	159.8
9	190.1	162.0	155.0	164.7
Average	186.5 (26)	157.4 (27)	154.0 (49)	164.7 (42)

Last row shows the average recovery from all nine runs per case followed in parentheses by the average number of iterations the optimizer needed to converge. Best feasible results shown in bold

obtained in run 7. This clearly exceeds the feasible reference result of 152,200 m³. Results using heuristic constraint handling are shown in the third column. Here, the best result is a cumulative oil production of 163,500 m³ (run 8), which exceeds the reference solution by 7.4 %. In the next set of runs, we apply the formal constraint handling treatment, where lumping is applied for the entire model (“Frm. 1”). For these runs, the best optimum is 160,600 m³ of oil (run 4). This value exceeds the reference solution by 5.5 %, but it is about 2 % less than that achieved using heuristic constraint handling. Finally, we apply the formal lumping-free constraint handling approach and report the results in the fifth column (“Frm. 2”). We see that this approach outperforms the previous two in terms of the objective value achieved. The best optimum 167.0 m³ of oil was essentially achieved from two different initial guesses (run 3 and 5). This is an improvement of 9.7 % over the reference solution and 2.2 % more than the best solution obtained with the heuristic approach.

The oil production profiles for the best runs, along with the reference (heuristic) case, are shown in Fig. 2. Recall that we are maximizing cumulative oil, so the fact that early time production in the reference case exceeds that of the optimized cases is not of concern.

In the convergence of the optimization algorithm, for this case, the formal approach required 49 forward simulations (on average) to converge to the optimal solution, while the heuristic procedure needed only an average of 27 forward simulations. This difference results from the need to enforce

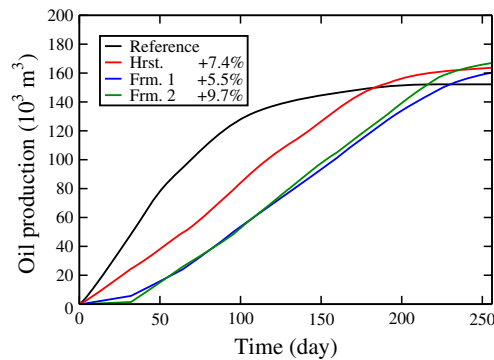


Fig. 2 Oil production versus time (example 1, 40 control variables). Results are for the feasible reference case (*black curve*), best heuristically constrained solution (run 8, *red curve*), best solution from formal lumping-based approach (run 6, *blue curve*), and best solution from formal lumping-free approach (run 2, *green curve*)

feasibility within the optimizer in the formal constraint handling approach. The new formal approach required 42 iterations on average.

6.2 Example 2: twelve-well channelized system.

Our second example uses the three-dimensional geological model introduced by Van Essen et al. [31]. We again consider CO₂ injection. This model contains a total of six components, defined in Table 5. Further details are given in Table 6. A map of the x -component of permeability (here $K_x = K_y = 10K_z$), along with the locations of the wells, is shown in Fig. 3.

The control parameters of our optimization problem are again the well BHPs. The wells are constrained to operate between a lower bound of 90 bar and an upper bound of 120 bar. We also specify nonlinear constraints on both injection and production in the form of maximum gas flow rates of 200,000 m³/d for the injectors and 40,000 m³/d for the producers (both at reservoir conditions). This model is run for a total of 100 days, and we control the BHPs at initial time and then every ten days (the simulation time frame is short in this case because the problem specification is such that oil is produced quickly). There are a total of 120 control parameters in this problem, and our objective is again to maximize cumulative oil production.

