



# A Second-Order Finite Volume Method for Field-Scale Reservoir Simulation

Anna Kvashchuk<sup>1,2</sup> · Robert Klöfkorn<sup>3</sup> · Tor Harald Sandve<sup>4</sup>

Received: 10 June 2022 / Accepted: 7 July 2023 / Published online: 18 August 2023  
© The Author(s) 2023

## Abstract

Subsurface reservoirs are large complex systems. Reservoir flow models are defined on complex grids that follow geology with relatively large block sizes to make consistent simulations feasible. Reservoir engineers rely on established reservoir simulation software to model fluid flow. Nevertheless, fluid front position inaccuracies and front smearing on large grids may cause significant errors and make it hard to predict hydrocarbon production efficiency. We investigate higher-order methods that reduce these undesired effects without refining the grid, thus making reservoir simulation more accurate and robust. For this paper, we implemented a second-order finite volume method with linear programming (LP) reconstruction in the open-source industry-grade reservoir simulator OPM Flow (part of the open porous media initiative, OPM). We benchmark it against the first-order method on full-scale cases with standard coarse and refined grids. We prepared open refined-grid models of a synthetic reservoir with an unstructured grid and refined Norne field example. Our results confirm that the LP method predicts front positions as accurately as the first-order method on the refined grid for problems dominated by transport. These include the water alternating gas scenario on the synthetic reservoir and piston-type injection on the Norne field. Moreover, we study the gains from the LP method for CO<sub>2</sub> injection problems on the Norne field with full multi-phase complexity beyond transport. We observe the relevant difference between the first- and the second-order methods in these cases. However, in some configurations, the reservoir complexity overshadows the gains from the second-order methods.

---

✉ Anna Kvashchuk  
kvashchuk.anna@gmail.com

Robert Klöfkorn  
robertk@math.lu.se

Tor Harald Sandve  
tosa@norce-research.no

<sup>1</sup> Equinor ASA, Forusbeen 50, 4035 Stavanger, Norway

<sup>2</sup> Faculty of Science and Technology, Department of Energy Resources, University of Stavanger, P.O. Box 8600, 4036 Stavanger, Norway

<sup>3</sup> Center for Mathematical Sciences, Lund University, P.O. Box 117, 221 00 Lund, Sweden

<sup>4</sup> Energy and Technology, NORCE Norwegian Research Centre, Nygårdsgaten 112, 5008 Bergen, Norway

**Keywords** Second-order method · Reservoir simulation · OPM · Norne · Realistic reservoir

## 1 Introduction

Modeling multi-phase multi-component flow in porous media requires accurate and robust numerical methods. It is an essential part of the oil and gas production (Trangenstein and Bell 1989; Coats et al. 1995), carbon storage (Class et al. 2009; Celia and Nordbotten 2009; Sandve et al. 2018; Mykkeltvedt et al. 2021), enhanced oil recovery (Lake 1989; Mykkeltvedt et al. 2017) and many other applications.

The first-order finite volume (FV) is the usual method for modeling multi-phase multi-component flow. It is the default option in many standard reservoir simulators, both commercial and open-source, for example, ECLIPSE (2014), OPM (Open Porous Media) (Lauser et al. 2018), the Matlab Reservoir Simulation Toolbox (MRST) (Lie 2019), DuMu<sup>c</sup> (Flemisch et al. 2011), PFLOTRAN (Lichtner et al. 2019), an Automatic Differentiation General Purpose Research Simulator (ADGPRS) (Voskov and Tchelepi 2012) or TOUGH3 (Jung et al. 2017). First-order methods are widely used because of their robustness and ease of implementation. Unfortunately, those are known to suffer from numerical diffusion, which leads to smearing of the fluid fronts and, therefore, incorrect computations of the front position, components concentrations, water breakthrough, etc.

To reduce the numerical diffusion and increase the accuracy, there are mainly two options: to refine the grid or increase the order of the numerical method. Grid refinement, while being a valid solution in many cases, is often impractical for reservoir simulation due to the reservoir size and complex grids, faults, etc. It complicates the model and increases the simulation time significantly. On the other hand, increasing the order of the numerical method can provide a more accurate solution with reduced grid-orientation effects and better front resolution on a practical grid block size without modifying the grid (Sammon et al. 2001). However, a practical challenge with the higher-order methods is unphysical oscillations (undershoots and overshoots) around the front, which require the introduction of slope-limiters (LeVeque 2002).

The early papers applying higher-order methods in reservoir simulation date back to the 1980s. Bell and Shubin (1985) presented a higher-order Godunov scheme for one- and two-dimensional five-spot problems with miscible and immiscible displacement flow models. In Rubin and Blunt (1991), Blunt and Rubin (1992) and Rubin and Edwards (1993), the authors show how higher-order total variation diminishing (TVD) schemes can be applied to one- and two-dimensional simplified reservoir simulation. Chen et al. (1993) used second-order TVD- and third-order essentially non-oscillatory (ENO) schemes to minimize grid orientation effects and improve front resolution on the 2D five-spot model. In May and Berger (2013) proposed to use constraint optimization with linear programming to compute the reconstruction in a higher-order method. And Chen and Li (2016) further proposed an improved linear-programming scheme that did not require an initial gradient computation.

Despite continued research of higher-order methods in reservoir simulation (Durlafsky et al. 1992; Harten 1997; Geiger-Boschung et al. 2009; Lamine and Edwards 2015; Contreas et al. 2016; Mykkeltvedt et al. 2017), most implementations are done in academic codes with Cartesian or simplex meshes. Only a few researchers applied them to implementation-intensive corner-point grids that capture complex geometries of subsurface reservoirs. In Lie et al. (2020), a weighted-ENO (WENO, introduced in Jiang and Shu 1996; Liu et al.

1994) method was applied to simplified test cases on reservoir-type grids. In Klemetsdal et al. (2020), a discontinuous Galerkin (DG) method (introduced for transport problems in Cockburn and Shu 1989) was applied for compositional flow on realistic reservoir meshes. DG are very promising for such problems; however, since most industrial simulators rely on a data layout corresponding to that of first-order FV methods, the implementation of DG methods in industrial codes will be even more complicated than reconstruction-based higher-order FV methods.

In Klöfkor et al. (2017), the authors compared three second-order schemes, each with several limiting techniques on different 2D and 3D grids, including corner-point examples. The study concluded that the second-order method with reconstruction based on optimization with linear programming (second-order LP method) was among the best methods for 3D cases. Moreover, the second-order LP method was most suitable for implementation in the full-scale reservoir simulator OPM Flow.

