

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΓΕΝΙΚΟ ΤΜΗΜΑ



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΕΦΑΡΜΟΣΜΕΝΕΣ ΕΠΙΣΤΗΜΕΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑ

**ΔΙΠΛΩΜΑΤΙΚΗ ΔΙΑΤΡΙΒΗ ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ
ΚΑΤΕΥΘΥΝΣΗ : «ΕΦΑΡΜΟΣΜΕΝΑ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΑ ΜΑΘΗΜΑΤΙΚΑ»**

**ΕΦΑΡΜΟΓΗ ΤΗΣ ΜΕΘΟΔΟΥ ΧΡΟΝΟΣΕΙΡΩΝ ΣΤΗΝ ΠΡΟΒΛΕΨΗ
ΠΑΡΑΓΩΓΗΣ ΠΕΤΡΕΛΑΙΟΥ**

ΠΑΠΑΔΟΠΟΥΛΟΥ ΓΕΩΡΓΙΑ

Επιβλέπων: Επίκουρος Καθηγητής Δάρας Τρύφων

ΧΑΝΙΑ, 2015



petroleum engineering
MSc course

MASTER OF SCIENCE
IN PETROLEUM ENGINEERING

PRODUCTION OPTIMIZATION THROUGH
WATER-FRONT CONTROL USING ADJOINT
GRADIENT-BASED TECHNIQUES

MASTER THESIS

CHRISTINI FANDRIDI
Technical University of Crete
Department of Mineral Resources Engineering

Chania 19 January 2017

PRINCIPAL ADVISOR: Dr. Drosos Kourounis

EXAMINATION COMMITTEE

Prof. Nikolaos Pasadakis

Prof. Dionysios Hristopoulos

Dr. Vassileios Gaganis

To my family

Contents

I Oil production 1

1	Oil production methods	3
1.1	Oil production methods	4
1.2	Primary recovery	4
1.3	Secondary recovery	5
1.4	Enhanced oil recovery	6
1.4.1	Thermal techniques	6
1.4.2	Non thermal techniques	7
1.4.3	Nitrogen Injection	8
1.5	Need for simulation and optimization	10
1.6	Previous work	10
1.7	The scope of this work	12

II Theory on production optimization and methods 13

2	Production Optimization	15
2.1	Gradient-free methods	15
2.1.1	Genetic algorithms (GAS)	15
2.1.2	Particle Swarm Optimization(PSO)	16
2.2	Gradient-based methods	16
2.2.1	Sequential quadratic programming (SQP)	16
2.2.2	Interior point optimization (IPO)	17
2.3	Oil-gas compositional simulation equations	18
2.4	Adjoint equations for the compositional system	19
2.4.1	Automatic differentiation	19
2.4.2	Discrete adjoint formulation	20
2.4.3	Continuous adjoint formulation	22

2.4.4	Continuous versus discrete adjoint formulation	24
2.4.5	Solution of adjoint equations	25
2.5	Gradient-based optimization and related software	26
2.6	Nonlinear constraints	27
2.6.1	Constraint handling in the optimizer	28
2.6.2	Constraint handling in the simulator	31

III Numerical examples?? 35

3 Numerical results 37

3.0.1	Example 1 - Π obstacle	39
3.0.2	Example 2 - Top layer of SPE 10 model	44
3.0.3	Example 3: Twelve-well channelized system	46
3.1	Concluding remarks	48

IV Numerical examples 61

Bibliography 63

Abstract

The optimization of oil production is a tedious and computationally intensive process that requires the solution of time dependent nonlinear set of partial differential equations describing the flow of hydrocarbons in anisotropic porous media. Optimization of production is usually performed using either gradient free techniques like genetic algorithms, particle swarm algorithms, or gradient-based techniques where the gradients are computed through the solution of the adjoint problem. A gradient-based optimization method, in which the gradient is computed using an adjoint formulation, is often the method of choice since in contrast to numerical perturbation techniques that require as many objective function evaluations as the number of control parameters, the gradient using adjoint-based techniques is obtained only at a small fraction of the time spent for the evaluation of the objective function. The production optimisation problem is usually subject to industrial constraints, for example, maximum gas rate specification in injection or production wells, when the control variables are well bottom-hole pressures. At the same time optimising time-varying well settings, such as injection rates, production rates or bottom-hole pressures, is an important aspect of optimal reservoir management that increases significantly the dimension of the search space. It is well known that for non-convex optimisation problems, gradient-based techniques are likely to get trapped in poor local optima. A common practise is to lunch several independent optimisation runs from different initial guesses or to combine ideas from gradient-free algorithms with gradient-based to benefit from the merits of both. An adequate sampling of the search space would require an intractable number of simulations and it is thus impossible.

The aim of this work is to exploit an observation in homogeneous reservoirs, where the global optimum, when optimising cumulative oil recovery, is usually achieved from any initial guess. We perform continuation with respect to a parameter that transforms the homogenous reservoir with respect to porosity and permeability, to the current one, solving an optimal control problem for each distinct value of the parameter, where the objective function now is defined to be the misfit between a subset of the states of the two problems. The proposed technique

is tested in several examples of increased complexity.

Part I

Oil production

Chapter 1

Oil production methods

Need for energy

Modern industrial societies consume large quantities of energy in order to maintain today's high standard of living. Most of that energy comes as a result of the technology of oil production. According to the U.S. Energy Information Administrations (EIA) International Energy Outlook 2016, the global supply of crude oil, other liquid hydrocarbons, and biofuels is expected to be adequate to meet the world's demand for liquid fuels through 2040. There is substantial uncertainty about the levels of future liquid fuels supply and demand. According to current prognosis, oil production in matured reservoirs is expected to decline and this could create gap between supply and demand of hydrocarbons in various parts of the world. To counterpart this growing demand-supply discrepancy, the petroleum industry will have to give more attention to their mature fields to sustain current production levels. It should be realized that most operators have not exploited the full capacity of mature fields to their potential. In addition to this they face the challenge of developing green fields in such a way that they can be produced to their maximum potential in the future. With a mean recovery factor of about 36%, there is an immense opportunity for "production optimization".

The primary objective within reservoir management is to provide optimal production scenarios, accompanied by estimates of expected hydrocarbon recovery, ultimately resulting in an optimized field development plan. Elements of such a plan include recovery techniques, well types and position or pattern designs, completion types, and production scheduling. In fields with significant complexity, automated workflows based on numerical algorithms will need to be used to find optimal choices for all these variables.

1.1 Oil production methods

During the life of a producing oil field, several production stages are encountered. Initially, when a field is brought into production, oil flows naturally to the surface due to current reservoir pressure in the primary stage. As reservoir pressure drops, water is typically injected to boost the pressure, so that it displaces the oil in the so called “secondary” stage. Lastly, the remaining oil can be recovered by a variety of methods such as CO₂ injection, natural gas miscible injection, and steam recovery in a tertiary or enhanced oil recovery (EOR) phase [Meyer, 2007].

1.2 Primary recovery

Glover [Glover, 2001] explained all recovery methods, including primary recovery mechanism as it is the stage when the natural energy of the reservoir is used to transport hydrocarbons towards and out of the production wells. The earliest possible determination of the drive mechanism is a primary goal in the early life of the reservoir, as its knowledge can greatly improve the management and recovery of reserves from the reservoir in its middle and later life. There are five important drive mechanisms: (i) Solution gas drive; (ii) Gas cap drive; (iii) Water drive; (iv) Gravity drainage; (v) Combination or mixed drive. All these mechanisms maintain the reservoir pressure, though water drive maintains much higher pressure than the gas drive mechanisms (Figure 1.2).

Solution gas drive

In solution gas drive, the expansion of the dissolved gases in the oil and water provides most of the reservoirs drive energy. Oil recovery from this type is typically between 20% and 30% of original oil in place.

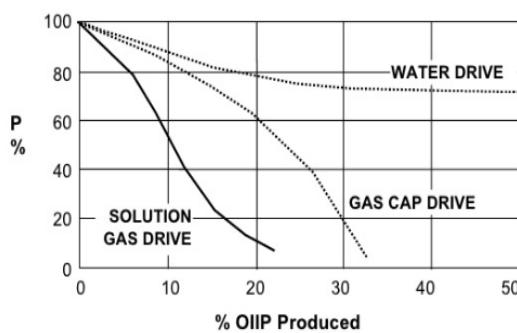


Figure 1.1: Pressure trends under various drive mechanisms.

Gas cap drive

As production continues, the gas cap expands pushing the gas-oil contact (GOC) downwards. Eventually the GOC will reach the production wells and the gas oil ratio (GOR) will increase by large amounts. The recovery of gas cap reservoirs can be (20% to 40% OOIP).

Water drive

The drive energy is provided by an aquifer that interfaces with the oil in the reservoir at the oil-water contact (OWC). The recovery from water driven reservoirs is usually good (20-60% OOIP).

Gravity drainage

Gravity drainage is the fourth drive force that might be considered for drive mechanism where the density differences between oil and gas and water result in their natural segregation in the reservoir. This process is relatively weak and in practice is only used in combination with other drive mechanisms.

1.3 Secondary recovery

After initial discover and production, typical oil reservoirs lose the drive mechanism of gas or water that originally forced the oil to the surface. The second stage of hydrocarbon production in which an external fluid such as water: usually named water flooding or water injection or gas: referred to as gas flooding or gas injection, is injected into the reservoir through injection wells. By secondary recovery methods, another 15-20% may be produced. [Fleshman and Lekic, 1999].

Water flooding

Water Flooding is implemented by injecting water into a set of wells while producing from the surrounding wells. Water flooding projects are generally implemented to accomplish reservoir pressure maintenance and as a water drive to displace oil from the injector wells to the producer wells[Fleshman and Lekic, 1999].

Gas flooding

This method is similar to water flooding in principal, and is used to maintain gas cap pressure even if oil displacement is not required. Usually the produced natural gas is re-injected to the reservoir in order to maintain reservoir pressure rather than to displace the hydrocarbon.

6 · OIL PRODUCTION METHODS

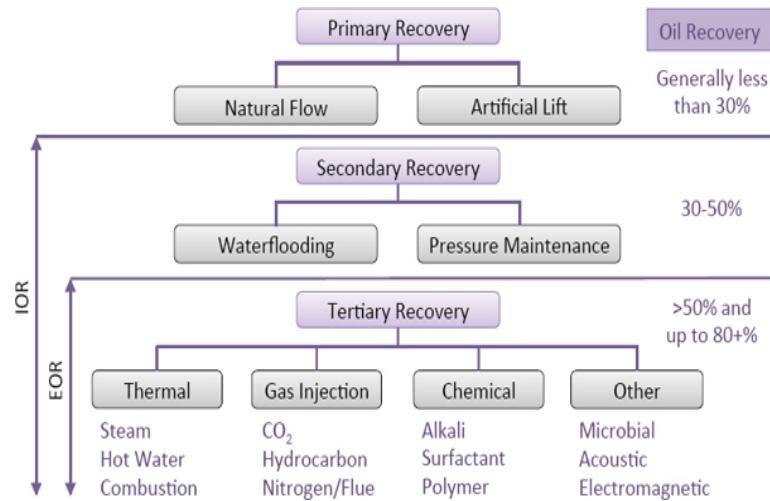


Figure 1.2: The different oil recovery stages and the corresponding oil recovery factor.

1.4 Enhanced oil recovery

Enhanced oil recovery techniques refer to the recovery of oil through the injection of fluids and energy not normally present in the reservoir [Romero-Zeran, 2012]. The objectives of the injected fluids are to achieve mainly two purposes; First is to boost the natural energy in the reservoir; second is to interact with the reservoir rock/oil system to create conditions favourable for residual oil recovery that leads to reduce the interfacial tension between the displacing fluid and oil, increase the capillary number, reduce capillary forces, increase the drive water viscosity, provide mobility-control, create oil swelling, reduce oil viscosity, alter the wettability of reservoir rock [Romero-Zeran, 2012]. Enhanced oil recovery can be divided into two: thermal and non-thermal recovery [Anazi, 2007]. Fig. 2 illustrates oil recovery stages by the different EOR techniques.

1.4.1 Thermal techniques

Thermal methods raise the temperature of the reservoir to heat the crude oil in the formation and therefore reduce its viscosity and/or vaporise part of the oil and thereby decrease the mobility ratio. The increase in heat reduces the surface tension, increases the permeability of the oil and improves the reservoir seepage conditions. The heated oil may also vaporise and then condense to be produced. This operation, however, requires substantial investment in special equipment.

In-situ combustion (ISC)

In-situ combustion or fire flooding is a process in which an oxygen containing gas is injected into a reservoir where it reacts with the oil contained within the pore space to create a high temperature self-sustaining combustion front that is propagated through the reservoir.

Steam Injection

Steam is injected into the reservoir either continuously or in cycles. Steam floods are easier to control than in-situ combustion. For the same pattern size, the response time is 25-50% lower than the response time for additional production by in-situ combustion [et al., 2010].

Hot water flooding

Water-flooding in heavy oils is generally not an efficient way of production due to high viscosity of heavy oil compared to water. In hot water-flooding, thermal energy will increase oil mobility, and possibly improve sweep efficiency [Kermen, 2011].

1.4.2 Non thermal techniques

Chemical Flooding

These processes use chemicals added to water in the injected fluid of a water flood to alter the flood efficiency in such a way as to improve oil recovery by: (i) Increasing water viscosity (polymer floods) (ii) Decreasing the relative permeability to water (cross-linked polymer floods) (iii) Increasing the relative permeability to oil (micellar and alkaline floods) [Glover, 2001]

Polymer flooding

Polymers improve both vertical and areal sweep efficiency by reducing water-oil ratio. Polymers are injected through water injection wells in order to displace the residual oil. Increasing the displacing fluids viscosity and lowering its relative permeability through plugging [Anazi, 2007].

Micellar polymer flooding

It is well known that water and oil cannot be mixed until the third component, surfactant or soap, is added to reduce the interfacial tension between oil and water. Since micellar solution makes fluids miscible in the reservoir, almost 100% of oil can be displaced especially in the presence of alkaline. However, due to reservoir rock non-uniformity in the field, the amount of oil recovered is reduced.