We simulate this model using the same procedures as in the previous examples. Results for the nine runs for each case are presented in Table 7. The feasible reference case

Table 6 Model parameters for example 2

Parameter	Value	Units
Grid size	$60 \times 60 \times 7$	–
Δx	24	m
Δy	24	m
Δz	4	m
Depth	2538	m
Initial pressure	100	bar
Temperature	372	°C
Rock compressibility	10^{-5}	1 / bar
Simulation time	300	d
Pressure upper bound	120	bar
Pressure lower bound	90	bar
Residual gas saturation	0	–
Residual oil saturation	0	–
End point rel perm gas	1	–
End point rel perm oil	1	–
Corey exponent gas	2	–
Corey exponent oil	2	–

Well locations [grid block no.]	i	j
Injector 1	5	57
Injector 2	30	53
Injector 3	2	35
Injector 4	27	29
Injector 5	50	35
Injector 6	8	9
Injector 7	32	2
Injector 8	57	6
Producer 1	16	43
Producer 2	35	40
Producer 3	23	16
Producer 4	43	18

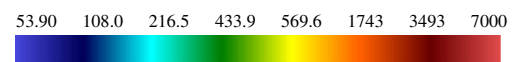


Fig. 3 Reservoir model and wells for example 2 (from [31]). Background shows log K_x

Table 5 Fluid description for example 2

Component	CO ₂	C ₁	C ₂	C ₃	C ₄	C ₁₀
Initial comp. (%)	1	20	30	19	10	20
Injection comp. (%)	95	1	1	1	1	1

Table 7 Oil production in 10^6 m^3 (example 2, 120 control variables) for the optimized objective function without satisfying the nonlinear constraints (“Unc.”), satisfying the nonlinear constraints using the heuristic treatment (“Hrst.”), satisfying the nonlinear constraints using the lumping-based formal approach (“Frm. 1”), and using the lumping-free formal approach (“Frm. 2”)

Run	Unc.	Hrst.	Frm. 1	Frm. 2
Reference	5.030	5.030		
1	5.450	5.449	5.284	5.485
2	5.467	5.457	5.294	5.496
3	5.171	5.171	5.306	5.485
4	5.288	5.287	5.132	5.440
5	5.424	5.423	5.224	5.413
6	5.344	5.348	5.260	5.402
7	5.321	5.230	4.994	5.414
8	5.207	5.205	5.196	5.414
9	5.353	5.349	4.986	5.409
Average	5.336 (25)	5.324 (26)	5.186 (48)	5.440 (29)

Last row shows the average recovery from all nine runs per case followed in parenthesis by the average number of iterations the optimizer needed to converge. Best feasible results shown in bold

yields $5.030 \times 10^6 \text{ m}^3$ of oil, while the best heuristically constrained case (run 2) provides $5.457 \times 10^6 \text{ m}^3$ of oil, an improvement of 8.5 %. The best formally constrained case (run 3) achieves an optimum of $5.306 \times 10^6 \text{ m}^3$ of oil, which exceeds the reference case by 5.5 % but is less than the best heuristic case. The highest objective was achieved by the lumping-free approach where $5.496 \times 10^6 \text{ m}^3$ of oil were produced (run 2), a solution exceeding the reference one by 9.2 %. Moreover, we see that the objective achieved by the lumping-free formal approach exceeds for almost every run (with the exception of run 5) the corresponding solutions obtained by the heuristic treatment, and even the objectives

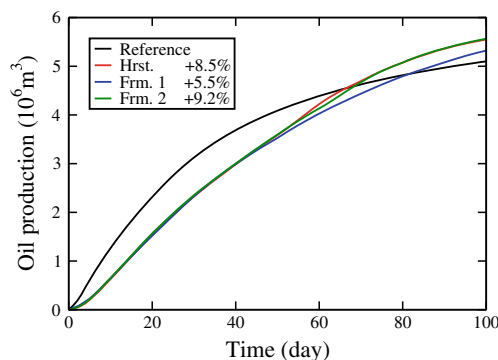


Fig. 4 Oil production versus time (example 2, 120 control variables). Results are for the feasible reference case (black curve), best heuristically constrained solution (run 2, red curve), best solution from formal lumping-based approach (run 3, blue curve), and best solution from formal lumping-free approach (run 2, green curve)

obtained by the unconstrained runs, which are not restricted by the satisfaction of the rate constraints. The oil production profiles for the best runs, along with the feasible reference case, are shown in Fig. 4. We again see that the early time production in the reference case exceeds that of the optimized cases, though the cumulative oil produced in the optimized cases is of course higher.