In this paper, we present and verify the implementation of the second-order LP method in a full-scale reservoir simulator. To show the method's capabilities, we run WAG and CO<sub>2</sub> injection scenarios on a medium-sized synthetic reservoir with an unstructured grid and the openly available Norne field. We constructed experiments that isolate and highlights the effects of components' transport, for which the simulation quality will deteriorate for the classical methods. The implementation is done in OPM Flow, an open reservoir simulator capable of modeling the black-oil model as fast and accurately as conventional commercial reservoir simulators (Rasmussen et al. 2021; Lauser et al. 2018; Sandve et al. 2018). The open-source access allows us to integrate new features directly and verify them on industry-standard test cases and field studies. This means that the second-order method is readily available for practical reservoir simulation. A reservoir engineer wanting to increase the accuracy and improve the fluid front positioning and resolution can start the simulation with the higher order flag `-enable-higher-order` instead of going through the time-consuming and work-intensive process of grid refinement.

The paper is organized as follows: We start by describing the first- and second-order finite volume methods on reservoir discretizations. Section 3 presents the numerical results on realistic cases. We summarise the results in Sect. 4.

## 2 Discretization

In this section, we discuss discretization used in the fully-implicit numerical method for solving the black oil model extended with a solvent component (22–24).

Let us start by introducing a computational grid  $\mathcal{T}$  of the domain  $\Omega \subset R^d$  and denoting  $\partial\mathcal{T}$  as a tessellation of the boundary  $\partial\Omega$ . In the classical cell-centered finite volume method the unknown function is approximated by a piece-wise constant function, which takes the average value of the unknowns for each discretization element (Eymard et al. 2000). The finite volume method uses an integral formulation of the model Eqs. (22–24) applied on each element of the computational grid. For the sake of simplicity, all further derivations will be done for the water component equation from (23):

$$\frac{\partial}{\partial t} A_w - \nabla \cdot \frac{\lambda_w k}{B_w} (\nabla p_\alpha - \rho_\alpha g) = Q_w. \quad (1)$$

As for time discretization we use an implicit Euler method. Following the derivations from LeVeque (2002) and Eymard et al. (2000), we integrate the water component equation over

the element  $E \in \mathcal{T}$ , apply the Gauss theorem and get the implicit formulation of a classical finite volume scheme:

$$(\phi A_w)^{n+1} = (\phi A_w)^n + \frac{dt}{|E|} \oint_{\partial E} \frac{\lambda_w^{n+1} k}{B_w} (\nabla P_w^{n+1} - \rho_w g) \cdot \mathbf{n} \, dx + Q_\kappa^{n+1}, \tag{2}$$

where the superscripts  $n$  and  $n + 1$  denote the time step,  $Q_\kappa^{n+1}$  is the averaged in time integral over the element of the right-hand side,  $A_w$  is the accumulation term of the water component,  $n_{ij}$  is the outer normal to the intersection  $e_{ij}$  and  $\lambda_w^{n+1}$  is the water mobility, which is equal to the ratio of the relative permeability function to the water viscosity. As we are modeling reservoirs, which typically are isolated from the surrounding rocks, a no-flow boundary condition is assumed.

The two accumulation terms are linear and can be easily computed using the averaged values at each element. The integral in the right-hand side can, in principle, be approximated using any quadrature rule, we choose the midpoint rule:

$$\begin{aligned} &\oint_{\partial E} \frac{\lambda_w^{n+1} k}{B_w} (\nabla P_w^{n+1} - \rho_w g) \cdot \mathbf{n} \, dx \\ &\approx \sum_{e_{ij} \in \partial E} |e_{ij}| \frac{\lambda_w^{n+1}(x_{ij}) k(x_{ij})}{B_w} (\nabla P_w^{n+1}(x_{ij}) - \rho_w(x_{ij}) g(x_{ij})) \cdot \mathbf{n}_{ij}. \end{aligned} \tag{3}$$

To calculate the gradient of the pressure we use the standard two-point flux approximation (Aavatsmark 2007; Alyaev et al. 2014) and the gravitational acceleration is approximated using the arithmetic mean. The mobility function  $\lambda_w^{n+1}$ , on the other hand, is non-trivial to compute. Let us denote it as:

$$\lambda_w^{n+1}(x_{ij}) = \lambda_w^{ij}. \tag{4}$$

The reconstruction of the phase mobility function will determine the type of the numerical method. As discussed in Lie et al. (2020), one can use either primary variables or the phase mobilities to reconstruct the second-order method. We compared the results when reconstructing primary variables and mobilities in the test run and observed that the choice did not influence the results significantly. Also reconstructing mobilities extends the current version of OPM more naturally. Therefore, in this paper, we chose to reconstruct the phase mobilities in the presented second-order method implementation.

In the first-order method, we use a simple upwind scheme for the mobility evaluation and the mid-point rule for the integral:

$$\lambda_w^{ij} = \begin{cases} \lambda_{ij}^- & \text{if } (\nabla P_w^{n+1} - \rho_w g) \cdot n_{ij} \leq 0, \\ \lambda_{ij}^+ & \text{otherwise,} \end{cases} \tag{5}$$

where  $\lambda_{ij}^-$  is the mobility function on the element  $E$  and  $\lambda_{ij}^+$ —on the neighboring element  $E'$ . For boundary elements, the boundary condition value is used instead of the neighboring value or the same value in the case of a no-flow boundary, which is typical in reservoir modeling. Now, when all the terms in (2) are discretized, we can repeat the same procedure with the obvious modifications for the other components and get a non-linear system of equations. The solution of the system is computed by the Newton method with an automatic differentiation approach for the computation of Jacobian matrices. More details on the use of automatic differentiation in OPM can be found in Lauser et al. (2018).

For second-order methods we use a reconstructed linear function for every element:

$$\begin{aligned} L_E(x) &:= \lambda_E + \nabla L_E \cdot (x - \mathbf{w}_E), \\ L_E(\mathbf{w}_{E'}) &= \lambda_{E'}, \quad \forall (E, E') \in \partial E, \end{aligned} \tag{6}$$

where  $\mathbf{w}_E$  is the coordinates of the barycenter of the element  $E$ , and  $\nabla L_E$  is the gradient that needs to be computed. For clarity of notation, we are dropping the water component index  $w$  in the mobility function. The same procedure is applied to all the mobility functions. Afterward, the same upwind scheme (5) is used, however, instead of  $\lambda_{ij}^-$  and  $\lambda_{ij}^+$  we use the reconstructed  $L_{ij}^-$  and  $L_{ij}^+$ , which corresponds to the linear reconstructions functions evaluated on the intersections barycenter on element  $E$  and its neighbor  $E'$ , respectively:

$$\lambda_w^{ij} = \begin{cases} L_{ij}^-(\mathbf{w}_{(E,E')}) & \text{if } (\nabla p_w^{n+1} - \rho_w g) \cdot n_{ij} \leq 0, \\ L_{ij}^+(\mathbf{w}_{(E,E')}) & \text{otherwise,} \end{cases} \tag{7}$$

In order to reconstruct the linear function we need to compute a gradient  $\nabla L_E$  on each element. Below we discuss one possible approach to computing the gradient and therefore the linear function.