Alkaline-surfactant-polymer (ASP) flooding

During waterflooding residual oil is trapped due to low water viscosity and high water-oil interfacial tension, therefore another way is to inject the three chemicals; alkaline to minimize surface adsorption; surfactant to lower interfacial tension and stabilizes the emulsion. On the other hand, polymer is used to increase viscosity and to improve mobility control and sweep efficiency[et al., 2011].

1.4.3 Nitrogen Injection

Nitrogen itself is an inert gas that gets miscible at very high pressure and efficiently reduces the oil viscosity while providing efficient miscible displacement [Syed et al., 2011]. Nitrogen can be used for the following enhanced oil recovery applications:

Nitrogen immiscible flooding

Gas is being injected into the crest of the structure to maintain the pressure, to recover the hydrocarbon liquids in the gas cap, and to stabilize the gas/oil contact.

Nitrogen miscibility displacement mechanism

There are three types of miscibility including; First contact miscibility; Multi- contact miscibility; Vaporizing mass-transfer miscibility [Shine and Holtz, 2008].

Multi-contact miscibility

This type of miscibility is subdivided into vaporizing gas drive, condensing gas drive[Juttner, 1997].

Gas Flooding Injection

Gas is generally injected single or intermittently with water and this manner of injection called Water-Alternating-Gas (WAG), has become widely practiced over all of worlds oil fields [Kulkarni, 2003]. According to miscibility between gas injected and oil displaced, gas injection can be classified into two major types: miscible gas injection and immiscible gas injection. In miscible gas injection, the gas is injected at or above minimum miscibility pressure (MMP) which causes the gas to be miscible in the oil. In contrast in immiscible gas injection, flooding by the gas is conducted below MMP. This low pressure injection of gas is used to maintain reservoir pressure to prevent production cut-off and thereby increase the rate of production [Anazi, 2007]. In miscible flooding, the incremental oil recovery is obtained by one of the three mechanisms: oil displacement by solvent

through the generation of miscibility (i.e. zero interfacial tension between oil and solvent hence infinite capillary number), oil swelling, and reduction in oil viscosity [Kulkarni, 2003]. Miscible fluids are 100% soluble in each other. The interfacial tension between miscible fluids is zero. Injection gases include:

LPG injection

Miscible LPG products such as ethane, propane, or butane have first contact miscibility, which means they will be miscible from the first contact with oil. However, LPGs are in such demand as marketable commodity that their use in EOR is limited [Romero-Zeran, 2012]. In particular, this process uses a slug of propane or other liquified petroleum gas (2 to 5% of pore volume PV) followed by natural gas, inert gas, and/or water. Thus, the solvent will bank oil and water ahead, and fully displace all contacted oil [Anazi, 2007].

Enriched gas miscible process

In this process, a slug of methane (C_1) enriched with ethane (C_2), propane (C_3), or butane (C_4) (10% to 20% of the PV) and followed by lean gas and/or water is injected from water injection well into the reservoir. When the injected gas contacts virgin reservoir oil, C_1-C_3 are quenched from the injected gas and absorbed into the oil [Anazi, 2007]. The injected HC solvent is usually displaced with cheaper chase leaner or inert gas like Methane or Nitrogen. At reservoir conditions the most usual problem occurs with the hydrocarbon miscible flood is the gravity over-ride because of its lighter density than the oil and water. So that in any miscible flood the Minimum Miscibility Pressure (MMP) plays the most major role to overcome this problem. As a remedial factor the solvent is to be injected at or above the MMP of the reservoir fluid. Once it becomes miscible then it improves the sweep efficiency and fallouts in optimum recovery [Syed et al., 2011].

Carbon dioxide (CO_2) injection

Is one of the most proven of these methods. Almost pure CO_2 (>95% of the overall composition) has the property of mixing with the oil to swell it, make it lighter, detach it from the rock surfaces, and causing the oil to flow more freely within the reservoir so that it can be swept up in the flow from injector well to producer well [Melzer and Midland, 2012]. Flooding a reservoir with CO_2 can occur either miscibility or immiscibly. Miscible CO_2 displacement is only achieved under a specific combination of conditions, which are set by four variables: reservoir temperature, reservoir pressure, injected gas composition, and oil chemical composition. From a fundamental point of view, CO_2 EOR works on a very simple principle, namely, that given the right physical conditions, CO_2 will mix miscibly with

oil, acting much like a thinning agent, the same way that gasoline does with motor oil. After miscible mixing, the fluid is displaced by a chase phase, typically water [Meyer, 2007]. In this thesis CO₂ injection as EOR method is going to be examined.

1.5 Need for simulation and optimization

Use of reservoir simulation has grown because of its ability to predict the future performance of oil and gas reservoirs over a wide range of operating conditions. Reservoir simulators use numerical methods and high-speed computers to model multidimensional fluid flow in reservoir rock. Technology improvements have enabled a widespread use of integrated simulation models for a better asset management to be fully combined with measured field data. Reliable simulators and adequate computing capacity are available to most reservoir engineers, so simulation is usually practical for all reservoir sizes and all types of reservoir performance studies. Although the use of simulation frequently is optional, it may be the only reliable way to predict the performance of a large, complex reservoir, especially if such external considerations as government regulations influence the production schedule. Even for small reservoirs where simple calculations or extrapolations may be adequate, simulation is often faster, cheaper, and more reliable than alternative methods for predicting performance.[Mattax and Dalton, 1990]

In petroleum fields, hydrocarbon production is often constrained by reservoir conditions, deliverability of the pipeline network, fluid handling capacity of surface facilities, safety and economic considerations, or a combination of these considerations. Optimization of reservoir development requires many evaluations of the possible combinations of the decision variables in order to obtain the best economical strategies. The objective of dynamic production optimization is to find the best operational settings at a given time, subject to all constraints, this gives you greater production gains for longer. Overall, optimization delivers a faster return on investment during initial production, yields greater revenues during plateau and decline, and delays well abandonment. Given the fact that oil prices continue to drop to their lowest levels in several years, oil industry will inevitably turn to optimization in order to continue to deliver the dividend levels that investors have come to expect. Production optimization is no longer an option it is a necessity.

1.6 Previous work

In petroleum industry, optimization methods are necessary for history matching, where we adjust the physical properties of the reservoir model, and for optimization of production, where the objective is to maximize either the net present value or the cumulative production of hydrocarbons. All the above methods can be implemented both with gradient-free and gradient-based techniques. Generally, gradient-free techniques are not necessarily guaranteed to find the true global optimal solution, they converge very slowly and require high performance computing infrastructures. On the other hand, for gradient-based techniques, once the gradient is computed there are several options for finding an optimum. Furthermore, proper exploitation of gradient information can significantly enhance the speed of convergence in comparison with a method that does not compute gradients. Another feature of gradient-based methods is that they provide a clear convergence criterion.

Among gradient-based algorithms we consider only the adjoint approach for compositional reservoir simulation problems. Procedures of this type entail the application of optimal control theory and have their roots in the calculus of variations [Bryson and Ho, 2001; Stengel, 1986]. Adjoint-based optimization techniques have been used in a reservoir simulation setting both for history matching (see, e.g., [Chavent et al., 1975; Chen et al., 1974; Li et al., 2003; Oliver et al., 2008; Sarma et al., 2006]) 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 [Liu et al., 1993; Mehos and Ramirez, 1989; Ramirez, 1987]. In subsequent work, the focus was on gradient-based optimization (and in some cases on the optimization of ‘smart wells’) for water flooding [Asheim, 1988; Brouwer and Jansen, 2004; Sarma et al., 2006; Sudaryanto and Yortsos, 2000; Virnovski, 1991]. Refer to [Jansen, 2011] for a more complete overview of adjoint-based optimization methods. We note additionally that, although not considered here, derivative-free methods can also be applied for production optimization problems – see [Echeverría Ciaurri et al., 2011] for discussion and examples. Recently an adjoint treatment for multicomponent oil-gas compositional systems was presented in [Kourounis et al., 2014]. 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 [Kourounis and Schenk, 2015] the author introduces 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 [Kourounis et al., 2014] for several different examples of increased complexity.

1.7 The scope of this work

Optimization using gradients converges much faster than gradient-free techniques resulting in significant saving in computational time but it usually gets trapped to poor local optima [Kourounis et al., 2014; Kourounis and Schenk, 2015].

The aim of this work is to exploit an observation in homogeneous reservoirs, where the global optimum, when optimising cumulative oil recovery, is usually achieved from any initial guess. We perform continuation with respect to a parameter that transforms the homogenous reservoir with respect to porosity and permeability, gradually to the original inhomogeneous one, solving an optimal control problem for each distinct value of the parameter. This approach allows us to follow the optimal solution (cumulative recovery or ...?residual oil?...) as the geology switches from homogeneous to inhomogenous. This novel technique is presented at the best of our knowledge for first time for production optimization problems and tested in several examples of increased complexity.

Part II

Theory on production optimization and methods

Chapter 2

Production Optimization

The optimization of oil production is a tedious and computational intensive process that requires the solution of time dependent nonlinear set of partial differential equations describing the flow of hydrocarbons in anisotropic porous media. The optimization of time-varying well settings, such as injection rates, production rates or bottom-hole pressures, is an important aspect of optimal reservoir management. Optimization of production is usually performed using either gradient free techniques like genetic, particle swarm algorithms, or gradient-based techniques where the gradients are computed through the solution of the adjoint problem.

2.1 Gradient-free methods

In this section we review the basic gradient-free method employed in industry and academia for the solution of several optimal control problems of the oil industry.

2.1.1 Genetic algorithms (GAS)

Genetic algorithms are commonly used to generate high-quality solutions to optimization and search problems by relying on bio-inspired operators such as mutation, crossover and selection. The method is a general one, capable of being applied to an extremely wide range of problems. Genetic algorithms are based on three essential components:

- Survival of the fittest (Selection).

- Reproduction processes where genetic traits are propagated (Crossover).
- Variation (Mutation).

2.1.2 Particle Swarm Optimization(PSO)

Particle swarm optimization (PSO) is a stochastic, population-based computer algorithm. It applies the concept of swarm intelligence (SI) to problem solving. Swarm intelligence is the property of a system whereby the collective behaviors of (unsophisticated) agents interacting locally with their environment cause coherent functional global patterns to emerge (e.g. self-organization, emergent behavior). Gradient-free methods are not necessarily guaranteed to find the true global optimal solutions, but they are able to find many good solutions.

2.2 Gradient-based methods

Gradient-based optimization methods offer the advantage to construct additional information about the shape of the surface for the particular problem. Hence, the gradient of a function provides information about the behavior of a function such as steepness and extrema in the parameter space. The gradients are computed through the solution of the adjoint problem. With this additional information, the convergence of the search algorithm can be drastically enhanced.

2.2.1 Sequential quadratic programming (SQP)

Sequential quadratic programming (SQP) is an iterative procedure that utilizes a 2.order approximation of the Lagrangian function of a problem. The quadratic formulation of the problem is a local approximation of the real problem and consists of a quadratic objective function and linear equality and/or inequality constraints. SQP can be used both within a trust-region and a line search framework. In a line search framework, the algorithm proceeds by first calculating a search direction. If we are trying to maximize the original problem, a function is then solved that maximize the quadratic approximation in the search direction. When a new iteration point in the direction that was searched has been reached, a new local approximation is constructed and the algorithm proceeds to the next iteration given that a set of optimality conditions has not been fulfilled. SQP is a generalization of Newtons method for unconstrained problems as it uses a quadratic approximation of the Lagrangian function, steps in a direction it believes the optimum lies,

and then creates a new approximation of the original model when a new iteration point has been reached. The main difference between Newtons method and SQP is that for constrained nonlinear problems the Taylor approximation of the original problem cannot be used, as the model problem also needs to incorporate the constraints of the original problem. Instead, the Lagrangian function is used and the constraints of the problem thereby taken into account. SQP is appropriate for small and large problems and it is well-suited to solving problems with significant nonlinearities[Bonnans et al., 2006].

2.2.2 Interior point optimization (IPO)

An interior point method is a linear or nonlinear programming method that achieves optimization by going through the middle of the solid defined by the problem rather than around its surface. This method is further discussed in section....

Both gradient-based and derivative-free methods have been considered for this problem, and both are applicable in different situations. A gradient-based optimization method, in which the gradient is computed using an adjoint formulation, is often the method of choice since in contrast to numerical perturbation techniques that require as many objective function evaluations as the number of control parameters, the gradient using adjoint-based techniques is obtained only at a small fraction of the time spent for the evaluation of the objective function. The development of adjoint procedures for general compositional flow problems is much 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. Furthermore, the production optimisation problem is usually subject to industrial constraints, for example, maximum gas rate specification in injection or production wells, when the control variables are well bottom-hole pressures. At the same time optimising time-varying well settings, such as injection rates, production rates or bottom-hole pressures, is an important aspect of optimal reservoir management that increases significantly the dimension of the search space. It is well known that for non-convex optimisation problems, gradient-based techniques are likely get trapped in poor local optima. A common practise is to lunch several independent optimisation from different initial guesses or to combine ideas from gradient-free algorithms with gradient-based to benefits from the merits of both. An adequate sampling of the search space would require an intractable number of simulations and it is thus impossi-

ble.

2.3 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 Cao [2002]; Voskov and Tchelepi [2012]; Voskov et al. [2009]:

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

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 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 2.6.2. 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:

$$\sum_{i=1}^{n_c} x_{i0} - 1 = 0, \quad \sum_{i=1}^{n_c} x_{ig} - 1 = 0. \quad (2.3)$$

A capillary pressure relationship also appears in cases with nonzero capillary pressure, though here we neglect capillary pressure so $p_o = p_g$.

As discussed by many authors (see, e.g., Cao [2002]; Coats [1980]; Voskov and Tchelepi [2012]; Young and Stephenson [1983]), 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 Voskov and Tchelepi [2012] for 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.

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 Aziz and Settari [1979]. 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 Younis et al. [2010]. 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 Han et al. [2013]. 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.

2.4 Adjoint equations for the compositional system

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

2.4.1 Automatic differentiation

It is quite common for comprehensive computational platforms, in reservoir simulation and other application areas, to undergo frequent modification and enhance-

ment. This poses a problem for adjoint formulations because, when an existing feature is modified the corresponding adjoint code may also be impacted, and when a new feature is added, the associated adjoint code must (in many cases) be written. The maintenance and development of adjoint code poses challenges because the necessary derivatives are generally complicated. This is particularly the case in compositional simulation where variables couple in many ways, including through the nonlinear equation of state.