In this example, convergence of the optimizations using the formal lumping-based constraint handling approach typically required about 48 forward simulations, while the formal lumping-free approach required only 29. The heuristic treatment required about 26. Our findings for this example clearly illustrate the potential advantages of the formal lumping-free approach for complex optimization problems involving multiple wells operating under nonlinear constraints.

6.3 Example 3: Norne model

In our final example, we consider the Norne benchmark problem, which is a model of a real field located at offshore Norway [24]. The actual Norne model involves a three-phase black-oil system. Here, we use the prescribed Norne geological model and well positions (for wells that were operational in January 2005 in the original model). The Norne model contains 29 wells, as shown in Fig. 5, though

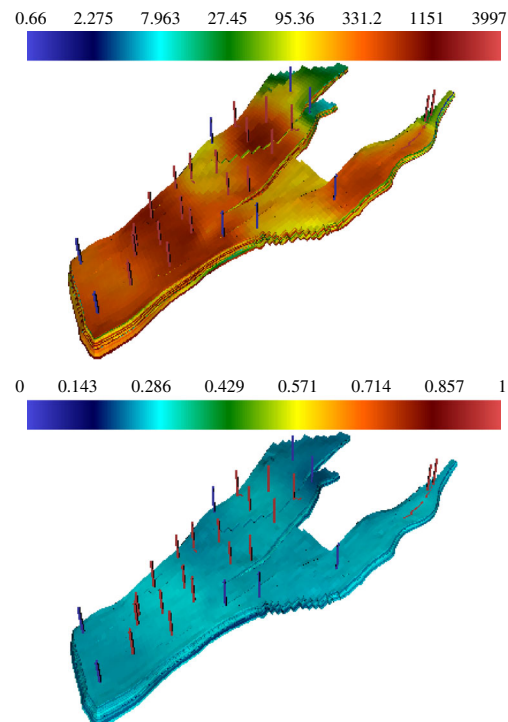


Fig. 5 Injection wells (blue) and production wells (red). Background at top shows the logarithm of the x-component of the permeability tensor, $\log(K_x)$, and at the bottom the porosity, ϕ

Table 8 Fluid description for example 3

Component	CO ₂	NC ₄	C ₈	C ₁	C ₁₅
Initial composition (%)	1	9	40	10	40
Injection composition (%)	90	7	1	1	1

our model involves only 28 of these wells (we do not include the injector C-4H because it does not operate from 2005–2008). Instead of black oil, we consider a five-component compositional system with CO₂ injection (see Table 8 for the fluid description). The model contains a total of 113,344 grid blocks, though only 44,431 of these blocks are active. Other model parameters are provided in Table 9.

The control parameters for the optimization are again the well BHPs, constrained to lie between 50 and 150 bar. The nonlinear constraints are the maximum gas injection rate of 10⁵ m³/day at reservoir conditions for each injector. The simulation is run for 300 days, and we control the BHPs at initial time and then every 30 days thereafter. Because this problem involves 28 wells, there are 280 control parameters. Our objective is to maximize cumulative oil production.

Results for the four sets of runs are reported in Table 10. It is evident from the large differences between the unconstrained runs (second column) and the constrained runs (third and fourth columns) that the consideration of nonlinear constraints significantly restrains oil production. The heuristic approach performs one additional forward simulation, starting from the optimal solution obtained from the unconstrained runs (second column) enforcing the rate