### 2.1 Linear Programming Reconstruction

One option to compute the gradient for the linear reconstructions is by solving an optimization problem for each cell. The idea of using linear programming (LP) for reconstruction and limiting the gradient was presented in May and Berger (2013). The algorithm for obtaining the linear reconstruction still consists of two steps: computation of the initial gradient and solving a constrained optimization problem, where the goal is to minimize the difference between the initial and the limited gradients while satisfying monotonicity constraints. Each component of the multidimensional gradient is limited separately using scalars, which means that in the end the direction of the initial gradient could be changed (in contrast with scalar limiters, when all gradient's components are multiplied with the same scalar). Chen and Li (2016) addressed this in the paper, where they presented a method that does not require an initial gradient computation. Indeed, since both the direction and the length of the gradient can be changed by the LP algorithm, it would be beneficial computationally to omit the initial gradient computation. As the authors point out in Chen and Li (2016), the two algorithms give different optimal solutions, however, they also present a theorem, where they prove that the obtained gradient is sufficiently close to the unlimited least squares gradient. The approach used in this paper is based on Chen and Li (2016) and will be briefly described below.

Linear programming is an optimization method to find an extremum of the objective function under the linear constraints (Nocedal and Wright 2006). In our case, we want to minimize the difference between reconstructed values and the cell-averaged values without violating the monotonicity conditions:

$$\min\{\lambda_E, \lambda_{E'}\} \leq L_E(\mathbf{w}_{E'}) \leq \max\{\lambda_E, \lambda_{E'}\}, \quad \forall (E, E') \in \partial E. \tag{8}$$

This means that the reconstructed function evaluated in the center of every neighboring element should be bounded by minimum and maximum from the cell-averaged values of the current element and the neighbor. The elements center is calculated as an average of the corner coordinates. This will allow better numerical accuracy and reduce the numerical

diffusion without creating spurious oscillations. This monotonicity condition essentially acts like a minmod limiter applied to the gradient of the reconstructed function.

The standard form of the linear programming problem is:

$$\min \tilde{c}^T \tilde{x} \quad \text{subject to } \tilde{A}\tilde{x} = \tilde{b}, \tilde{x} \geq 0, \tag{9}$$

where  $\tilde{c}$  and  $\tilde{x}$  are vectors from  $\mathbb{R}^n$ , vector  $\tilde{b}$  belongs to  $\mathbb{R}^m$ , and  $\tilde{A} \in \mathbb{R}^{(m \times n)}$ .

As it was said before, we are minimizing the total gaps between the reconstructed values and the cell-averaged values at the neighboring cells:

$$\delta(L) := \sum_{\forall(E,E') \in \partial E} | \lambda_{E'} - L_E(\mathbf{w}_{E'}) |, \tag{10}$$

which is equal to

$$\sum_{\forall(E,E') \in \partial E} | \lambda_{E'} - L_E(\mathbf{w}_{E'}) | = \sum_{\forall(E,E') \in \partial E} | \lambda_{E'} - \lambda_E - \nabla L_E \cdot (\mathbf{w}_{E'} - \mathbf{w}_E) | \tag{11}$$

$$= \sum_{\forall(E,E') \in \partial E} | v_{E'} - \tilde{v}_{E'} |, \tag{12}$$

where  $v$  and  $\tilde{v}$  are variables defined as

$$\tilde{v}_{E'} = \nabla L_E \cdot (\mathbf{w}_{E'} - \mathbf{w}_E) \quad \text{and} \quad v_{E'} := \lambda_{E'} - \lambda_E. \tag{13}$$

As it was pointed out in Chen and Li (2016), the difference  $v_{E'} - \tilde{v}_{E'}$  has the same sign as  $v_{E'}$ , which brings us to the following LP problem:

$$\begin{aligned} \max \quad & \sum_{\forall(E,E') \in \partial E} \text{sgn}(v_{E'}) (\mathbf{w}_{E'} - \mathbf{w}_E) \cdot \nabla L_E \\ \text{subject to} \quad & v_{E'}^- \leq (\mathbf{w}_{E'} - \mathbf{w}_E) \cdot \nabla L_E \leq v_{E'}^+, \end{aligned} \tag{14}$$

where

$$\begin{aligned} v_{E'}^- &= \min\{0, \lambda_E - \lambda_{E'}\}, \\ v_{E'}^+ &= \max\{0, \lambda_E - \lambda_{E'}\}. \end{aligned} \tag{15}$$

The transition from max to min formulation in (14) is straightforward, so we will just write down how matrix  $A$  and vectors  $b$  and  $c$  from (9) will look like. The unknown vector  $x$  is the gradient of the linear reconstruction  $x = [\nabla L_E^x, \nabla L_E^y, \nabla L_E^z]^T$ . The matrix  $A$  and vectors  $c$  and  $b$  are:

$$c = \begin{bmatrix} \sum_{\forall(E,E') \in \partial E} \text{sgn}(v_{E'}) (x_{E'} - x_E) \\ \sum_{\forall(E,E') \in \partial E} \text{sgn}(v_{E'}) (y_{E'} - y_E) \\ \sum_{\forall(E,E') \in \partial E} \text{sgn}(v_{E'}) (z_{E'} - z_E) \end{bmatrix}, \tag{16}$$

$$A = \begin{bmatrix} x_{E_1} - x_E & y_{E_1} - y_E & z_{E_1} - z_E \\ \vdots & \vdots & \vdots \\ x_{E'} - x_E & y_{E'} - y_E & z_{E'} - z_E \\ -(x_{E_1} - x_E) & -(y_{E_1} - y_E) & -(z_{E_1} - z_E) \\ \vdots & \vdots & \vdots \\ -(x_{E'} - x_E) & -(y_{E'} - y_E) & -(z_{E'} - z_E) \end{bmatrix}, \quad b = \begin{bmatrix} v_{E_1}^+ \\ \vdots \\ v_{E'}^+ \\ -v_{E_1}^- \\ \vdots \\ -v_{E'}^- \end{bmatrix}, \quad (17)$$