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 Voskov et al. [2009]. 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 Younis and Aziz [2007]. This library generates efficient computer code for the evaluation of the Jacobian matrix and the corresponding partial derivatives from discrete algebraic expressions of the governing conservation equations, associated constraint relations, and equations of state. We refer to Younis and Aziz [2007] for a detailed description of the underlying theory.

2.4.2 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:

$$\mathbf{g}_n(\mathbf{x}_n, \mathbf{x}_{n-1}, \mathbf{u}_n) = \mathbf{0}, \quad (2.4)$$

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 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:

$$J(\mathbf{x}, \mathbf{u}) = \int_{t_0}^{t_N} f(\mathbf{x}(t), \mathbf{u}(t)) dt + \varphi(\mathbf{x}(t_N)), \quad (2.5)$$

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

$$J \approx \sum_{n=1}^N \Delta t_n f_n(\mathbf{x}_n, \mathbf{u}_n) + \varphi(\mathbf{x}_N). \quad (2.6)$$

Using (2.6) we can state the optimal control problem as:

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

In general, a number of linear and nonlinear constraints may need to be included in the optimal control problem. We postpone the discussion of their treatment until Section 2.6. 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:

$$J_A = \sum_{n=1}^N \left(\Delta t_n f_n(\mathbf{x}_n, \mathbf{u}_n) + \boldsymbol{\lambda}_n^T \mathbf{g}_n(\mathbf{x}_n, \mathbf{x}_{n-1}, \mathbf{u}_n) \right) + \varphi(\mathbf{x}_N). \quad (2.7)$$

In (2.7), 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:

$$\begin{aligned} \delta J_A = & \left(\frac{\partial \varphi_N}{\partial \mathbf{x}_N} + \Delta t_N \frac{\partial f_N}{\partial \mathbf{x}_N} + \boldsymbol{\lambda}_N^T \frac{\partial \mathbf{g}_N}{\partial \mathbf{x}_N} \right) \delta \mathbf{x}_N \\ & + \sum_{n=1}^{N-1} \left(\Delta t_n \frac{\partial f_n}{\partial \mathbf{x}_n} + \boldsymbol{\lambda}_{n+1}^T \frac{\partial \mathbf{g}_{n+1}}{\partial \mathbf{x}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{x}_n} \right) \delta \mathbf{x}_n \\ & + \sum_{n=1}^N \left(\Delta t_n \frac{\partial f_n}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n} \right) \delta \mathbf{u}_n. \end{aligned} \quad (2.8)$$

In order to achieve $\delta J_A = 0$, we require $\delta J_A / \delta \mathbf{x}_n = \mathbf{0}^T$ (for $n = 1, 2, \dots, N$) and $\delta J_A / \delta \mathbf{u}_n = \mathbf{0}^T$. To satisfy $\delta J_A / \delta \mathbf{x}_n = \mathbf{0}^T$ for $n = 1, 2, \dots, N$, we require that the Lagrange multipliers satisfy the following equations:

$$\frac{\partial \mathbf{g}_n^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_n = - \left(\frac{\partial \mathbf{g}_{n+1}^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_{n+1} + \Delta t_n \frac{\partial f_n^T}{\partial \mathbf{x}_n} \right), \quad (2.9)$$

$$\frac{\partial \mathbf{g}_N^T}{\partial \mathbf{x}_N} \boldsymbol{\lambda}_N = - \left(\Delta t_N \frac{\partial f_N^T}{\partial \mathbf{x}_N} + \frac{\partial \varphi_N^T}{\partial \mathbf{x}_N} \right). \quad (2.10)$$

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

$$\frac{\delta J_A}{\delta \mathbf{u}} = \left[\frac{\delta f_1}{\delta \mathbf{u}_1}, \frac{\delta f_2}{\delta \mathbf{u}_2}, \dots, \frac{\delta f_N}{\delta \mathbf{u}_N} \right]. \quad (2.11)$$

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

$$\frac{\delta f_n}{\delta \mathbf{u}_n} = \Delta t_n \frac{\partial f_n}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n}, \quad n = 1, 2, \dots, N. \quad (2.12)$$

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, are 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 .

2.4.3 Continuous adjoint formulation

The continuous adjoint formulation employs the continuous representation of the objective function along with the spatially discretized reservoir flow equations.

The optimal control problem can then be stated as:

$$\begin{aligned} & \underset{\mathbf{u}}{\text{minimize}} \quad J(\mathbf{x}, \mathbf{u}) = \int_{t_0}^{t_N} f(\mathbf{x}(t), \mathbf{u}(t)) dt + \varphi(\mathbf{x}(t_N)) \\ & \text{subject to } \mathbf{g}(\dot{\mathbf{x}}(t), \mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0}. \end{aligned}$$

In this case we express the governing set of partial differential equations, for a specified dynamic well-control strategy $\mathbf{u}(t)$, as $\mathbf{g}(\dot{\mathbf{x}}(t), \mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0}$. We introduce the Lagrange multipliers $\boldsymbol{\lambda}(t)$ and define the Lagrangian L as:

$$L(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}) = f(\mathbf{x}, \mathbf{u}) + \boldsymbol{\lambda}^T \mathbf{g}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}), \quad (2.13)$$

The variables $\mathbf{u}(t)$, $\mathbf{x}(t)$, $\dot{\mathbf{x}}(t)$ and $\boldsymbol{\lambda}(t)$ are denoted as \mathbf{u} , \mathbf{x} , $\dot{\mathbf{x}}$ and $\boldsymbol{\lambda}$ to simplify notation. The augmented objective function, J_A , can be expressed as:

$$J_A(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}) = \int_{t_0}^{t_N} L(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}) dt + \varphi(\mathbf{x}_N). \quad (2.14)$$

The first variation of J_A is given by

$$\begin{aligned} \delta J_A = & \int_{t_0}^{t_N} \left(\frac{\partial L}{\partial \dot{\mathbf{x}}} \delta \dot{\mathbf{x}} + \frac{\partial L}{\partial \mathbf{x}} \delta \mathbf{x} + \frac{\partial L}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial L}{\partial \boldsymbol{\lambda}} \delta \boldsymbol{\lambda} \right) dt \\ & + \frac{\partial \varphi(\mathbf{x}_N)}{\partial \mathbf{x}_N} \delta \mathbf{x}_N. \end{aligned} \quad (2.15)$$

Note that $\delta \dot{\mathbf{x}} = d(\delta \mathbf{x})/dt$, so any variation in the state vector \mathbf{x} will introduce a variation in its time derivative $\dot{\mathbf{x}}$.

After integration by parts, using the fact that the variation of the initial conditions $\delta \mathbf{x}_0 = \mathbf{0}$, and taking into account that $\partial L^T / \partial \boldsymbol{\lambda} = \mathbf{g}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{u}) = \mathbf{0}$, the first variation of J_A can be written as:

$$\begin{aligned} \delta J_A = & \int_{t_0}^{t_N} \left(\frac{\partial L}{\partial \mathbf{x}} - \frac{d}{dt} \frac{\partial L}{\partial \dot{\mathbf{x}}} \right) \delta \mathbf{x} dt \\ & + \left(\frac{\partial L(\mathbf{x}_N)}{\partial \dot{\mathbf{x}}_N} + \frac{\partial \varphi(\mathbf{x}_N)}{\partial \mathbf{x}_N} \right) \delta \mathbf{x}_N \\ & + \int_{t_0}^{t_N} \frac{\partial L}{\partial \mathbf{u}} \delta \mathbf{u} dt. \end{aligned} \quad (2.16)$$

To achieve $\delta J_A / \delta \mathbf{x} = \mathbf{0}$, $\boldsymbol{\lambda}$ must be chosen to satisfy the following:

$$\frac{d}{dt} \left(\frac{\partial \mathbf{g}^T}{\partial \dot{\mathbf{x}}} \boldsymbol{\lambda} \right) - \frac{\partial \mathbf{g}^T}{\partial \mathbf{x}} \boldsymbol{\lambda} - \frac{\partial f^T}{\partial \mathbf{x}} = \mathbf{0} \quad (2.17)$$

$$\frac{\partial \mathbf{g}_N^T}{\partial \dot{\mathbf{x}}_N} \boldsymbol{\lambda}_N = - \frac{\partial \varphi^T(\mathbf{x}_N)}{\partial \mathbf{x}_N}. \quad (2.18)$$

The ordinary differential equation in (2.17) is integrated backwards in time starting from the final time condition (2.18). With the resulting $\boldsymbol{\lambda}$, the first variation of the objective function becomes:

$$\delta J_A = \int_{t_0}^{t_N} \left(\frac{\partial f}{\partial \mathbf{u}} + \boldsymbol{\lambda}^T \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \right) \delta \mathbf{u} dt. \quad (2.19)$$

In order to allow a direct comparison of the discrete and continuous adjoint formulations, we integrate the continuous adjoint backwards in time fully implicitly, using the same scheme as is applied for the forward problem. The discrete form of (2.17) is:

$$\frac{\partial \mathbf{g}_n^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_n = - \frac{\partial \mathbf{g}_{n+1}^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_{n+1} - \Delta t_n \frac{\partial f_n^T}{\partial \mathbf{x}_n}. \quad (2.20)$$

This equation is solved backwards in time, starting from the boundary condition (2.18). Once the Lagrange multipliers have been obtained, the gradient is computed using equations (2.11) and (2.12).

2.4.4 Continuous versus discrete adjoint formulation

The gradients obtained by the discrete adjoint formulation are, as would be expected, fully consistent with the discrete forward problem. Indeed, if we compute the gradients using numerical perturbation of the controls, we find that they coincide with those from the discrete adjoint solution in the first 5-8 significant digits (to achieve this level of agreement, tight tolerances must be used for linear and nonlinear convergence of the forward simulation). There are differences, however, between these gradients and those provided by the continuous adjoint formulation.

To illustrate this, consider the simplified case where $\varphi(\mathbf{x}_N) = 0$. The solution of (2.18) in this case will give $\boldsymbol{\lambda}_N = \mathbf{0}$, and as a result, the first term in (2.8) will not vanish. There will be a nonzero term left multiplying the variation of $\delta \mathbf{x}_N$:

$$\delta J_A = \Delta t_N \frac{\partial f_N}{\partial \mathbf{x}_N} \delta \mathbf{x}_N + \sum_{n=1}^N \left(\Delta t_n \frac{\partial f_n}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n} \right) \delta \mathbf{u}_n. \quad (2.21)$$

It is evident that, as the time step size $\Delta t_N \rightarrow 0$, the term multiplying $\delta \mathbf{x}_N$ will vanish, and the gradient provided by the continuous formulation will become consistent with that from the discrete problem. However, as long as Δt_N is significant, the two gradients will not coincide, especially at the last time step.

We implemented both the continuous and discrete adjoint formulations into our optimization framework. Using a small Δt_N , we observed that the computed gradients were very similar, consistent with (2.21). Even using small Δt_N , however, we did not observe any advantage of the continuous formulation over the discrete formulation. In cases where Δt_N was not small, the continuous formulation required more iterations of the optimizer, presumably because of errors in $\delta J_A / \delta \mathbf{u}$. In light of these observations, we do not present any detailed results using the continuous adjoint formulation since we do not see any advantages to this approach for our problem. We note that these findings are consistent with those reported in Hager [2000] and Walther [2007] for general Runge-Kutta time stepping methods, in Nadarajah and Jameson [2000], where the discrete and continuous adjoint approaches were applied to automatic aerodynamic optimization, and in Petra and Stadler [2011] for general variational inverse problems governed by partial differential equations. A similar gradient discrepancy between discretization/optimization versus optimization/discretization can also occur with respect to the spatial discretization – see the discussion for a shape optimization problem in Gunzburger [2003].

2.4.5 Solution of adjoint equations

The solution of the linear system of equations that arises when solving (2.9) 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 equations (2.9), (2.10) and (2.12). 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 (2.9), we use GMRES preconditioned by the transpose of the CPR (constrained pressure residual) preconditioner, as described in Han et al. [2013]. In these linear solutions, 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 is required for the forward problem.

2.5 Gradient-based optimization and related software

The SNOPT optimizer is used in this work for solving the nonlinear constrained optimization problem, and in this section we will provide a concise overview of the underlying theory. This discussion loosely follows Gill et al. [2005], which provides a much more in-depth description. 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.

We now discuss the main features of the SQP method used to solve our nonlinear program (NP). This discussion follows Gill et al. [2005]. As will be discussed in detail in Section 2.6.1, all realistic recovery optimization problems involve bound constraints on the inputs, and often several other linear or nonlinear equality or inequality constraints, which together constitute a so-called general nonlinear program (GNP). All features described in this section are readily specialized to the GNP discussed in Section 2.6.1. Here we take the problem to be

$$\begin{aligned} (\text{NP}) \quad & \underset{\mathbf{u}}{\text{minimize}} \quad f(\mathbf{u}) \\ & \text{subject to } \mathbf{c}(\mathbf{u}) \geq 0, \end{aligned}$$

where $\mathbf{u} \in \mathbb{R}^n$, $\mathbf{c} \in \mathbb{R}^m$, while the objective function $f(\mathbf{u})$ and the constraints $c_i(\mathbf{u})$, $i = 1, 2, \dots, m$ have continuous second derivatives. The gradient of f is denoted by the vector $\nabla f(\mathbf{u})$, and the gradients of each element of \mathbf{c} form the rows of the Jacobian matrix $\mathbf{J}(\mathbf{u})$.