Table 9 Model parameters for example 3

Parameter	Value	Units
Grid size	46 × 112 × 22	–
Δx	24	m
Δy	24	m
Δz	4	m
Depth	3000	m
Initial pressure	100	bar
Temperature	372	°C
Rock compressibility	8 × 10 ^{−5}	1 / bar
Simulation time	300	d
Pressure upper bound	150	bar
Pressure lower bound	50	bar
Residual gas saturation	0	–
Residual oil saturation	0	–
End point rel perm gas	1	–
End point rel perm oil	1	–
Corey exponent gas	2	–
Corey exponent oil	2	–

Table 10 Oil production in 10⁶ m³ (example 3, 280 control variables) for the optimized objective function without satisfying the nonlinear constraints (“Unc.”), satisfying the nonlinear constraints using the heuristic treatment (“Hrst.”), satisfying the nonlinear constraints using the formal lumping-based approach (“Frm. 1”), and the formal lumping-free approach (“Frm. 2”)

Run	Unc.	Hrst.	Frm. 1	Frm. 2
Reference	252.0	142.0		
1	270.2	146.0	138.5	153.1
2	266.7	143.3	138.3	149.5
3	261.6	124.3	129.3	132.6
4	270.3	147.8	137.5	153.5
5	270.7	146.5	138.0	146.7
6	268.8	146.8	137.8	139.2
7	270.4	146.2	129.7	150.8
8	270.8	146.8	129.9	149.5
9	270.6	146.2	136.6	135.2
Average	268.9 (35)	143.8 (36)	135.1 (178)	145.6 (37)

Last row shows the average recovery from all nine runs per case followed in parentheses by the average number of iterations the optimizer needed to converge. Best feasible results shown in bold

constraints whenever they are violated. With the exception of run 3 (third column), the rest of the initial guesses allowed the heuristic approach to converge to a higher maximum, than the reference feasible case. The best maximum obtained was 147.8 × 10⁶ m³ (run 4), an improvement of 4.2 % over the heuristically constrained reference case. This level of improvement is less than that observed for the other examples. We also see that the formal constraint handling approach leads to a result for cumulative oil production (138.5 × 10⁶ m³ in the best case, run 1) that is lower than that for the heuristic reference case, which does not involve any optimization. The results using the formal treatment when lumping is performed for the entire model illustrate the potential challenges that can arise in complex problems with realistic geological configurations, large numbers of control parameters, and many active nonlinear constraints.

On the contrary, the formal lumping-free approach (fifth column), converges to higher optima than the optima obtained with the heuristic approach (run 4), from almost every initial guess with the exception of run 6, and 9 where the heuristic approach obtained higher objectives. We also observe that apart from run 3, run 6, and 9, all other runs converged to an optimal solution of higher objective than the best optimal solution obtained with the heuristic approach (run 4). The best solution of the new formal approach was 153.5 × 10⁶ m³ (run 4), an improvements of 8.1 % over the reference case. The success of the new formal lumping-free approach stems from the discretization of the nonlinear

constraints. Usually, feasibility is violated by one or more wells only at particular time steps and not throughout the simulation. Constraint lumping replaces individual well rates by a single function for the whole time history, and as a result, a descent step suggested by the optimizer aiming to reduce constraint violation, constrains inevitably all well rates involved in the lumping function. On the contrary, discretizing the constraints both per well and per time step allows the optimizer to control the rate of the particular well at the particular time step where feasibility is violated, at the cost of more expensive to compute constraint Jacobian matrices.

The plots of oil production versus time are shown in Fig. 6, and the detailed BHP and gas injection rates (for the injection wells) versus time for the three cases are shown in Figs. 7, 8, and 9. For the reference (Fig. 7), the pressure of the injectors D-3AH, F-2H, and F-4H is kept constant throughout because the corresponding gas rates do not violate the upper bound. For the rest of the wells, the BHP had to be adjusted in order to obtain a feasible solution, however, as we see both BHPs and gas rates are smooth slowly varying functions of time. The best solution obtained with the heuristic approach (Fig. 8) reduces the BHP of wells D-1H, F-3H, and F-4H approximately at the end of the simulation allowing for higher oil production. In the formally constrained lumping-free case (Fig. 9), the BHPs of wells D-1H, D-2H, D-3H, F-1H, and F-3H jump in the beginning simulation, vary slowly, and decrease again gradually close to the end of the simulation causing frequent jumps to the corresponding gas injection rates.