We start the iteration process with zero gradient

$$x = [\nabla L_E^x, \nabla L_E^y, \nabla L_E^z]^T = [0, 0, 0]^T, \quad (18)$$

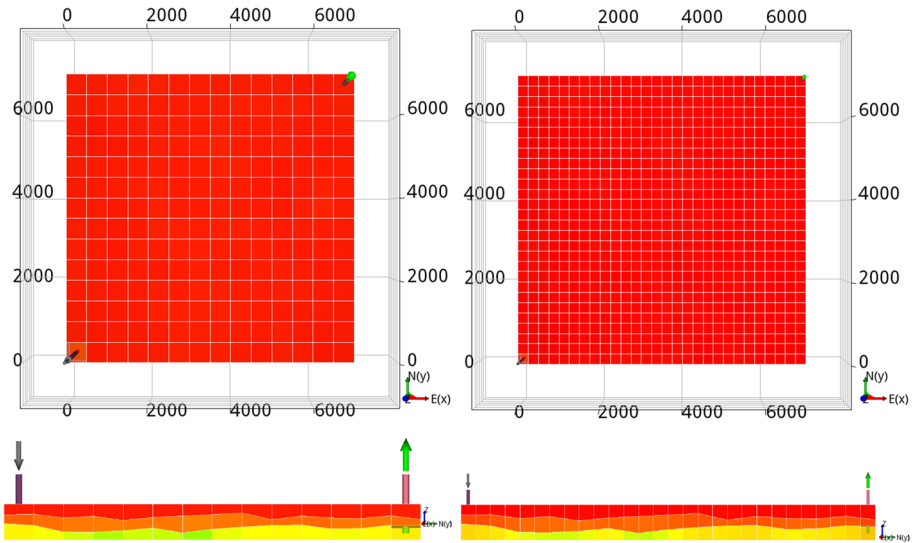
which corresponds to the first-order scheme. Following May and Berger (2013) and Chen and Li (2016), an all-inequality simplex method is used as a linear programming solver. Compared to the usual simplex method, the advantage of this method is that it is much faster on problems where the number of constraints is bigger than the number of unknowns, which is the case for polyhedral meshes where each cell has many neighbors. A detailed description of the all-inequality simplex algorithm can be found in the appendix of May and Berger (2013). Further on in this paper, we will refer to this method as the second-order LP method.

### 3 Numerical Results

In this chapter, we test and compare the methods on two realistic reservoirs. In the first test case, we compare the methods on a medium-sized realistic reservoir with an unstructured corner point grid when modeling water-alternating gas injection (WAG), Sect. 3.1. This test case provides a complicated setup in terms of the presented fluids and their interactions; however, the reservoir is relatively small and does not contain complex features like faults, regions, etc. After that, we present the simulation results on the Norne field, an open data set of a real reservoir on the Norwegian Continental Shelf. We run three different experiments on the Norne field. First, we model a piston-type injection on homogeneous and heterogeneous Norne fields to observe reduced smearing for the second-order method, Sect. 3.2. Second, we run the first-order method on a refined Norne model to validate the front resolution of both methods, Sect. 3.2. And finally, we present the carbon dioxide injection test case on a heterogeneous Norne field with its full complexity, Sect. 3.3. We run several setups to compare the accuracy of the front position of the second-order method in less and more complex conditions. We constructed experiments that isolate and highlights effects of components' transport, for which the simulation quality will deteriorate for the classical methods.

#### 3.1 Medium-Sized Realistic Reservoir

In this test case, we model a three-phase flow with four components on a 3D domain with an unstructured grid. The phases are the same as in the standard black-oil model: oleic, gaseous, and aqueous. The components are oil, gas, water, and solvent. The domain is a rectangular hexahedron, which is 7014 m long in  $X$  and  $Y$  directions and 120 m long in the  $Z$  direction, having  $14 \times 14 \times 3$  grids cells, each 501 m by 501 m by 40 m, see left part of Fig. 1. It is located 8325 m below the surface and has two wells—an injector in the lower



**Fig. 1** Top and west views of the coarse and refined medium-sized reservoir, described in the Sect. 3.1

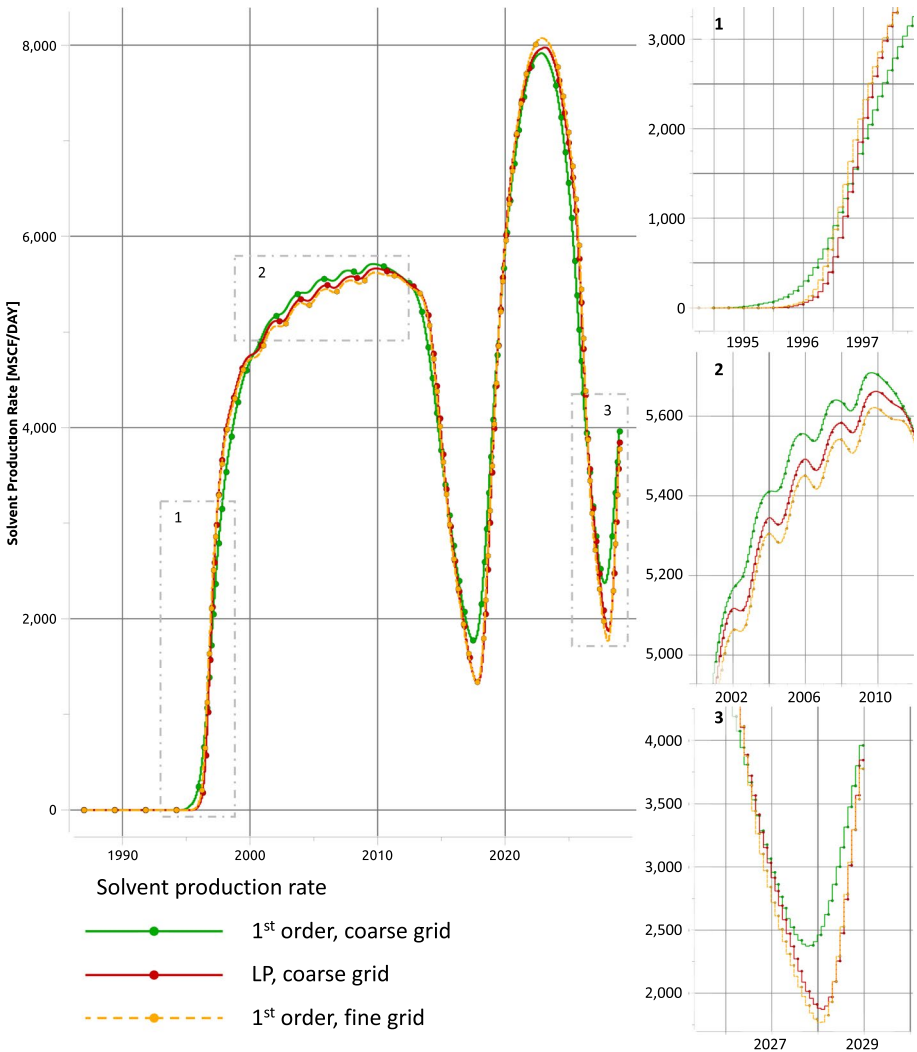
left corner and a producer in the upper right. Both wells are under rate control with a target rate of 12,000 STB/day. The BHP target is set to 1000 psia for the producer and 10000 psia for the injector. The fluid properties are adopted from the well-known SPE5 benchmark (Killough and Kossack 1987). On this reservoir, we will run both first- and second-order methods. As we do not know the “true” solution, we will also run first-order method on a refined grid ( $28 \times 28 \times 3$  grid cells) and view it as the reference solution. Note, we refine the grid only in  $X$  and  $Y$  direction, since fluid flow mostly happening in this direction.

The schedule in this test case emulates WAG (water alternating gas) injection with 1 and 5-year cycles. During the first 2 years, we perform depletion of the reservoir and only produce without injecting. After the first 2 years, we do 20 years of WAG injection with 1 year cycle, meaning that we inject water for one full year and follow it with gas injection for the next full year, both with a constant rate. Afterward, we change the schedule and inject gas for 5 consecutive years and water for the next 5 years. The simulation is finished with 5 more years of gas injection, which gives us 15 years of WAG injection with 5-year cycle. The schedule has two cycle schedules in one, first with the short and second with the long span, which will help us to show how the methods perform in different conditions.