An SQP method obtains search directions (for the primal \mathbf{u} and dual variables $\boldsymbol{\pi}$) from an iterative sequence of QP subproblems. Each QP subproblem, in turn, iteratively minimizes a convex quadratic model of a certain Lagrangian function subject to linearized constraints associated with (NP), namely,

$$\mathcal{L}(\mathbf{u}, \mathbf{u}_k, \boldsymbol{\pi}_k) = f(\mathbf{u}) - \boldsymbol{\pi}_k^T \mathbf{d}_L(\mathbf{u}, \mathbf{u}_k), \quad (2.22)$$

defined in terms of the constraint linearization $\mathbf{c}_L(\mathbf{u}, \mathbf{u}_k)$ and the departure from linearity $\mathbf{d}_L(\mathbf{u}, \mathbf{u}_k)$:

$$\begin{aligned} \mathbf{c}_L(\mathbf{u}, \mathbf{u}_k) &= \mathbf{c}_k + \mathbf{J}_k(\mathbf{u} - \mathbf{u}_k), \\ \mathbf{d}_L(\mathbf{u}, \mathbf{u}_k) &= \mathbf{c}(\mathbf{u}) - \mathbf{c}_L(\mathbf{u}, \mathbf{u}_k), \end{aligned}$$

subject to linearized constraints. In this formulation k is the SQP (major) iteration counter, and π are Lagrangian multipliers to adjoin \mathbf{d}_L to f . The first and second derivatives of the modified Lagrangian with respect to \mathbf{u} are

$$\begin{aligned}\nabla \mathcal{L}(\mathbf{u}, \mathbf{u}_k, \boldsymbol{\pi}_k) &= \nabla f(\mathbf{u}) - (\mathbf{J}(\mathbf{u}) - \mathbf{J}_k)^T \boldsymbol{\pi}_k, \\ \nabla^2 \mathcal{L}(\mathbf{u}, \mathbf{u}_k, \boldsymbol{\pi}_k) &= \nabla^2 f(\mathbf{u}) - \sum_i (\boldsymbol{\pi}_k)_i \nabla^2 c_i(\mathbf{u}).\end{aligned}$$

Observe that $\nabla^2 \mathcal{L}$ is independent of \mathbf{u}_k and is the same as the Hessian of the conventional Lagrangian. At $\mathbf{u} = \mathbf{u}_k$, i.e., after convergence of the inner iterations, the modified Lagrangian has the same function and gradient values as the objective: $\mathcal{L}(\mathbf{u}_k, \mathbf{u}_k, \boldsymbol{\pi}_k) = f_k$, $\nabla \mathcal{L}(\mathbf{u}_k, \mathbf{u}_k, \boldsymbol{\pi}_k) = \nabla f_k$. The modified augmented Lagrangian is ‘less’ nonlinear than the augmented Lagrangian itself because linear terms in the constraints disappear, especially in the quadratic penalty term. The number of nonlinear variables in the modified augmented Lagrangian is the same as in the original problem.

The merit function

$$\mathcal{M}_\rho(\mathbf{u}, \boldsymbol{\pi}, \mathbf{s}) = f(\mathbf{u}) - \boldsymbol{\pi}^T (\mathbf{c}(\mathbf{u}) - \mathbf{s}) + \frac{1}{2} \sum_{i=1}^m \rho_i (c_i(\mathbf{u}) - s_i)^2, \quad (2.23)$$

where ρ and \mathbf{s} are vectors of penalty parameters and slack variables, respectively, is reduced along each search direction to ensure convergence from any starting point.

In summary, the basic structure of an SQP method involves major and minor iterations. The major iterations generate a sequence of iterates $(\mathbf{u}_k, \boldsymbol{\pi}_k)$ that converge to the optimal solution $(\mathbf{u}^*, \boldsymbol{\pi}^*)$. At each major iterate a QP subproblem is solved to generate a search direction towards the next iterate $(\mathbf{u}_{k+1}, \boldsymbol{\pi}_{k+1})$. 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.

2.6 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 Brouwer and Jansen [2004]; Van Essen G. et al. [2009]; Virnovski [1991]; Wang et al. [2009], and

more systematic approaches, valid for a broader range of constraint equations Chen and Oliver [2010]; De Montleau et al. [2006]; Kraaijevanger et al. [2007]; Sarma et al. [2008]; Suwartadi et al. [2012]. 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 bottom-hole pressure (bottom-hole pressure, or BHP, is the wellbore pressure at a specified depth within the reservoir). 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 both a formal constraint handling approach and a simpler heuristic procedure for satisfying rate constraints, which can be used in compressible systems.

2.6.1 Constraint handling in the optimizer

In the presence of nonlinear output inequality constraints, it can be difficult to determine an initial guess for the control parameters that does not violate these constraints. In such cases the optimizer must apply a process to find a feasible set of control parameters. Some optimizers may also intentionally exit the feasible region, in an attempt to (eventually) find better optima. For these reasons, constraint and infeasibility handling are key architectural components of any general purpose optimizer. Here we described how they are handled within SNOPT.

To simplify notation, we first write the constrained optimization problem in the more general form (GNP) assuming lower and upper bounds on the controls and the nonlinear constraints

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

where $f(\mathbf{u})$ is 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.

Achieving feasibility

SNOPT deals with infeasibility using ℓ_1 penalty functions. It solves (GNP) as given, using QP subproblems based on linearizations of the nonlinear constraints. If a QP subproblem proves to be infeasible or unbounded (or if the Lagrange multiplier estimates for the nonlinear constraints become large), SNOPT reverts to a technique known as ‘elastic programming’ to solve the problem

$$\begin{aligned} (\text{GNP}(\gamma)) \quad & \underset{\mathbf{u}, \mathbf{v}, \mathbf{w}}{\text{minimize}} \quad f(\mathbf{u}) + \gamma \mathbf{e}^T (\mathbf{v} + \mathbf{w}) \\ & \text{subject to } \mathbf{b}_l \leq \begin{pmatrix} \mathbf{u} \\ \mathbf{c}(\mathbf{u}) - \mathbf{v} + \mathbf{w} \end{pmatrix} \leq \mathbf{b}_u, \\ & \mathbf{v} \geq 0, \quad \mathbf{w} \geq 0, \end{aligned}$$

where \mathbf{e} is a vector containing all ones and \mathbf{v} and \mathbf{w} are ‘elastic variables’ which allow the constraints to stretch in directions that increase the size of the feasible region. The resistance to stretching is obtained by minimizing the composite objective function $f(\mathbf{u}) + \gamma \mathbf{e}^T (\mathbf{v} + \mathbf{w})$, where the penalty parameter γ ($\gamma \geq 0$) may take a finite sequence of increasing values. If (GNP) has a feasible solution and γ is sufficiently large, the solutions to (GNP) and (GNP(γ)) are identical. If (GNP) has no feasible solution, (GNP(γ)) will tend to determine a ‘reasonable’ infeasible point if γ is again sufficiently large. If γ were infinite, the nonlinear constraint violations would be minimized subject to the linear constraints and bounds.

This process may require many major iterations until a feasible initial set of control parameters is obtained. For this reason, optimization with simple bound constraints on the inputs usually converges much faster than optimization with additional bounds on output quantities.

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 of this are the specification of maximum gas injection or production rates (either for an individual well or for a group of wells) in a general compositional problem where the control variables are BHPs. A variety of methods have been proposed to incorporate nonlinear constraints in the adjoint formulation; see, e.g., Chen and Oliver [2010]; Jansen [2011]; Sarma et al. [2006]; Suwartadi et al. [2012] for detailed discussion.

As indicated above, operational constraints in reservoir simulation are often inequality constraints specified for input and/or output rates at every time step. In some cases these constraints must be satisfied by many, or by all, of the wells in the problem. This means that the vector $\mathbf{c}(\mathbf{u})$ in (GNP) 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.

A more efficient way of approximating these gradients is to ‘lump’ the constraints over the full simulation time frame Sarma et al. [2008]. 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 gradient, or over the entire model, in which case only $O(N)$ linear systems must be solved. 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. 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.

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.,

$$q_{jn} \leq q_{max}, \quad (2.24)$$

where q_{jn} is the rate of the well j at time step n and q_{max} is a specified maximum rate. To satisfy this constraint we must guarantee that

$$\max_{j,n} (q_{jn}) \leq q_{max}. \quad (2.25)$$

However, the max function is not differentiable so it cannot be used to provide gradient information. We thus approximate it by a smooth function, specifically one suggested by Bertsekas [1999]:

$$c = \max_{j,n} (q_{jn}) \approx \alpha \log Q, \\ Q = \sum_{j=1}^{N_w} \sum_{n=1}^N e^{q_{jn}/\alpha}, \quad (2.26)$$

$$\alpha = 0.05q_{max}. \quad (2.27)$$

With this definition of the constraint function, the gradient of the constraint with respect to the control variables, which is required by the optimizer, is given by:

$$\begin{aligned}\frac{\delta c}{\delta \mathbf{u}_n} &= \frac{1}{Q} \sum_{j=1}^{N_w} e^{q_{jn}/\alpha} \frac{\partial q_{jn}}{\partial \mathbf{u}_n} + \boldsymbol{\lambda}_n^T \frac{\partial \mathbf{g}_n}{\partial \mathbf{u}_n}, \\ \frac{\partial q_{jn}}{\partial \mathbf{u}_n} &= \left[0, 0, \dots, \frac{\partial q_{jn}}{\partial u_{jn}}, \dots, 0 \right].\end{aligned}\quad (2.28)$$

The Lagrange multipliers for the constraints $\boldsymbol{\lambda}_n$ are computed from the solution of the following adjoint problem:

$$\frac{\partial \mathbf{g}_n^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_n = -\frac{\partial \mathbf{g}_{n+1}^T}{\partial \mathbf{x}_n} \boldsymbol{\lambda}_{n+1} + \frac{1}{Q} \sum_{j=1}^{N_w} e^{q_{jn}/\alpha} \frac{\partial q_{jn}^T}{\partial \mathbf{x}_n}, \quad (2.29)$$

$$\frac{\partial \mathbf{g}_N^T}{\partial \mathbf{x}_N} \boldsymbol{\lambda}_N = -\frac{1}{Q} \sum_{j=1}^{N_w} e^{q_{jN}/\alpha} \frac{\partial q_{jN}^T}{\partial \mathbf{x}_N}. \quad (2.30)$$

The smaller the coefficient multiplying q_{max} in (2.26), 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 (2.26) 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.

2.6.2 Constraint handling in the simulator

A simpler way to render an infeasible solution feasible, when the constraints are upper and lower bounds on output quantities (e.g., rates when BHPs are specified), is to satisfy these constraints in the forward model. This is often readily accomplished because, during the forward simulation, wells can be controlled by specifying either the bottom-hole pressure or the rate. In the former case, the well rate q_j^w for phase j in block l is calculated from the well model, which (subject to some simplifications) is of the general form

$$(q_j^w)_l = \left(T_w \frac{k_{rj}}{\mu_j} \right)_l (p_l - p_{w,l}), \quad (2.31)$$

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 well rate is specified, equations

of the form of (2.31) 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 Cao [2002].

This flexibility in defining the controls of the wells allows some constraints to be easily satisfied during the forward simulation. For example, 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. This treatment is illustrated in Fig. 2.1. The BHP (top left) is prescribed to be 50 bar throughout the simulation, but we also have a maximum gas rate of 5,000 m³/d at reservoir conditions (indicated by the blue line in the upper right figure). The well operates at 50 bar for the first 480 days, at which point the maximum rate constraint is reached. The well then switches to operate at a gas rate of 5,000 m³/d for the rest of the simulation. The resulting BHP and gas production rate profiles actually used in the simulation (and which honor the constraints) are shown in the bottom two figures.

In the results below, we apply this treatment and compare it to the formal approach described in Section 2.6.1. Using this procedure, which we refer to as ‘heuristic constraint handling’, we first 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.

In our implementation of the heuristic constraint handling procedure, we also allow wells to switch back to BHP control (after they have switched from BHP control to rate control). The switch in control is accomplished, in either case, by checking (at each Newton iteration) to see if the current active control is violated. This assessment can be performed efficiently through use of the well equation. If the well does switch back to BHP control, it remains on BHP control unless (and until) the rate constraint is again violated. This process generally increases the number of Newton iterations required for convergence, though in most of the cases considered the total simulation time is within a factor of two of that for a typical forward simulation run (and, as noted above, the run using heuristic constraint handling is only performed once).

Although it is clearly approximate, this heuristic constraint handling approach has some potential advantages over the formal method described in Section 2.6.1. For example, the heuristic treatment allows the simulator to switch controls at any time step in the simulation, while the formal approach only allows controls to

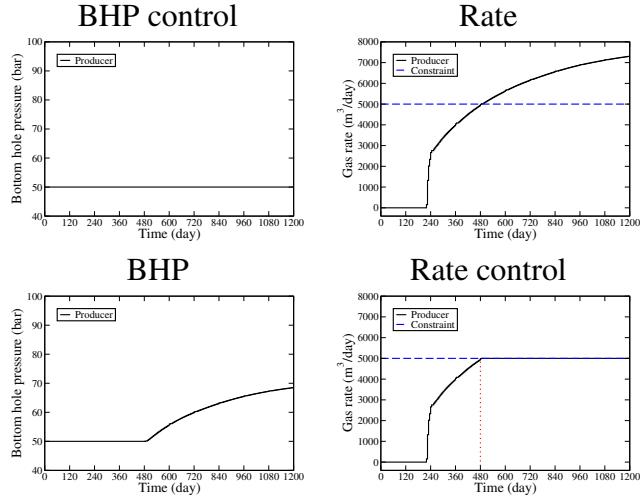


Figure 2.1: Schematic illustrating heuristic constraint handling. Top: Constant BHP and resulting gas rate. Bottom: BHP and gas rate satisfying constraint.

switch at a relatively small number of control steps (by way of comparison, in a typical problem we may have $O(10^2 - 10^3)$ time steps but only $O(10)$ control steps). The heuristic approach thus enables, in some sense, a more ‘fine-grained’ response, and it can be viewed as having many more ‘control’ variables (though these variables are not optimized formally). Increasing the number of control steps to provide the same granularity to the optimizer as in the forward problem (i.e., setting the control step size to equal the time step size) should theoretically result in better performance by the formal approach, though in practice the large increase in the number of control variables would result in a more difficult optimization problem, which could negatively impact the performance of the optimizer. In the examples below, we will compare the performance of these two approaches for handling nonlinear constraints.

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.

Part III

Numerical examples??