For the results presented here, the optimizations using formal constraint handling required about 178 iterations on average. Optimizations using the heuristic constraint handling, by contrast, required about 36 iterations on average, while the new formal constraint handling approach required

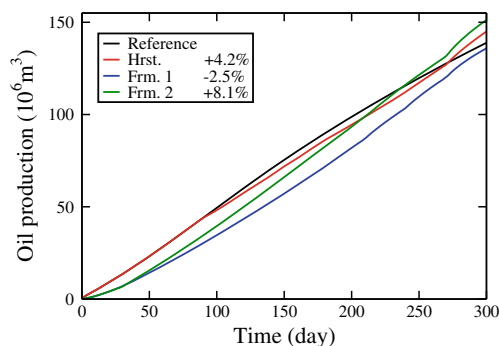


Fig. 6 Oil production versus time (example 3, 280 control variables). Results are for the feasible reference case (black curve), best heuristically constrained solution (run 4, red curve), best solution from formal lumping-based approach (run 3, blue curve), and best solution from formal lumping-free approach (run 2, green curve)

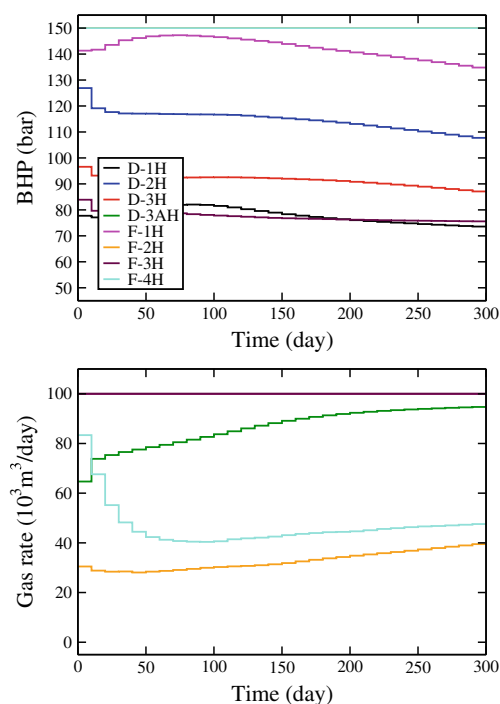


Fig. 7 Injector BHPs (top) and gas injection rates (bottom) for the feasible reference solution (example 3).

on average about 37 iterations. Thus, we again observe significant improvements in terms of convergence using the formal lumping-free approach.

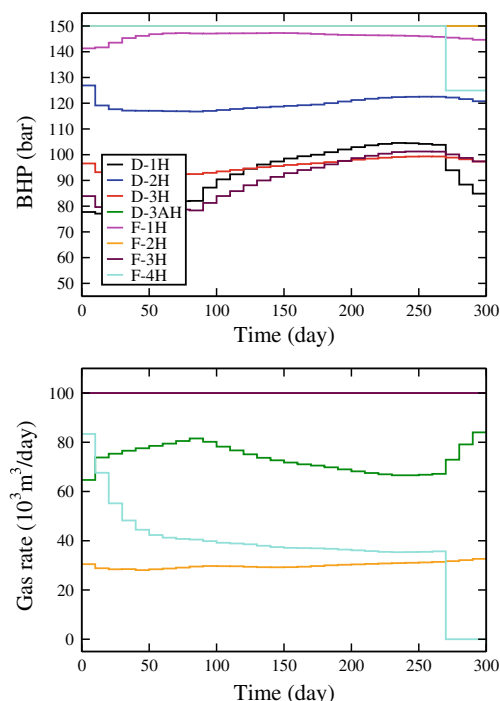


Fig. 8 Injector BHPs (top) and gas injection rates (bottom) for the best heuristically constrained solution (example 3, run 4).