Initially, the reservoir is filled with oil in the top layers and water in the bottom. The oil in the reservoir is light and has a certain fraction of gas mixed into it. The injected gas consists only of solvent, which is a mix of several components, but mostly consists of methane.

Let us now examine the production rate curves for all the components. We will start with the solvent production rate, see Fig. 2. Three curves in the figure correspond to the result produced by the first- and second-order methods on the coarse grid and a reference first-order method on the refined grid. Throughout the whole simulation time, we see an agreement between the second-order method and the reference method. We zoom into three parts of the production rate curve: first, to the time when the solvent first reaches the producer, second, the waves of 1-year WAG injection, and third, to the last wave of the 5-year WAG injection. In the first, we see that both the second-order method and first-order on fine grid method predict later arrival time and sharper front for the solvent production.





**Fig. 2** Solvent production rate: whole simulation time on the left and three zoom-ins on the right

This agrees very well with the results obtained for the simple immiscible fluid displacement cases, presented in Klöfkorn et al. (2017). We see a 3-month difference between the first- and second-order methods' curves reaching 200 MSCF/day level of production. In the two later stages, we also see a noticeable difference in the results: the second-order method tends to show later arrival of the fronts. It also predicts higher local maximums and lower local minimums, which is expected as the second-order method reduces smearing.

Gas and oil production rates are shown in Fig. 3. We plot them together because they share one interesting feature. We observe a gas wave formed due to the pressure drop in the reservoir during the first 2 years of depletion. The gas that was initially dissolved in the oil was realized and formed a separate sharp gas wave. And therefore, around 1995 we observe a pick in gas production and a drop in oil production. The first-order method on the refined grid confirms the front position obtained by the second-order method.

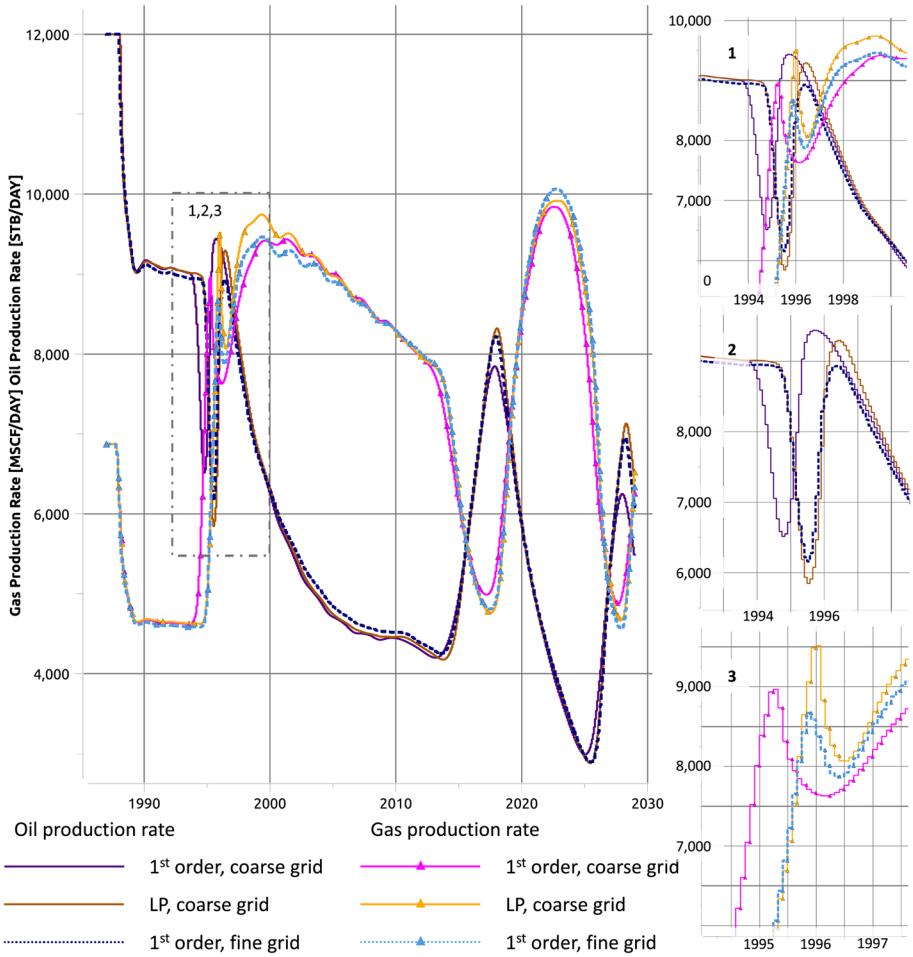
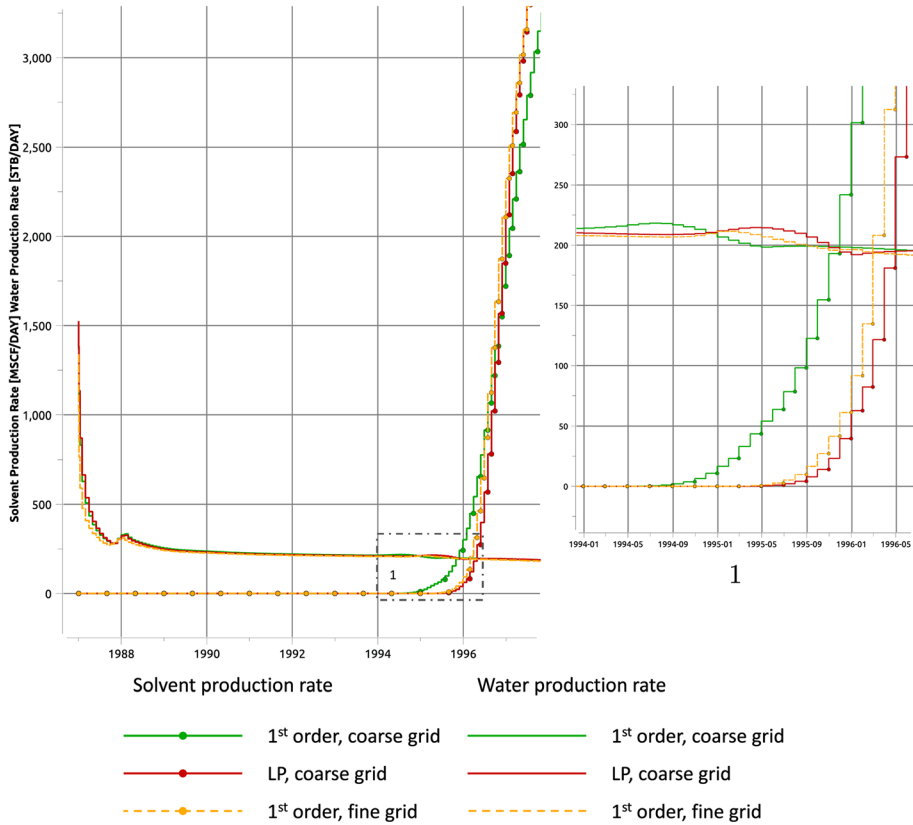


Fig. 3 Gas and oil production rates during the whole simulation time on the left and zoom-ins on the right

The second-order method, therefore, outperforms the first order in the accuracy of the positioning of the sharp front.