Chapter 3

Numerical results

We implement an adjoint treatment for multicomponent oil-gas compositional systems through use of a recently developed automatic differentiation capability [Younis et al., 2010]. The application of automatic differentiation in the context of Stanford’s General Purpose Research Simulator (AD-GPRS) [Cao, 2002], a modular simulator with many advanced features, enables us to construct a gradient-based optimization framework suitable for use in compositional problems. Our formulation includes the treatment of bound, linear and nonlinear constraints. In a discrete implementation, the governing equations for the so-called adjoint system are constructed based on the discretized-in-time forward model equations.

We will present results for four different cases. All involve bound and nonlinear constraints, and we will compare the performance of the two approaches described above for treating the nonlinear constraints. Because our gradient-based optimization will find only a local optimum, we run each case nine times, using a different initial guess 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 $p^l = p_{init} + 1$ bar for the injectors and $p^u = p_{init} - 1$ 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 3.1.

Table 3.1: Resulted objective values for every continuity step with respect to permeability for all benchmarks.

Run	SPE10TOP 1000D	SPE10TOP 3000D	Pi 256D
1	$5.443892E + 05$	$6.642702E + 05$	$1.252661E + 05$
2	$5.631894E + 05$	$6.706731E + 05$	$1.252819E + 05$
3	$5.873230E + 05$	$6.756695E + 05$	$1.244174E + 05$
4	$5.986247E + 05$	$6.763321E + 05$	$1.242821E + 05$
5	$5.998848E + 05$	$6.760098E + 05$	$1.241929E + 05$
6	$5.962259E + 05$	$6.782397E + 05$	$1.240130E + 05$
7	$5.909467E + 05$	$6.762526E + 05$	$1.242663E + 05$
8	$5.850723E + 05$	$6.724442E + 05$	$1.241064E + 05$
9	$5.775845E + 05$	$6.667619E + 05$	$1.240303E + 05$
10	$5.688990E + 05$	$6.584114E + 05$	$1.234992E + 05$
11	$5.594653E + 05$	$6.501344E + 05$	$1.229465E + 05$
12	$5.507783E + 05$	$6.427540E + 05$	$1.229158E + 05$
13	$5.422065E + 05$	$6.365637E + 05$	$1.226925E + 05$
14	$5.339263E + 05$	$[p_I^a, p_P^a]$	$1.225814E + 05$
15	$5.256504E + 05$	$[p_I^a, p_P^a]$	$1.221619E + 05$
16	$5.180703E + 05$	$[p_I^a, p_P^a]$	$1.217791E + 05$
17	$5.120160E + 05$	$[p_I^a, p_P^a]$	$1.214983E + 05$
18	$5.071502E + 05$	$[p_I^a, p_P^a]$	$1.212952E + 05$
19	$5.044022E + 05$	$[p_I^a, p_P^a]$	$1.211731E + 05$
20	$5.033963E + 05$	$[p_I^a, p_P^a]$	$1.211338E + 05$
20*	$5.089974E + 05$	$6.018453E + 05$	$1.188555E + 05$

Table 3.2: Fluid description for Example 1

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

3.0.1 Example 1 - Π obstacle

In the first example we maximize cumulative oil recovery under CO₂ injection. The two-dimensional geological model is depicted in Fig. 3.2. A Π -shaped region is located at the center of a homogeneous reservoir. The model is discretized on a grid of dimensions 80×80 . The permeability in most of the domain (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. In all of our examples we describe the permeability with a diagonal tensor: $\mathbf{K} = \text{diag}(\mathbf{k}_x, \mathbf{k}_y, \mathbf{k}_z)$; here, in addition, the permeability is isotropic and uniform within each of the regions. Four injection wells are placed at the corners of the model, and the single production well is located inside the Π -shaped region. The model includes a total of four components (three hydrocarbon components plus CO₂), as specified in Table 3.2. Further details on the reservoir model are provided in Table 3.3.

The control parameters in the optimization problem are the well BHPs. These are constrained to lie between an upper bound of 120 bar and a lower bound of 90 bar. We additionally specify a maximum (per-well) gas injection rate of 500 m³/d at reservoir conditions. The total simulation period is 256 days, and the well controls are determined at initial time and for every subsequent 32-day interval. There are thus a total of eight control steps and 40 control parameters.

Two reference solutions are generated. First, we run the simulation with the production wells operating at the minimum BHP and the injection wells at the maximum BHP. This solution is infeasible because it violates the nonlinear output constraints (maximum gas injection rate of 500 m³/d). Next, we apply the heuristic constraint handling approach described above, with the maximum gas injection rate set to 500 m³/d. The cumulative oil production for these two cases is given in the first row ('Reference') of Table 3.5. 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 3.5 in the column labeled 'Unconstr.' The best optimum achieved is a cumulative oil production of 190,200 m³, obtained in Run 7. This clearly exceeds the unconstrained reference result of 163,900 m³. Results using

40 · NUMERICAL RESULTS

Table 3.3: Model parameters for example 1.

Parameter	Value	Units
Grid size	$80 \times 80 \times 1$	—
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	j	
Injector 1	1	1
Injector 2	1	80
Injector 3	80	1
Injector 4	80	80
Producer 1	40	48

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 feasible reference solution (152,200 m³) by 7.4%. In the next set of runs we apply the formal constraint handling treatment. For these runs, the best optimum is 160,600 m³ of oil (Run 4). This value exceeds the feasible reference solution by 5.5%, but it is about 2% less than that achieved using heuristic constraint handling. We will show below that results using the formal procedure can be improved through use of more control steps.

The oil production profiles for the best runs, along with the reference (heuristic) case, are shown in Fig. 3.3. 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. The detailed BHP and gas rate profiles for each case are shown in Figs. 3.4, 3.5 and 3.6. The oil rates for all three cases are depicted in Fig. 3.7. Although the BHPs for the two optimized cases are clearly different, the oil rate profiles do show some general similarity. For example, they both show less variation in oil rate over the course of the simulation than the reference case.

It is important to note that the heuristic constraint handling approach is more efficient computationally than the formal treatment. In terms of computational requirements, 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 significant difference results from the need to enforce feasibility within the optimizer in the formal constraint handling approach.

We now assess the impact of using additional control variables. Theoretically, if the optimization problem remains sufficiently ‘tractable’, as we increase the number of control variables the formal approach should (eventually) outperform the heuristic approach. However, if the optimization problem becomes significantly more difficult with increasing numbers of control variables (which may be related to the constraint lumping procedure), or if a large number of local optima associated with relatively poor objective function values appear, then the formal approach will not necessarily outperform the heuristic treatment.

To test the performance of our procedures, we now consider optimizations with 64 control steps, which corresponds to 320 control variables (the results above used eight control steps and 40 control variables). Results for this case are presented in Table 3.6. The best result using heuristic constraint handling provides cumulative oil production of 159,400 m³ (Run 7). This is slightly lower than that achieved using 40 controls, which presumably reflects the fact that this is a more difficult optimization problem. Using the formal constraint handling approach, however, we achieve cumulative oil production of 170,200 m³ (Run 1). This exceeds the

42 · NUMERICAL RESULTS

RUN	UNCONSTR	HEURISTIC	FORMAL
0	163.9	152.2	–
1	187.5	156.6	158.2
2	189.1	162	146.2
3	177.3	149	149.2
4	183.4	150.2	160.6
5	186.1	152.2	152.9
6	185	158.9	160.2
7	190.2	162.3	142.5
8	190.1	163.5	158.5
9	190.1	162	158

Table 3.4: Oil production in 10^3 m^3 (Example 1, **40 control variables**) for the optimized objective function without satisfying the nonlinear constraints ('Unconstr.'), satisfying the nonlinear constraints using the heuristic treatment ('Heuristic'), and satisfying the nonlinear constraints using the formal approach ('Formal'). Best feasible results shown in bold.

feasible reference solution ($152,500 \text{ m}^3$) by 11.6%, which represents a substantial improvement. It also exceeds the best solution found using heuristic constraint handling ($163,500 \text{ m}^3$ in Run 8, using 40 controls) by 4.1%. In fact, three of the nine local optima achieved in this case using formal constraint handling exceed the best result obtained using heuristic constraint handling. These findings suggest that, for this problem, the formal approach does indeed outperform the heuristic approach given a sufficient number of control variables.

Finally, it is worth noting that the spread in the results from run to run for (optimized) cumulative oil is larger with 320 control variables than it is with 40 control variables. In fact, with 320 control variables, optimizations using both constraint handling procedures lead to some local optima that are below the corresponding lowest optima achieved using 40 control variables (Runs 3 and 5 for the heuristic approach; Runs 2, 3 and 4 for the formal approach). Again, we believe this is indicative of the challenges associated with performing constrained optimization with increasing numbers of control variables. These results suggest that it may be useful to explore the use of a sequence of optimizations, with increasing numbers of control periods, for production optimization problems.

Table 3.5: Oil production in 10^3 m^3 (Example 1, **40 control variables**) for the optimized objective function without satisfying the nonlinear constraints ('Unconstr.'), satisfying the nonlinear constraints using the heuristic treatment ('Heuristic'), and satisfying the nonlinear constraints using the formal approach ('Formal'). Best feasible results shown in bold.

Run	Unconstr.	Heuristic	Formal
Reference	163.9	152.2	
1	187.5	156.6	158.2
2	189.1	162.0	146.2
3	177.3	149.0	149.2
4	183.4	150.2	160.6
5	186.1	152.2	152.9
6	185.0	158.9	160.2
7	190.2	162.3	142.5
8	190.1	163.5	158.5
9	190.1	162.0	158.0

Table 3.6: Oil production in 10^3 m^3 (Example 1, **320 control variables**) for the optimized objective function without satisfying the nonlinear constraints ('Unconstr.'), satisfying the nonlinear constraints using the heuristic treatment ('Heuristic'), and satisfying the nonlinear constraints using the formal approach ('Formal'). Best feasible results shown in bold.

Run	Unconstr.	Heuristic	Formal
Reference	150.1	152.5	
1	188.1	149.9	170.2
2	192.4	155.6	131.4
3	186.5	136.4	117.8
4	186.8	149.9	133.7
5	186.5	142.2	156.1
6	192.5	157.8	168.2
7	192.5	159.4	158.8
8	192.4	156.9	161.1
9	192.2	157.0	165.8

44 · NUMERICAL RESULTS

Table 3.7: Fluid description for Example 2

Component	CO ₂	C ₁	C ₄	C ₁₀
Initial composition (%)	1	20	29	50
Injection composition (%)	90	10	-	-

3.0.2 Example 2 - Top layer of SPE 10 model

In the second example we again maximize cumulative oil recovery under CO₂ injection. The two-dimensional geological model is the top layer of the model defined in the SPE comparative solution project Christie and Blunt [2001], referred to as SPE 10. The model includes a total of four components, as specified in Table 3.7. Details on the reservoir model are provided in Table 3.8.

The well locations, along with a map of the permeability field, are depicted in Fig. 3.8. The control parameters in the optimization problem are again well BHPs, constrained to lie between 150 bar and 50 bar. A maximum (per well) gas production rate of 200 m³/d at reservoir conditions is also specified. The total simulation period is 1000 days. The well controls are determined at initial time and then at every 100-day interval. There are a total of ten control steps and 80 control parameters in this example.

We first generate two reference solutions as in the previous example. The cumulative oil production for these two cases, given in the first row ('Reference') of Table 3.9, are nearly identical because the nonlinear constraint violation in the unconstrained case is small. We next perform (nine) optimizations that honor the bound constraints but not the nonlinear constraints. The best optimum achieved in this case provides a cumulative oil production of 22,130 m³ (Run 6). Using the heuristic constraint handling procedure (third column), the best result is 22,030 m³ of oil (Runs 5 and 9), which exceeds the reference heuristic result by 8.8%. The best optimum achieved using the formal constraint handling treatment is 22,200 m³ of oil (Run 6). This value exceeds the reference solution by 9.5% and the best result using the heuristic treatment by 0.8%. It also exceeds the best unconstrained result (again, unconstrained here refers to the nonlinear constraints) of 22,130 m³, which is presumably because the nonlinear constraints are not very important in this example.

Oil production profiles are shown in Fig. 3.9. These profiles are very similar for the runs using the heuristic and formal constraint handling procedures. The BHP, gas rate, and oil rate profiles for the three cases are shown in Figs. 3.10, 3.11 and 3.12. We see that the gas production rates satisfy the nonlinear constraints (200 m³/d) at all times for both constraint handling procedures. Consistent with

Table 3.8: Model parameters for Example 2

Grid size	$60 \times 220 \times 1$	
Parameter	Value	Units
Δx	6.096	m
Δy	3.048	m
Δz	0.6096	m
Depth	2574	m
Initial pressure	75	bar
Temperature	100	°C
Rock compressibility	7.2×10^{-5}	1 / bar
Simulation time	1000	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	-
Well locations [grid block no.]	<i>i</i>	<i>j</i>
Injector 1	58	9
Injector 2	58	126
Injector 3	2	67
Injector 4	2	211
Producer 1	2	3
Producer 2	58	67
Producer 3	2	143
Producer 4	58	210

46 · NUMERICAL RESULTS

Table 3.9: Oil production in 10^3 m^3 (Example 2) for the optimized objective function without satisfying the nonlinear constraints ('Unconstr.'), satisfying the nonlinear constraints using the heuristic treatment ('Heuristic'), and satisfying the nonlinear constraints using the formal approach ('Formal'). Best feasible results shown in bold.

Run	Unconstr.	Heuristic	Formal
Reference	20.29	20.28	
1	21.77	21.77	21.77
2	22.01	22.01	22.01
3	18.16	18.16	18.17
4	21.68	21.68	21.68
5	22.03	22.03	22.04
6	22.13	20.67	22.20
7	21.68	21.68	21.99
8	22.07	21.48	22.07
9	22.03	22.03	22.03

Table 3.10: 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

Fig. 3.9, the oil production rates in Figs. 3.11 and 3.12 resemble one another, and they are clearly different than the reference solution in Fig. 3.10. For this example, the formal approach required an average of 56 forward simulations, while the heuristic approach required an average of only 24.

3.0.3 Example 3: Twelve-well channelized system

Our third example uses the three-dimensional geological model introduced in Van Essen G. et al. [2009]. We again consider CO₂ injection, though this model contains a total of six components, defined in Table 3.10. Further details are given in Table 3.11. A map of the x -component of permeability (here $\mathbf{k}_x = \mathbf{k}_y = 10\mathbf{k}_z$), along with the locations of the wells, is shown in Fig. 3.13.