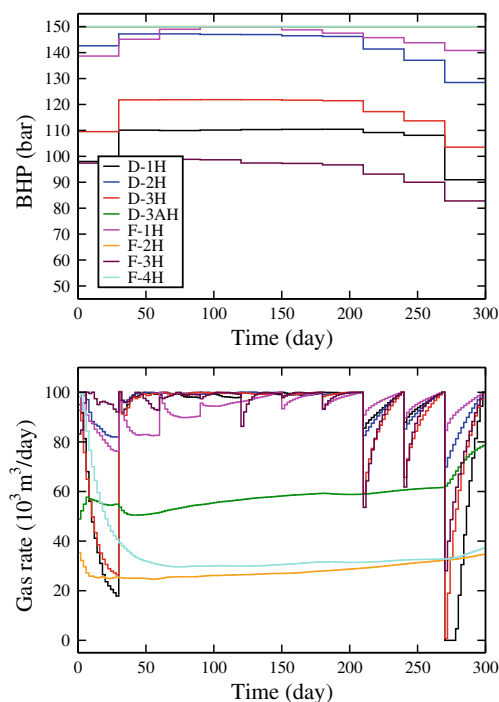


Fig. 9 Injector BHPs (*top*) and gas injection rates (*bottom*) for the best formally constrained solution (example 3, run 1).

7 Concluding remarks

A new formal nonlinear constraint-handling method was suggested for the optimization of compositional reservoir flow and was applied on several benchmark cases including a realistic oil field for maximizing cumulative oil recovery. In contrast to computationally efficient lumping-based techniques where a smooth function is introduced to replace the nondifferentiable max on a per well basis or over the entire model, the new approach relaxes the demands on computational efficiency, discretizing output constraints per well and per time step of the forward simulation. This way, many more constraints are introduced slowing down the evaluation of the constraints Jacobian, but providing more flexibility to the optimizer to restore feasibility, by constraining only the individual rates of particular wells at the particular time steps where the prescribed upper bound is violated. This is in contrast to lumping-based approaches where a reduction suggested by the optimizer propagates through the smoothing function to all well rates even if feasibility is not directly violated there, reducing rates that need to be kept as high as possible in order to increase the oil recovery.

The new approach being theoretically superior than the efficient heuristic one was also demonstrated to converge to higher optima from most of the initial guesses. This is due to the fact that the new formal method optimizes directly in the feasible space in contrast to the heuristic one that optimizes

in the infeasible space at first and through one additional forward simulation it projects the optimal infeasible solution obtained in the first step onto the feasible space. As a result, the best optimal solution found by the new approach almost doubled the increase of the cumulative oil recovery obtained by the heuristic approach over the reference solution, for the realistic benchmark case (example 3, Norne field).

There exist several areas in which future research should be directed. Since numerical evidence shows that better well scheduling techniques than the one applied in the field do exist, the next logical step is to embed the suggested approach in a robust optimization framework to account for geological uncertainty.

It will be of interest to apply the general optimization framework to larger and more realistic simulation models. We realize that for sufficiently fine geological models direct sparse solution methods may become prohibitively expensive and then iterative techniques may be the only possible candidates. Preconditioners based on domain decomposition techniques seem to be particularly attractive for such problems, since they are designed for parallel architectures and the direct sparse solvers used for the inversion of the diagonal blocks can efficiently handle systems with many right-hand sides. These could either replace the entire CPR approach or the preconditioners applied at its two individual steps.

Acknowledgments We would like to thank Dr. Denis Voskov for his assistance with (AD-GPRS), and Prof. Dr. Michael Saunders for his support on SNOPT. We are grateful to the industrial affiliates of the Stanford University Smart Fields Consortium for partial funding of this work and the Swiss National Supercomputing Centre CSCS for providing the computational resources.