Let us examine closely the water production rate, see Fig. 4. Here we see two “waves”: first at the beginning of 1988 and second at the beginning of 1995. However, the second-order method gives identical results to the first order on the first wave and predicts a later arrival of the waterfront for the second wave. The reason for it is the nature of those peaks in water production. The water production rate is plotted together with the solvent production rate, such that we know when the injected fluid reaches the producer. And in our case, it happens only in 1995 (May 1995 for the first order and December 1995 for the second to reach 50 STB/day production rate). This means that before that time we produce the reservoir water and its behavior, especially during the first 2 years when there is no injection, is pressure driven. And as we do not improve the pressure calculations, the production curves for the first- and second-order methods are identical. However, when the injected water reaches the producer, we see the transport



**Fig. 4** Production rates of water and solvent during the whole simulation on the right and zoom in on the left

phenomena and the second-order method predicts the later arrival of the waterfront, and the reference method agrees with it.

In summary, whenever the transport phenomena were dominant for all production rate curves, we saw an improved front position and reduced smearing for the second-order method compared to the first-order. The result was verified by the result of the first-order method on the refined grid.

### 3.2 Simulation on the Norne Field

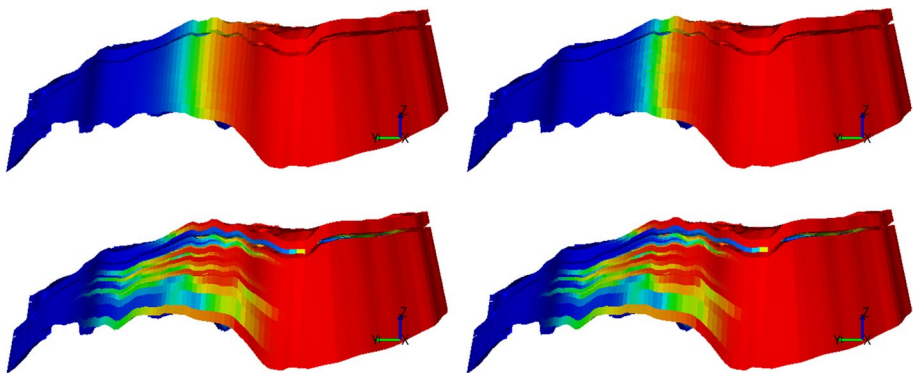
The Norne field is an oil and gas sandstone reservoir on the Norwegian continental shelf. It is one of the few fields where the simulation model together with production data was published under an open content license. Since then it became a benchmark field model.

In order to compare the performance of the first- and second-order FV methods on a realistic reservoir, we are going to use the benchmark simulation model with a grid block size of approximately 100 m and a refined Norne model with a grid block size of approximately 50 m. Since this is a real reservoir, we do not know the analytical solution even to the simplest production/injection schedule. That is why we will use the Norne field with the refined grid, such that we can compare the results of the second-order method with

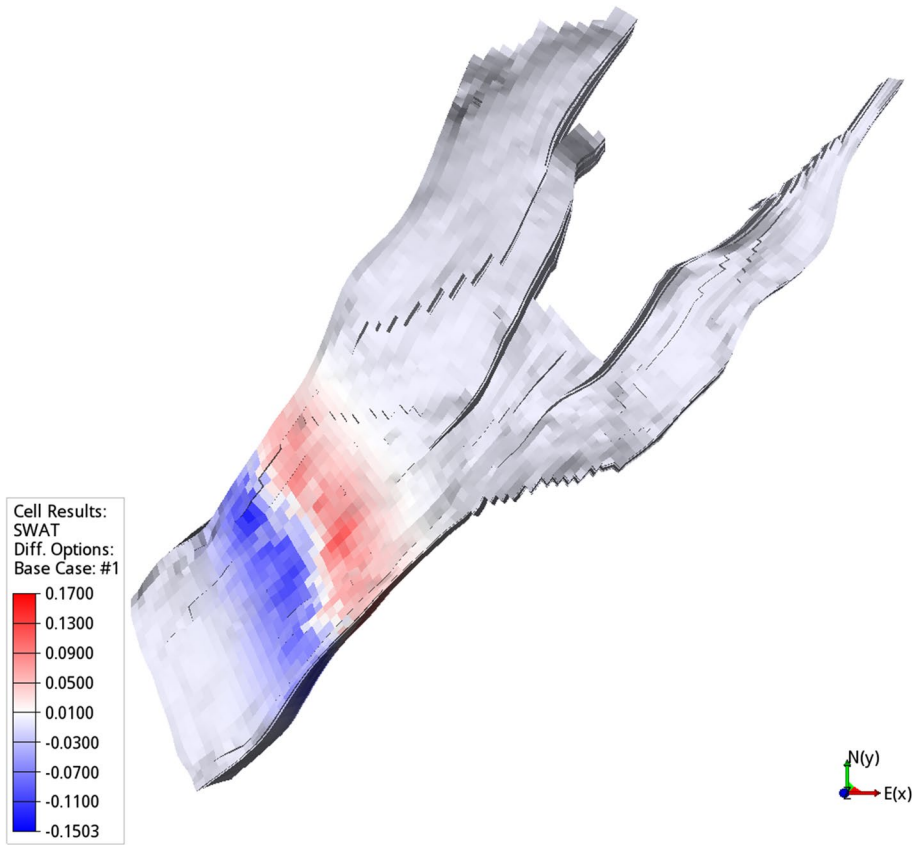
the results of the first-order method on the refined grid. The previous study on the simple geometry reservoirs (Kvashchuk et al. 2019) suggests that the second-order method accuracy in the front position detection can be compared with the first-order only on the refined grid.

Following Lie et al. (2020) we construct a simple piston-type injection example on the Norne field. In order to do so, first we use homogeneous porosity and permeability throughout the whole reservoir and neglect regions and fault multipliers, that are used in the original model. Second, we fill the whole reservoir with “red” fluid and inject “blue” fluid through the east side. Red and blue liquids have the same properties such that we can isolate the fluid displacement effects. Third, we put two production wells on the other side such that the injected fluid can flow through the reservoir. Finally, we run simulations for almost 25 years (9000 days) for both the first- and second-order methods. The field after 4.5 years is shown in Fig. 5. The top row corresponds to the homogeneous case and the bottom to the heterogeneous. The difference is more visible for the homogeneous grid, which also agrees with the curves in Fig. 7. The curves in the figure represent the ratio of the red fluid produced in the reservoir to the total fluid volume. Figure 6 shows the actual difference between the results on each grid cell in December 2002, 5 years and 1 month after the simulation started. The curves in Fig. 7 agrees with the results reported in Lie et al. (2020). We see that the second-order finite volume method reduces the smearing effect and that the reduction is more pronounced on the homogeneous media.