The control parameters of our optimization problem are again the well BHPs. The wells are constrained to operate between an upper bound of 120 bar and a lower bound of 90 bar. We also specify nonlinear constraints on both injection

Table 3.11: 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>i</i>	<i>j</i>	
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

and production in the form of maximum gas flow rates of 200,000 m³/d for the producers and 40,000 m³/d for the injectors (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 3.12. The feasible reference case yields 5.030×10^6 m³ of oil, while the best heuristically constrained case (Run 2) provides 5.457×10^6 m³ of oil, an improvement of 8.5%. The best formally constrained case (Run 3) achieves an optimum of 5.306×10^6 m³ of oil, which exceeds the reference case by 5.5% but is less than the best heuristic case. The oil production profiles for the best runs, along with the feasible reference case, are shown in Fig. 3.14. 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 constraint handling approach typically required about 48 forward simulations, while the heuristic treatment required only about 26. Our findings for this example clearly illustrate the potential advantages of the heuristic treatment for complex optimization problems involving multiple wells operating under nonlinear constraints.

3.1 Concluding remarks

In this work we formulated and tested an adjoint-based optimization procedure for compositional reservoir simulation. The method we employed was implemented into Stanford's Automatic Differentiation-based General Purpose Research Simulator (AD-GPRS). The use of automatic differentiation simplifies the adjoint implementation and subsequent code enhancements. Two different treatments for handling nonlinear constraints were presented. In the formal constraint handling procedure, lumped constraints and their gradients are provided to the optimizer, and feasibility is enforced by the optimization algorithm. In the second (heuristic) procedure, an optimization satisfying only the bound and linear constraints is performed first. Then, the forward model is run using the controls from the first stage, but the simulator is allowed to switch from BHP to rate control (for a problem in which BHPs are the control variables) as required to satisfy the nonlinear rate constraints.

Table 3.12: Oil production in 10^6 m^3 (Example 3) for the optimized objective function without satisfying the nonlinear constraints ('Unconstr.'), satisfying the nonlinear constraints using the heuristic treatment ('Heuristic'), and satisfying the nonlinear constraints using the formal approach ('Formal'). Best feasible results shown in bold.

Run	Unconstr.	Heuristic	Formal
Reference	5.030	5.030	
1	5.450	5.449	5.284
2	5.467	5.457	5.294
3	5.171	5.171	5.306
4	5.288	5.287	5.132
5	5.424	5.423	5.224
6	5.344	5.348	5.260
7	5.321	5.230	4.994
8	5.207	5.205	5.196
9	5.353	5.349	4.986

Numerical results were presented for four example cases of increasing complexity. Nine runs, starting from different initial conditions, were performed in all cases, for both the heuristic and the formal nonlinear constraint treatments. In our examples, the control variables were the time-varying well BHPs, and maximum injection or production rate specifications entered as nonlinear constraints. The total number of control variables ranged from 40 to 320. Improvement in cumulative oil produced (which was the objective function in all cases) using our optimization procedures ranged from 4.2% to 11.6% relative to the reference solutions.

In the simplest case (Example 1), the formal constraint handling approach was shown to outperform the heuristic procedure when we increased the number of control variables from 40 to 320. In the next (somewhat more complicated) case, Example 2, the formal constraint treatment continued to outperform the heuristic procedure, though its advantage was very slight. In the other two cases, which were more challenging in terms of model size and number of wells (and they involved three-dimensional models, while the first two examples were two-dimensional), the heuristic treatment provided better objective function values than the formal approach.

These observations suggest that, although the formal constraint handling approach is theoretically superior, complications related to constraint lumping and the existence of poor local optima (and the additional complexity inherent in problems

with large numbers of optimization variables and nonlinear constraints) may render the formal procedure less effective than the heuristic approach in challenging cases. Thus, though we expect (and observe) the formal procedure to be the method of choice in relatively simple cases, the heuristic approach should be considered for use in more complex problems. It may even be beneficial to apply some type of hybrid technique, where the result from the heuristic method is used as the initial guess for the formal procedure. The use of a sequence of optimizations, with increasing numbers of control periods, should also be considered. Finally, it is important to note that the heuristic constraint treatment was found to be significantly more efficient than the formal approach.

In addition to the discrete adjoint procedure, which was used for all of the examples presented in this paper, we also derived and implemented a continuous adjoint formulation. We showed that the two formulations use different final time conditions and as a result the gradients do not agree in general. However, the two boundary conditions become almost identical as the size of the last time step approaches zero, in which case the gradients obtained by both formulations are essentially the same.

There are a number of areas in which future research should be directed. Other treatments for nonlinear constraints, both formal and heuristic, should be considered, and the relative benefits of controlling rates instead of BHPs should be assessed. It will be of interest to apply the general optimization framework to larger and more realistic simulation models. Other types of wells (horizontal, deviated, multilateral) should also be considered, along with the optimization of downhole inflow control devices. The overall approach can be extended to perform robust control (to account for geological uncertainty) and hierarchical (multi-objective) control to balance long-term and short-term objectives. Finally, our procedures could be applied for the optimization of CO₂ storage or for combined EOR-CO₂ storage operations.

Acknowledgements

We thank Denis Voskov and Oleg Volkov for useful discussions and assistance with AD-GPRS, and 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.

3.1 CONCLUDING REMARKS · 51

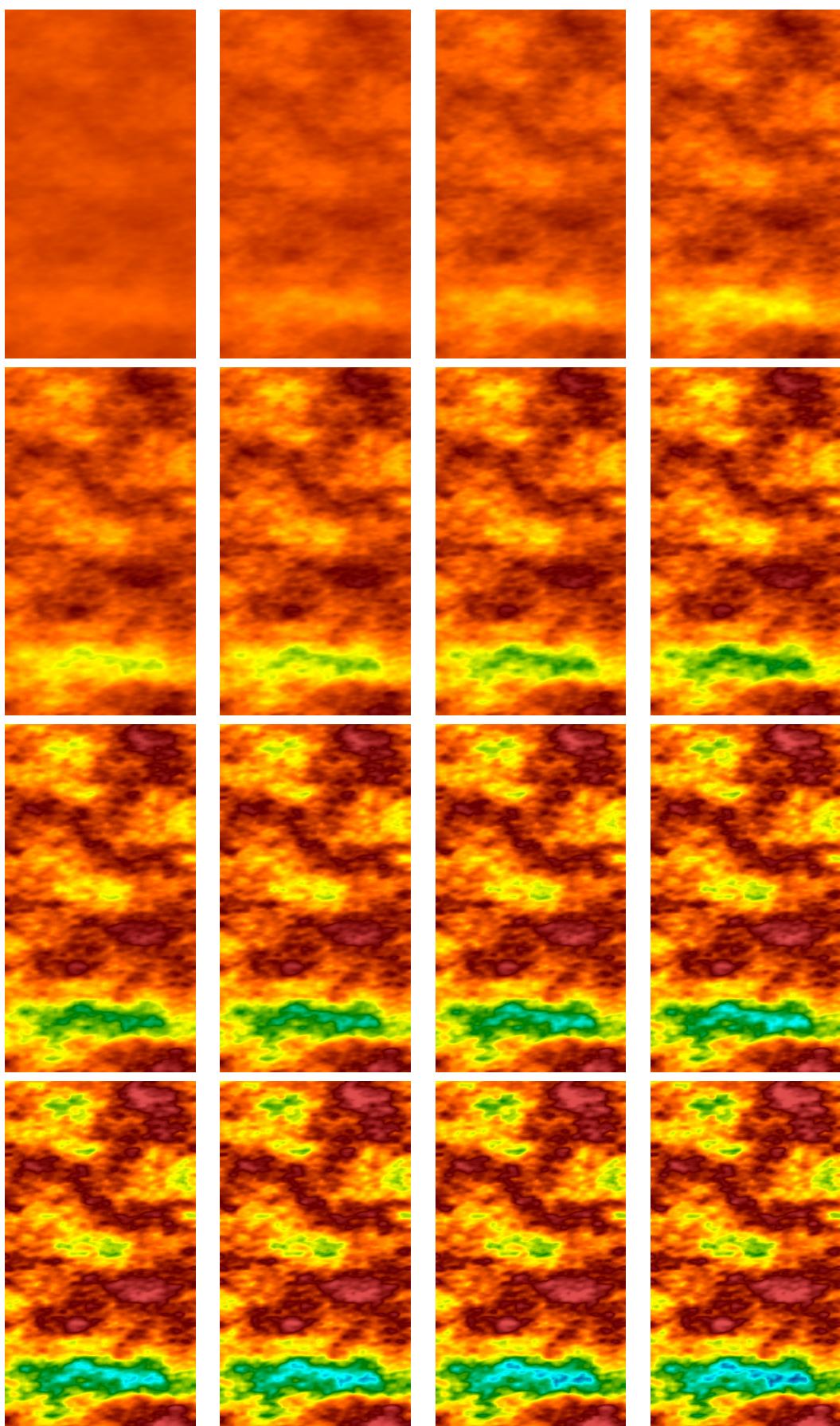


Figure 3.1: hgchgm

52 · NUMERICAL RESULTS

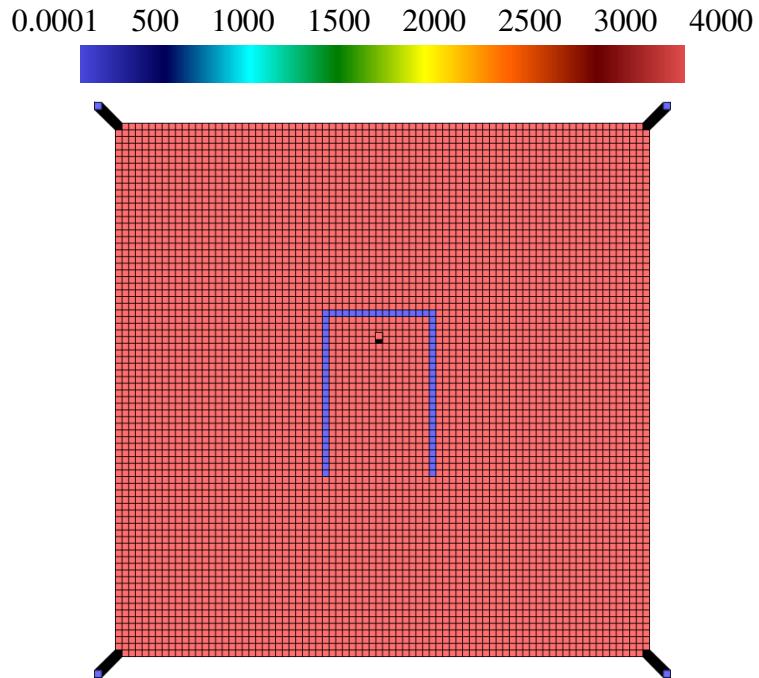


Figure 3.2: Injection wells (blue) and production well (red) for Example 1. Background shows \mathbf{k}_x ($\mathbf{k}_x = \mathbf{k}_y$).

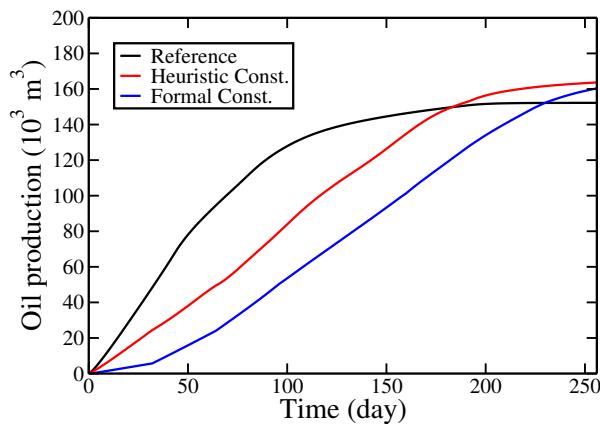


Figure 3.3: Oil production versus time (Example 1, 40 control variables). Results are for feasible reference case (black curve), best heuristically constrained solution (Run 8, red curve) and best formally constrained solution (Run 4, blue curve).

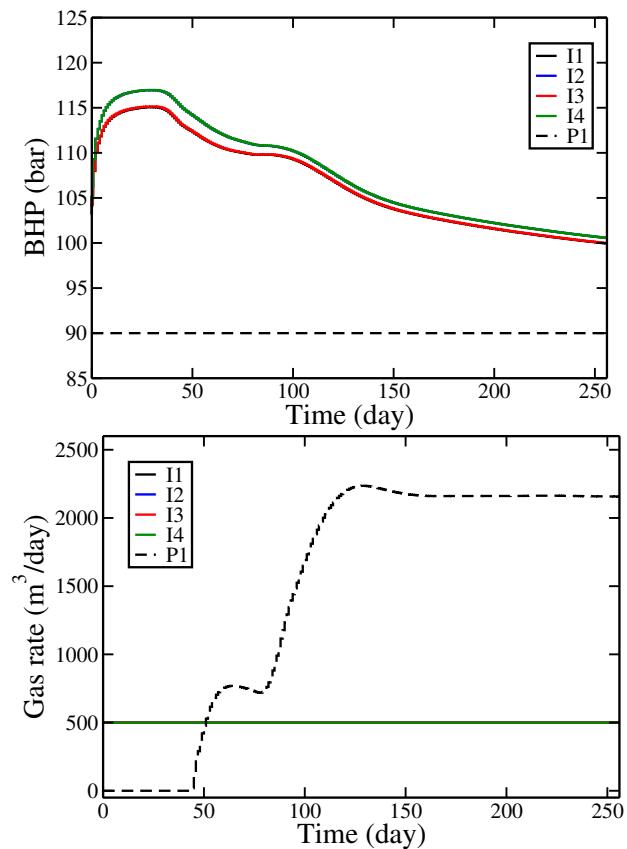


Figure 3.4: BHPs (top) and gas rates (bottom) for the feasible reference solution (Example 1). Gas injection rates are all equal to $500 \text{ m}^3/\text{d}$ during the entire simulation period.