References

1. Asheim, H.: Maximization of water sweep efficiency by controlling production and injection rates. In: SPE Paper 18365 Presented at the SPE European Petroleum Conference. London (1988)
2. Aziz, K., Settari, A.: Petroleum Reservoir Simulation. Applied Science Publishers, London (1979)
3. Bertsekas, D.P.: Nonlinear Programming. Athena Scientific (1999)
4. Brouwer, D.R., Jansen, J.D.: Dynamic optimization of water flooding with smart wells using optimal control theory. SPE J. **9**(4), 391–402 (2004)
5. Cao, H.: Development of techniques for general purpose simulators. Ph.D. thesis, Stanford University (2002)
6. Chavent, G., Dupuy, M., Lemonnier, P.: History matching by use of optimal control theory. SPE J. **15**, 74–86 (1975). doi:[10.2118/4627-PA](https://doi.org/10.2118/4627-PA)
7. Chen, C.L., Reynolds, A.: Robust constrained optimization of short- and long-term net present value for closed-loop reservoir management. SPE J. **17**(3), 849–864 (2012)
8. Chen, W., Gavalas, G., Wasserman, M.: A new algorithm for automatic history matching. SPE J. **14**(6), 593–608 (1974)
9. Chen, Y., Oliver, D.: Ensemble-based closed-loop optimization applied to Brugge field. SPE Reserv. Eval. Eng. **13**(1), 56–71 (2010)