We verify the results obtained above by running the same simulation on the refined Norne grid. We refine the original grid block, which is approximately 100 m, in  $X$  and  $Y$  directions, and get a refined Norne model with a grid block size of approximately 50 m. Since we are focusing on fluid displacement and it is happening in the  $X$ – $Y$  plane, we do not refine cells in the  $Z$  direction, where mostly gravity effects take place. The results on the refined grid confirm the behavior of the second-order method that we saw in Klöfkorn et al. (2017) and Kvashchuk et al. (2019) and in the previous example—it gives a sharper front and predicts its position closer to what is obtained with the first-order method on the refined grid, see Fig. 8. In Fig. 8 we see very good agreement between the higher order method and the first-order method on the refined grid in predicting the arrival time of the fluid front. However, when it comes to later stages of production (around the year 2020),



**Fig. 5** Homogeneous (top) and heterogeneous (bottom) Norne field in the middle of production, east view. The results for the first order are on the left, and for the second order on the right side. Here we can clearly see the reduced smearing for the second-order method on the homogeneous Norne



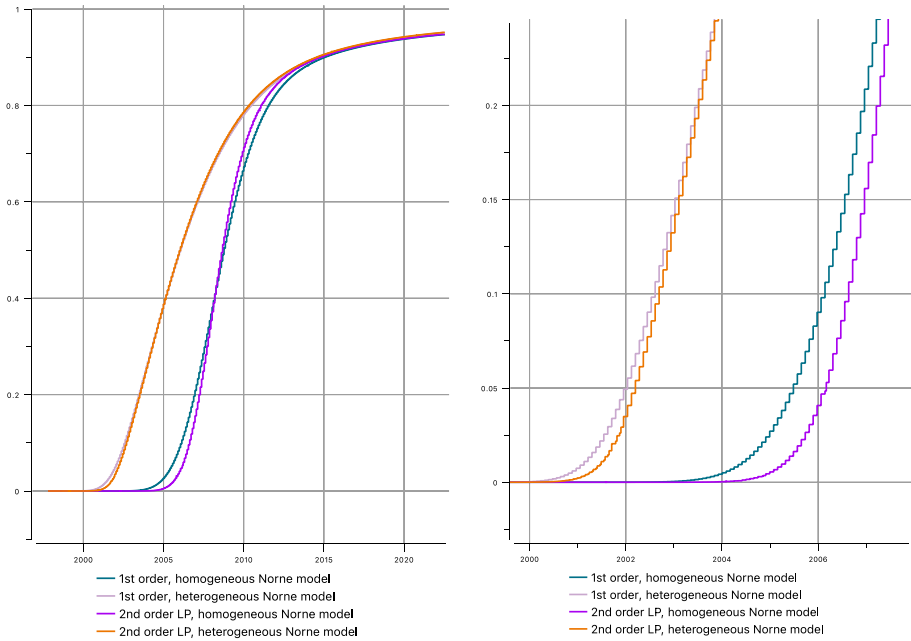
**Fig. 6** Difference between the results of the first- and second-order methods

the first- and second-order methods on the coarse grid produce the same result, while the first-order on the refined grid predicts a lower number. The results of the first- and second-order methods on the same grid differ only in how much the front is spread: it is spread more for the first-order method and is sharper (less spread) for the second-order. That is why after the front is resolved they agree and stay on the same level.

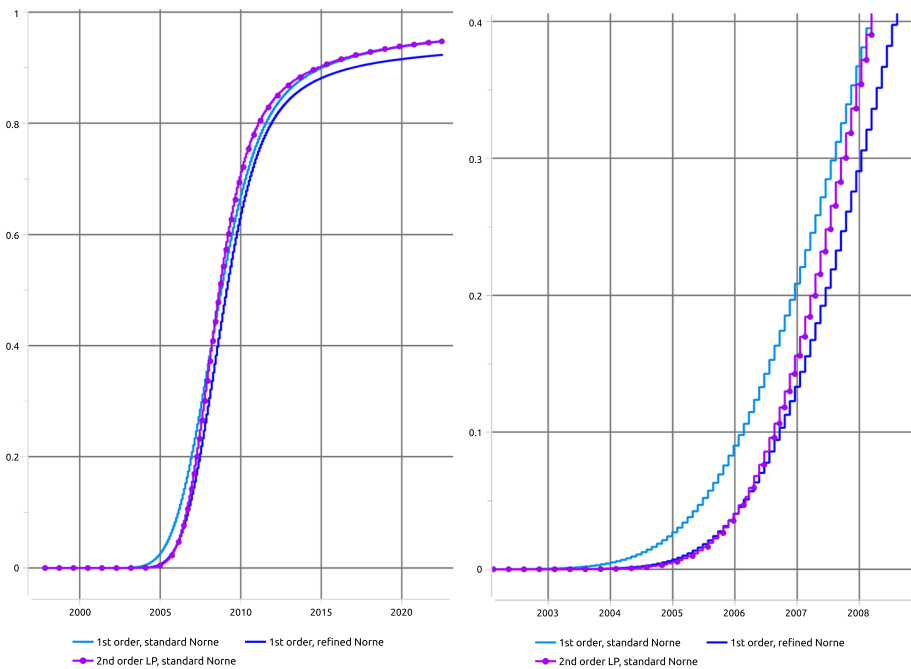
### 3.3 Carbon Dioxide Injection on Norne

In order to test the performance of the second-order method in more realistic conditions, we set up a CO<sub>2</sub> injection test case. Even though the Norne field is not a primary candidate for the CO<sub>2</sub> injection, it can give valuable insight into how the methods compare to the realistic field scale model.

For ease of simulation, we will have only two wells—one injector and one producer. However, the rest of the complexity of the field is present—we have faults, regions, heterogeneity, etc. Also, we change the initial conditions—in this test case, the reservoir is filled only with oil at the start of the simulation. The CO<sub>2</sub> is injected at a constant rate of 100,000 sm<sup>3</sup>/day from the start of the simulation and for consecutive 5000 days. We assume the injected CO<sub>2</sub> is miscible in oil when its gas fraction is more than 0.01. We use



**Fig. 7** The ratio of red liquid produced compared to the volume of total liquids produced in the Norne reservoir. The full simulation time is on the left, and on the right we zoom in from 2000 to 2006



**Fig. 8** The ratio of red liquid produced compared to the volume of total liquids produced in the standard and refined Norne reservoir. The full simulation time is on the left and from 2003 to 2008—on the right

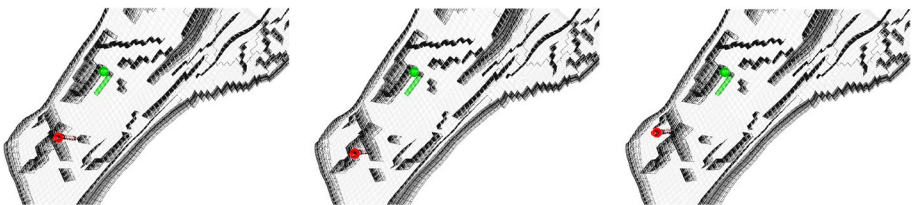
the “MISC” keyword (Baxendale 2022) to define that fluid is immiscible at low concentrations and switches to miscible behavior when the concentration increases. For all simulated scenarios, the production well is placed in the middle of the reservoir and is controlled by a liquid rate target of 4000  $\text{sm}^3/\text{day}$ . It is a horizontal well in layer 13 with wellhead coordinates  $I = 13$  and  $J = 39$ . The injector well will be placed in three different places: first, the two wells are in the same compartment meaning that there are no faults between them, second—the injector and producer are separated by just one fault; and third, the injector well is placed in the corner of the reservoir surrounded by faults, see Fig. 9.