54 · NUMERICAL RESULTS

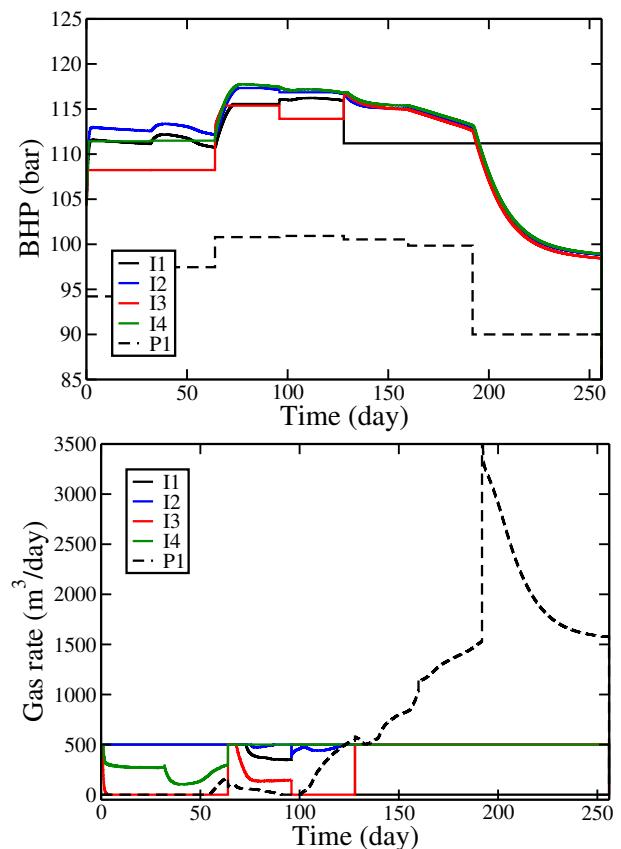


Figure 3.5: BHPs (top) and gas rates (bottom) for the best heuristically constrained solution using 40 controls (Example 1, Run 8).

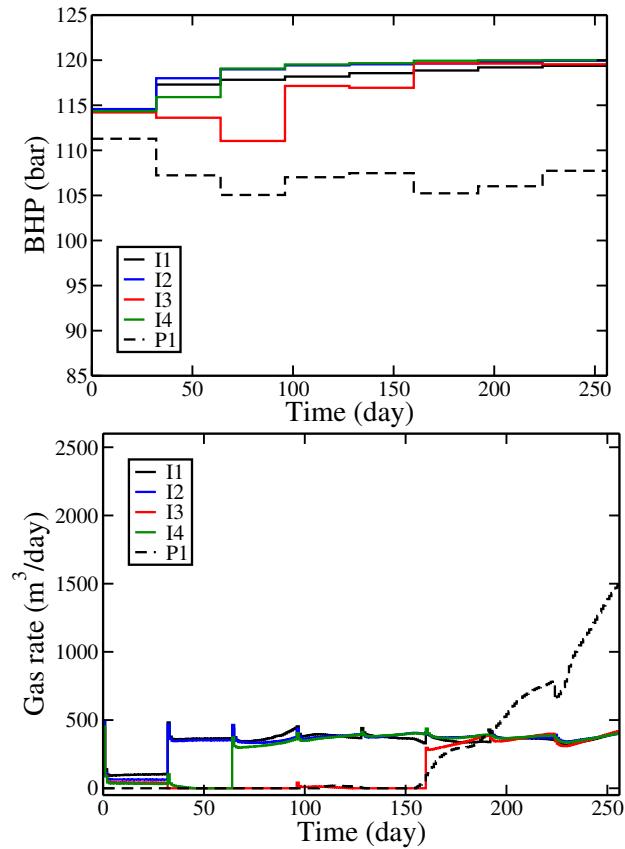


Figure 3.6: BHPs (top) and gas rates (bottom) for the best formally constrained solution using 40 controls (Example 1, Run 4).

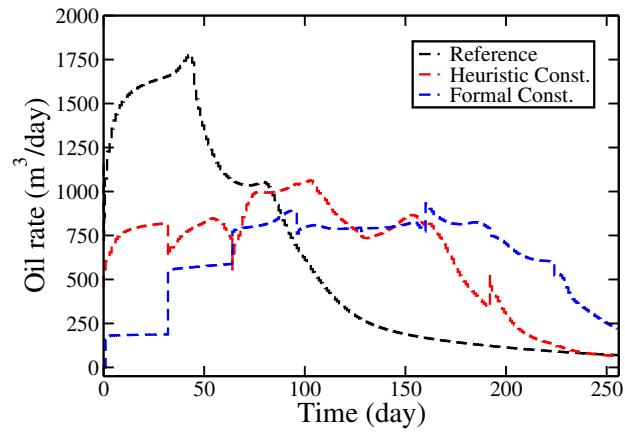


Figure 3.7: Oil rates for Example 1 (40 controls), reference (black dashed line), heuristically constrained (red dashed line) and formally constrained (blue dashed line).

56 · NUMERICAL RESULTS

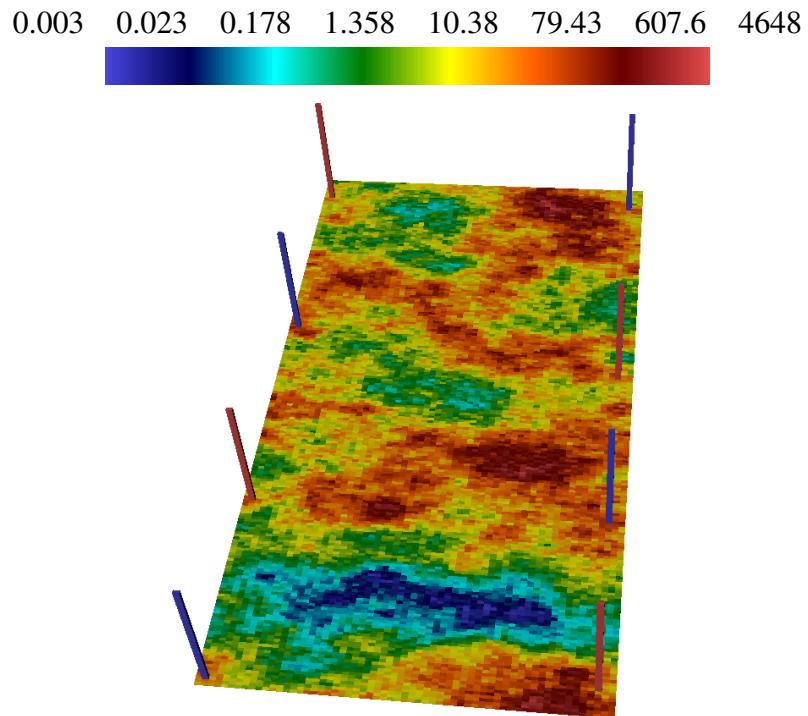


Figure 3.8: Injection wells (blue) and production wells (red) for Example 2. Background shows $\log k_x$ ($k_x = k_y$).

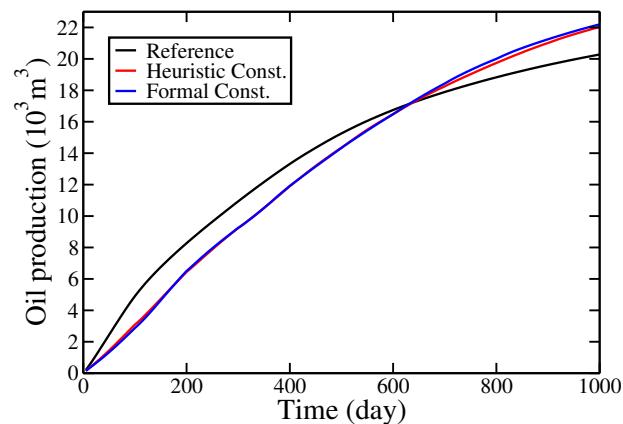


Figure 3.9: Oil production versus time for Example 2. Results are for feasible reference case (black curve), best heuristically constrained solution (Run 9, red curve) and best formally constrained solution (Run 6, blue curve).

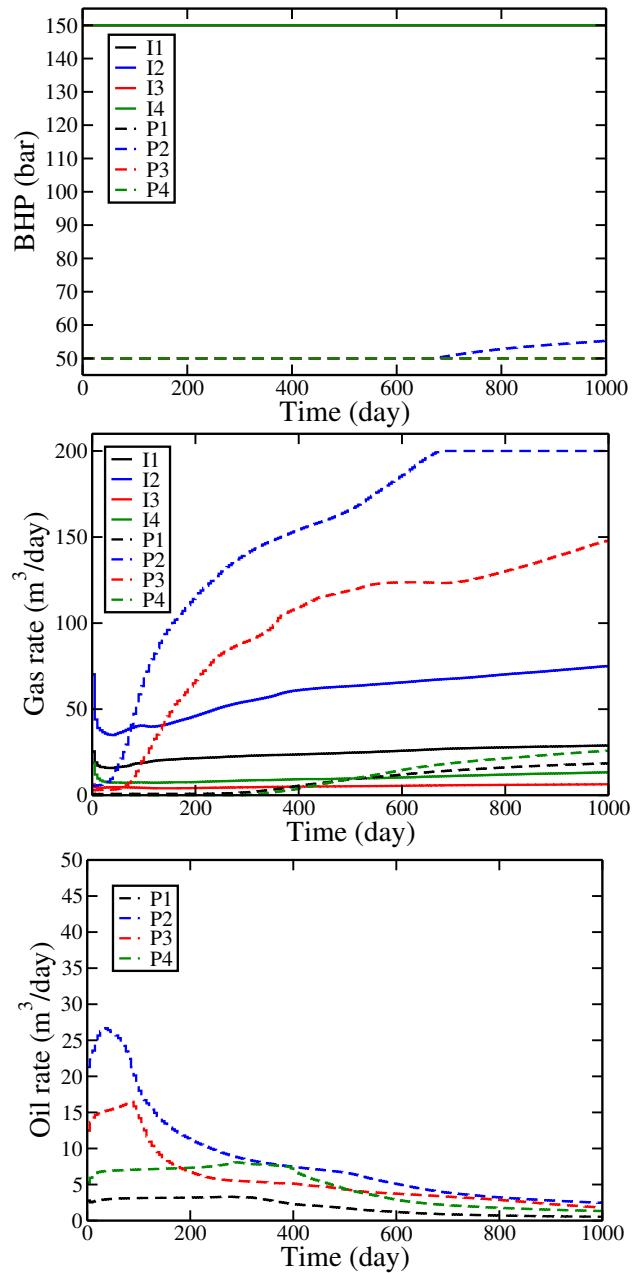


Figure 3.10: BHPs (top), gas rates (middle) and oil rates (bottom) for the feasible reference solution (Example 2).

58 · NUMERICAL RESULTS

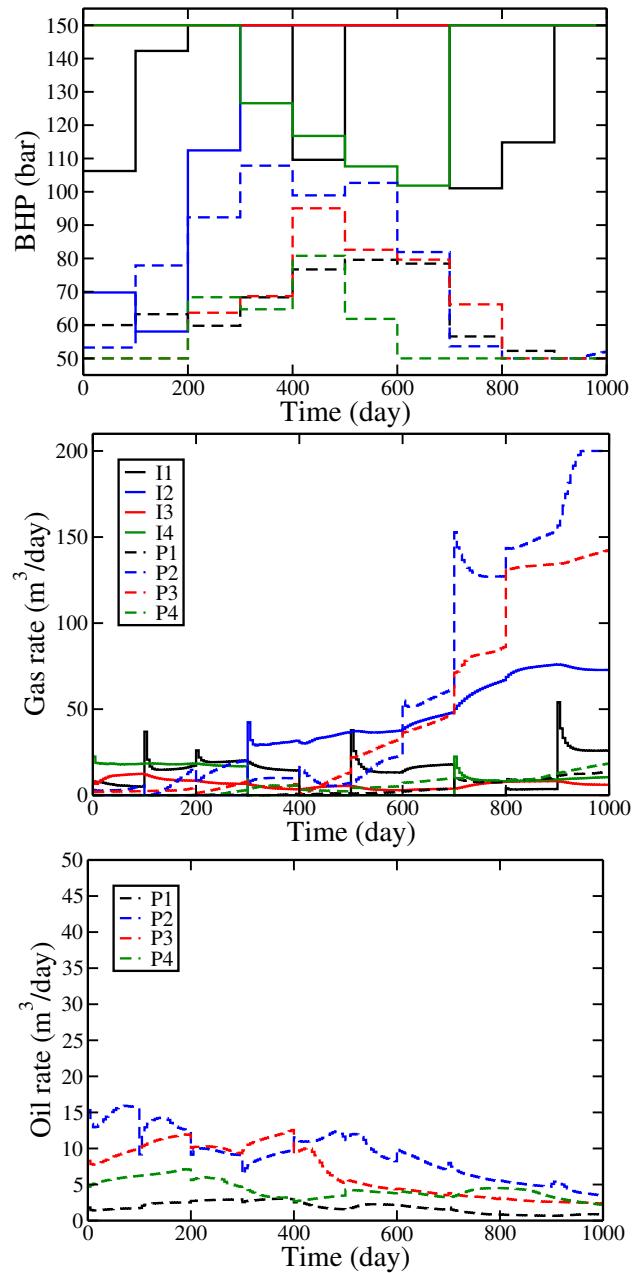


Figure 3.11: BHPs (top), gas rates (middle) and oil rates (bottom) for the best heuristically constrained solution (Example 2, Run 9).