10. Coats, K.: An equation of state compositional model. *SPE J.* **20**(5), 363–376 (1980)
11. De Montleau, P., Cominelli, A., Neylon, K., Rowan, D., Pallister, I., Tesaker, O., Nygard, I.: Production optimization under constraints using adjoint gradients. In: 10th European Conference on the Mathematics of Oil Recovery (2006)
12. Doublet, D., Aanonsen, S., Tai, X.: An efficient method for smart well production optimisation. *J. Petroleum Sci. Eng.* **69**(1–2), 25–39 (2009)
13. Echeverría Ciaurri, D., Isebor, O.J., Durllofsky, L.J.: Application of derivative-free methodologies to generally constrained oil production optimisation problems. *Int. J. Math. Modell. Numer. Optim.* **2**, 134–161 (2011)
14. Elman, H., Silvester, D., Wathen, A.: *Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics*. Numerical Mathematics and Scientific Computation, OUP Oxford (2005)
15. Gill, P.E., Murray, W., Saunders, M.A.: SNOPT: An SQP algorithm for large-scale constrained optimization. *SIAM Rev.* **47**(1), 99–131 (2005)
16. Han, C., Wallis, J., Sarma, P., Li, G., Schrader, M.L., Chen, W.: Adaptation of the CPR preconditioner for efficient solution of the adjoint equation. *SPE J.* **18**(2), 207–213 (2013). doi:[10.2118/141300-PA](https://doi.org/10.2118/141300-PA)
17. Jansen, J.D.: Adjoint-based optimization of multi-phase flow through porous media - a review. *Comput. Fluids* **46**(1), 40–51 (2011). doi:[10.1016/j.compfluid.2010.09.039](https://doi.org/10.1016/j.compfluid.2010.09.039)
18. Kourounis, D., Durllofsky, L.J., Jansen, J.D., Aziz, K.: Adjoint formulation and constraint handling for gradient-based optimization of compositional reservoir flow. *Computational Geosciences*, pp. 1–21 (2014). doi:[10.1007/s10596-013-9385-8](https://doi.org/10.1007/s10596-013-9385-8)
19. Kraaijevanger, J., Egberts, P., Valstar, J., Buurman, H.: Optimal waterflood design using the adjoint method. In: Paper SPE 105764 Presented at the SPE Reservoir Simulation Symposium. Houston, USA (2007). doi:[10.2118/105764-MS](https://doi.org/10.2118/105764-MS)
20. Li, R., Reynolds, A., Oliver, D.: History matching of three-phase flow production data. *SPE J.* **8**(4), 328–340 (2003). doi:[10.2118/87336-PA](https://doi.org/10.2118/87336-PA)
21. Lien, M., Brouwer, D., Manseth, T., Jansen, J.D.: Multiscale regularization of flooding optimization for smart field management. *SPE J.* **13**(2), 195–204 (2008)
22. Liu, W., Ramirez, W., Qi, Y.: Optimal control of steam flooding. *SPE Adv. Technol. Series* **1**(2), 73–82 (1993)
23. Mehos, G., Ramirez, W.: Use of optimal control theory to optimize carbon dioxide miscible-flooding enhanced oil recovery. *J. Petroleum Sci. Eng.* **2**(4), 247–260 (1989)
24. NTNU (IO Center): Center of Integrated Operations in Petroleum Industry. Website (2011). <http://www.ipt.ntnu.no/norne>
25. Oliver, D., Reynolds, A., Liu, N.: *Inverse Theory for Petroleum Reservoir Characterization and History Matching*. Cambridge University Press, Cambridge (2008)
26. Ramirez, W.: *Application of Optimal Control Theory to Enhanced Oil Recovery*. Elsevier Science Ltd, Amsterdam (1987)
27. Sarma, P., Chen, W., Durllofsky, L.J., Aziz, K.: Production optimization with adjoint models under nonlinear control-state path inequality constraints. *SPE Reserv. Eval. Eng.* **11**(2), 326–339 (2008)
28. Sarma, P., Durllofsky, 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). doi:[10.1007/s10596-005-9009-z](https://doi.org/10.1007/s10596-005-9009-z)
29. Sudaryanto, B., Yortsos, Y.: Optimization of fluid front dynamics in porous media using rate control. *Phys. Fluids* **12**(7), 1656–1670 (2000)
30. Suwartadi, E., Krogstad, S., Foss, B.: Nonlinear output constraints handling for production optimization of oil reservoirs. *Comput. Geosci.* **16**, 499–517 (2012)
31. Van Essen, G., Zandvliet, M., Van den Hof, P., Bosgra, O., Jansen, J.D.: Robust waterflooding optimization of multiple geological scenarios. *SPE J.* **14**(1), 202–210 (2009). doi:[10.2118/102913-PA](https://doi.org/10.2118/102913-PA)
32. Virnovski, G.: Waterflooding strategy design using optimal control theory. In: 6th European IOR Symposium, pp. 437–446. Stavanger, Norway (1991)
33. Voskov, D., Tchelepi, H.: Comparison of nonlinear formulations for two-phase multi-component EoS based simulation. *J. Petroleum Sci. Eng.* **82–83**, 101–111 (2012). doi:[10.1016/j.petrol.2011.10.012](https://doi.org/10.1016/j.petrol.2011.10.012)
34. Voskov, D.V., Younis, R., Tchelepi, H.A.: Comparison of nonlinear formulations for isothermal compositional flow simulation. In: SPE paper 118966 Presented at the SPE Reservoir Simulation Symposium. The Woodlands (2009)
35. Wang, C., Li, G., Reynolds, A.: Production optimization in closed-loop reservoir management. *SPE J.* **14**(3), 506–523 (2009)
36. Young, L., Stephenson, R.: A generalized compositional approach for reservoir simulation. *J. Petroleum Sci. Eng.* **23**(5), 727–742 (1983). doi:[10.2118/10516-PA](https://doi.org/10.2118/10516-PA)
37. Younis, R., Aziz, K.: Parallel automatically differentiable datatypes for next-generation simulator development. In: SPE Paper 106493 Presented at the SPE Reservoir Simulation Symposium. Houston (2007). doi:[10.2118/106493-MS](https://doi.org/10.2118/106493-MS)
38. Younis, R., Tchelepi, H., Aziz, K.: Adaptively localized continuation-Newton method–Nonlinear solvers that converge all the time. *SPE J.* **15**(2), 526–544 (2010). doi:[10.2118/119147-PA](https://doi.org/10.2118/119147-PA). SPE-119147-PA