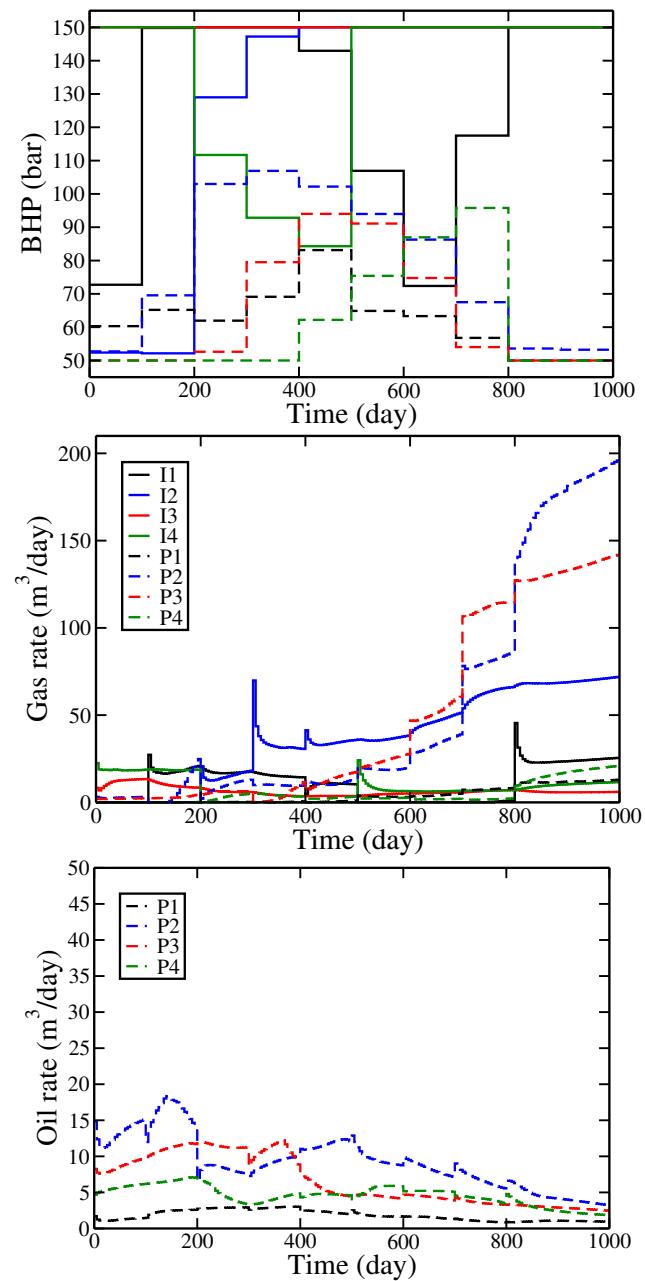


Figure 3.12: BHPs (top), gas rates (middle) and oil rates (bottom) for the best formally constrained solution (Example 2, Run 6).

60 · NUMERICAL RESULTS

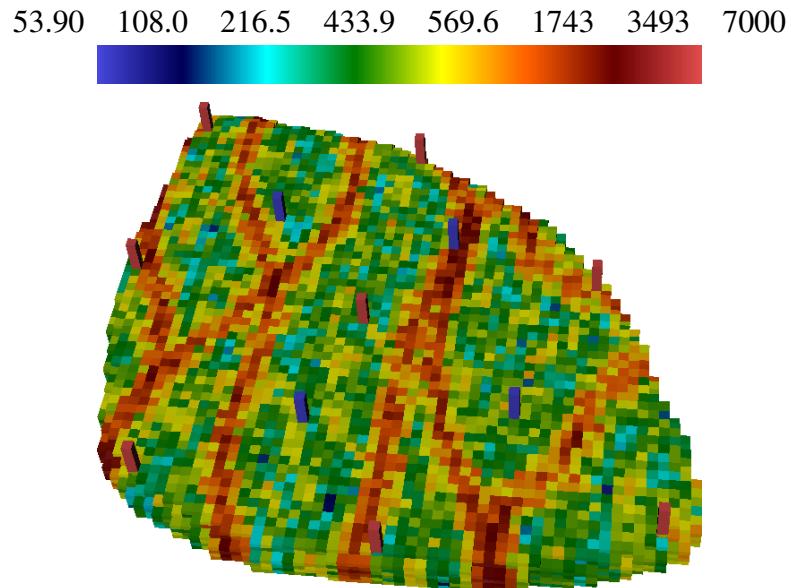


Figure 3.13: Reservoir model and wells for Example 3 (from Van Essen G. et al. [2009]). Background shows $\log k_x$.

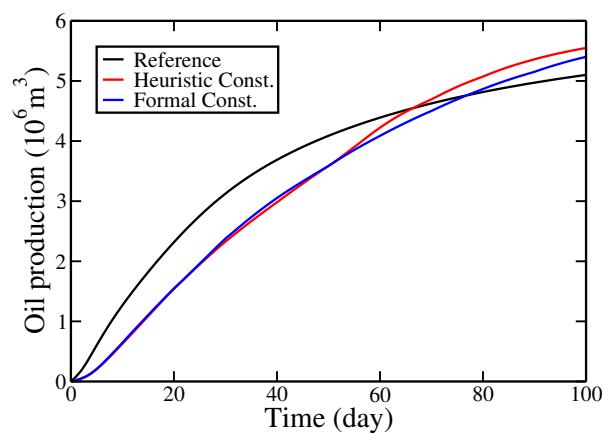


Figure 3.14: Oil production versus time for Example 3. Results are for feasible reference case (black curve), best heuristically constrained solution (Run 2, red curve) and best formally constrained solution (Run 3, blue curve).

Part IV

Numerical examples

Bibliography

- Anazi, B. D. A. (2007). Enhanced oil recovery techniques and nitrogen injection. CSEG Recorder, pages 29–33.
- Asheim, H. (1988). Maximization of water sweep efficiency by controlling production and injection rates. In SPE Paper 18365 presented at the SPE European Petroleum Conference, London, UK.
- Aziz, K. and Settari, A. (1979). Petroleum Reservoir Simulation. Applied Science Publishers, London.
- Bertsekas, D. P. (1999). Nonlinear Programming. Athena Scientific.
- Bonnans, Fredaric, J., Gilbert, Charles, J., Lemarachal, Sagastizabal, C., and A., C. (2006). Numerical optimization: Theoretical and practical aspects. Universitext, Berlin: Springer-Verlag.
- Brouwer, D. R. and Jansen, J. D. (2004). Dynamic optimization of water flooding with smart wells using optimal control theory. SPE Journal, 9(4):391–402.
- Bryson, A. and Ho, Y. (2001). Applied Optimal Control. Taylor and Francis (Hemisphere), Levittown.
- Cao, H. (2002). Development of techniques for general purpose simulators. Ph.D. thesis, Stanford University.
- Chavent, G., Dupuy, M., and Lemonnier, P. (1975). History matching by use of optimal control theory. SPE Journal, 15:74–86.
- Chen, W., Gavalas, G., and Wasserman, M. (1974). A new algorithm for automatic history matching. SPE Journal, 14(6):593–608.
- Chen, Y. and Oliver, D. (2010). Ensemble-based closed-loop optimization applied to Brugge field. SPE Reservoir Evaluation & Engineering, 13(1):56–71.
- Christie, M. and Blunt, M. (2001). Tenth SPE comparative solution project:

64 · Bibliography

- a comparison of upscaling techniques. SPE Reservoir Evaluation and Engineering, 4(4):308–317.
- Coats, K. (1980). An equation of state compositional model. SPE Journal, 20(5):363–376.
- De Montleau, P., Cominelli, A., Neylon, K., Rowan, D., Pallister, I., Tesaker, O., and Nygard, I. (2006). Production optimization under constraints using adjoint gradients. In 10th European Conference on the Mathematics of Oil Recovery.
- Echeverría Ciaurri, D., Isebor, O. J., and Durlofsky, L. J. (2011). Application of derivative-free methodologies to generally constrained oil production optimisation problems. International Journal of Mathematical Modelling and Numerical Optimisation, 2:134–161.
- et al., S. K. R. (2011). Surface and subsurface requirements for successful implementation of offshore chemical enhanced oil recovery.
- et al., T. A. J. (2010). Comparative study of different eor methods.
- Fleshman, R. and Lekic, O. (1999). Artificial lift for high volume production. Oilfield Review, (11):48–63.
- Gill, P. E., Murray, W., and Saunders, M. A. (2005). SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM Rev., 47(1):99–131.
- Glover, P. (2001). Formation Evaluation MSc Course Notes. Aberdeen University.
- Gunzburger, M. (2003). Perspectives in Flow Control and Optimization. SIAM.
- Hager, W. W. (2000). Runge-Kutta methods in optimal control and the transformed adjoint system. Numerische Mathematik, 87(2):247–282.
- Han, C., Wallis, J., Sarma, P., Li, G., Schrader, M. L., and Chen, W. (2013). Adaptation of the CPR preconditioner for efficient solution of the adjoint equation. SPE Journal, 18(2):207–213.
- Jansen, J. D. (2011). Adjoint-based optimization of multi-phase flow through porous media - a review. Computers & Fluids, 46(1):40 – 51.
- Juttner, I. (1997). Oil displacement in miscible condition. Rudarsko-Geoloko-NAFTNI Zbornik, 9(1):63–66.
- Kermen, E. (2011). Thermal Enhancement of Water-Flooding in Medium- Heavy Oil Recovery, in Applied Earth Sciences. Delft University of Technology, The Netherlands.

- Kourounis, D., Durlofsky, L. J., Jansen, J. D., and Aziz, K. (2014). Adjoint formulation and constraint handling for gradient-based optimization of compositional reservoir flow. *Computational Geosciences*, pages 1–21.
- Kourounis, D. and Schenk, O. (2015). Constraint handling for gradient-based optimization of compositional reservoir flow. *Computational Geosciences*, 19(5):1109–1122.
- Kraaijevanger, J., Egberts, P., Valstar, J., and Buurman, H. (2007). Optimal waterflood design using the adjoint method. In Paper SPE 105764 presented at the SPE Reservoir Simulation Symposium, Houston, USA.
- Kulkarni, M. M. (2003). Immiscible and Miscible Gas-Oil Displacements in Porous Media. Louisiana State University.
- Li, R., Reynolds, A., and Oliver, D. (2003). History matching of three-phase flow production data. *SPE Journal*, 8(4):328–340.
- Liu, W., Ramirez, W., and Qi, Y. (1993). Optimal control of steam flooding. *SPE Advanced Technology Series*, 1(2):73–82.
- Mattax and Dalton (1990). Reservoir simulation. Society of Petroleum Engineers.
- Mehos, G. and Ramirez, W. (1989). Use of optimal control theory to optimize carbon dioxide miscible-flooding enhanced oil recovery. *Journal of Petroleum Science and Engineering*, 2(4):247–260.
- Melzer, L. S. and Midland, T. (2012). Carbon dioxide enhanced oil recovery co2 eor): Factors involved in adding carbon capture, utilization and storage (ccus) to enhanced oil recovery.
- Meyer, J. P. (2007). Summary of carbon dioxide enhanced oil recovery co2 eor injection well technology. American Petroleum Institute, page 54.
- Nadarajah, S. and Jameson, A. (2000). A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. In AIAA 38th Aerospace Sciences Meeting and Exhibit, Reno, NV.
- Oliver, D., Reynolds, A., and Liu, N. (2008). Inverse Theory for Petroleum Reservoir Characterization and History Matching. Cambridge University Press, Cambridge.
- Petra, N. and Stadler, G. (2011). Model variational inverse problems governed by partial differential equations. ICES Report 11-05, The Institute for Computational Engineering and Sciences, The University of Texas at Austin.

- Ramirez, W. (1987). *Application of Optimal Control Theory to Enhanced Oil Recovery*. Elsevier Science Ltd, Amsterdam.
- Romero-Zeran, L. (2012). *Advances in Enhanced Oil Recovery Processes*. University of New Brunswick, Canada.
- Sarma, P., Chen, W., Durlofsky, L. J., and Aziz, K. (2008). Production optimization with adjoint models under nonlinear control-state path inequality constraints. *SPE Reservoir Evaluation and Engineering*, 11(2):326–339.
- Sarma, P., Durlofsky, L. J., Aziz, K., and Chen, W. H. (2006). Efficient real-time reservoir management using adjoint-based optimal control and model updating. *Computational Geosciences*, 10(1):3–36.
- Shine, J. and Holtz, M. (2008). Reserve growth & higher recovery using nitrogen gas injection. *Proc. 2008 Wyoming EOR/IOR Conference*.
- Stengel, R. (1986). *Stochastic Optimal Control Theory and Application*. Wiley, New York.
- Sudaryanto, B. and Yortsos, Y. (2000). Optimization of fluid front dynamics in porous media using rate control. *Physics of Fluids*, 12(7):1656–1670.
- Suwartadi, E., Krogstad, S., and Foss, B. (2012). Nonlinear output constraints handling for production optimization of oil reservoirs. *Computational Geosciences*, 16:499–517.
- Syed, F. I., Tunio, A. H., and Ghirano, N. A. (2011). Compositional analysis and screening for enhanced oil recovery processes in different reservoir and operating conditions. *International Journal of Applied*, 1(4).
- Van Essen G., Zandvliet, M., Van den Hof, P., Bosgra, O., and Jansen, J. D. (2009). Robust waterflooding optimization of multiple geological scenarios. *SPE Journal*, 14(1):202–210.
- Virnovski, G. (1991). Waterflooding strategy design using optimal control theory. In *6th European IOR Symposium*, pages 437–446, Stavanger, Norway.
- Voskov, D. and Tchelepi, H. (2012). Comparison of nonlinear formulations for two-phase multi-component EoS based simulation. *Journal of Petroleum Science and Engineering*, 82–83:101–111.
- Voskov, D. V., Younis, R., and Tchelepi, H. A. (2009). Comparison of nonlinear formulations for isothermal compositional flow simulation. In *SPE paper 118966 presented at the SPE Reservoir Simulation Symposium*, The Woodlands, Texas, USA.

- Walther, A. (2007). Automatic differentiation of explicit Runge-Kutta methods for optimal control. *Computational Optimization and Applications*, 36(1):83–108.
- Wang, C., Li, G., and Reynolds, A. (2009). Production optimization in closed-loop reservoir management. *SPE Journal*, 14(3):506–523.
- Young, L. and Stephenson, R. (1983). A generalized compositional approach for reservoir simulation. *Journal of Petroleum Science and Engineering*, 23(5):727–742.
- Younis, R. and Aziz, K. (2007). Parallel automatically differentiable data-types for next-generation simulator development. In SPE Paper 106493 presented at the SPE Reservoir Simulation Symposium, Houston, Texas.
- Younis, R., Tchelepi, H., and Aziz, K. (2010). Adaptively localized continuation-Newton method–Nonlinear solvers that converge all the time. *SPE Journal*, 15(2):526–544. SPE-119147-PA.