The resulting  $\text{CO}_2$  (solvent) production rate curves are presented in Fig. 10. In the first scenario, solvent reaches the producer faster than in the other two as there are no faults between the two wells. We also observe 105 days difference between the first- and second-order methods’ prediction of when the solvent production rate reaches 1000  $\text{sm}^3/\text{day}$ . In two other cases, when we have faults between the injector and producer, we see the same trend—the second-order method predicts the later arrival of the carbon dioxide; however, the difference is less: solvent production rate reaches 1000  $\text{sm}^3/\text{day}$  with 70 days difference for the “behind one fault” case and 65 days difference for the “corner” case. It shows that the increased complexity of the reservoir can overshadow the effects gained by using a higher-order computational method. However, we still observe a relevant difference in the front positioning when using the higher-order method.

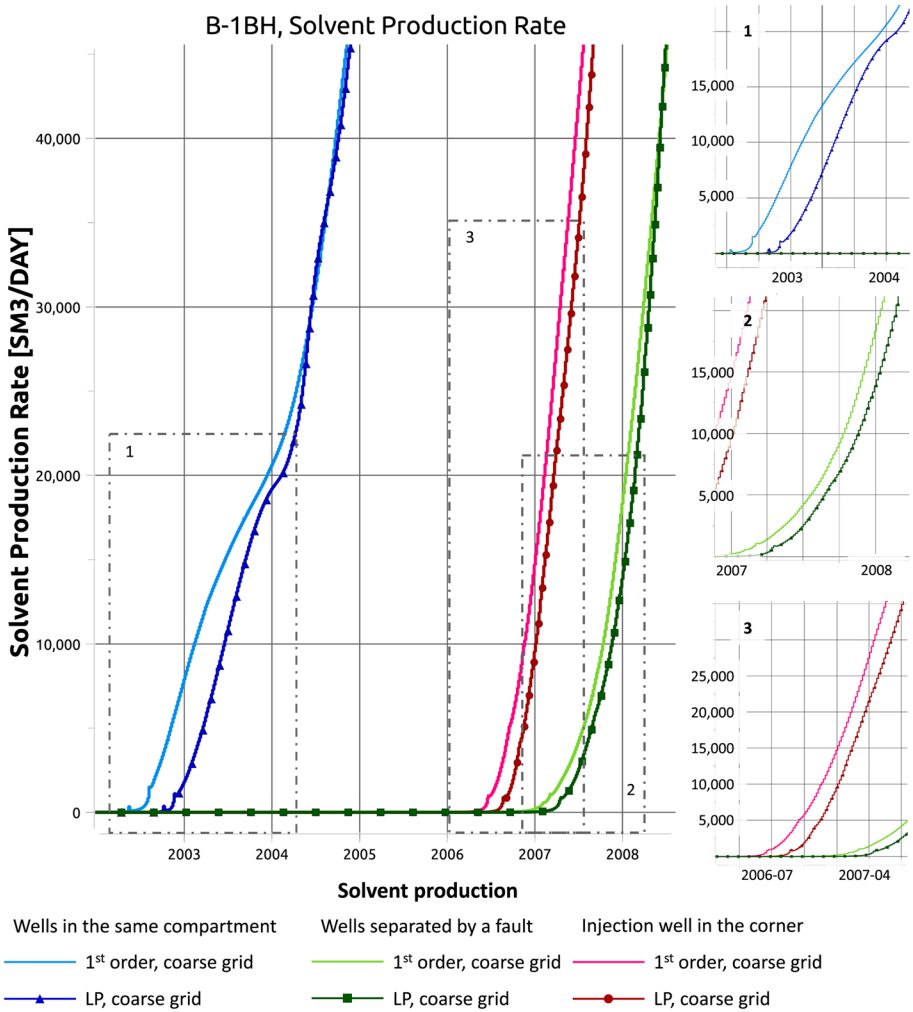
## 4 Conclusion

We showed that the presented second-order linear programming method predicts the fluid front position more accurately and decreases the numerical diffusion on realistic reservoir models. We also verified the results with the first-order method on the refined grid, both for the medium-sized reservoir and the Norne test case. In all the presented test cases, we saw that the second-order method would predict the position of the fluid front as accurately as the first order on the refined grid.

We also observed that the result depends on the type of fluid flow. When pressure effects dominated the fluid flow, we did not observe significant improvements, which is not surprising as we did not improve the accuracy of the pressure computation. In general, the more complex fluid flow we have, the harder it is to model the process and interpret the result. For example, in the simple scenario of carbon dioxide injection on a complex Norne field, we observed that the difference between the first- and second-order methods results depends on the number of faults between the injector and the producer. And for piston-type injection, the smearing was reduced more for



**Fig. 9** Positions of the wells in the Norne  $\text{CO}_2$  injection scenarios: left for the wells in the same compartment, middle—wells are separated by a fault, right—injection well in the corner



**Fig. 10** Solvent production rate for the three scenarios of solvent injection on Norne. The subplots zoom into the times of solvent arrival for each scenario: (1) Wells in the same compartment; (2) Wells separated by a fault; (3) Injection well in the corner

the homogeneous grid than for the heterogeneous. More testing in realistic settings is needed to answer under which conditions the higher-order methods are most helpful.

A significant advantage of the presented second-order method is the fact that it is implemented in the open reservoir simulator OPM Flow (Open porous media), where using the higher order method is a simple task of switching the flag when starting the simulation. This allows for immediate usage and further testing in realistic settings and facilitates future improvements and extensions of the presented method. This means faster learning and more informed decisions when modeling the fluid flow in the reservoir.



## Appendix 1. Modeling Equations

### Black-Oil Model

The black-oil model is a three-phase three-component model, which is an industry-standard model for the reservoir processes (Chen et al. 2006). Let us state below the assumptions and equations that are governing the model. As phases and components share the same names (gas, water, and oil), we will use different indices in order to distinguish between them: index  $\alpha \in \{w, g, o\}$  (“water”, “gas” and “oil”) for phases and index  $\kappa \in \{W, G, O\}$  (“Water”, “Gas” and “Oil”) for components. The miscibility assumptions are:

- water and gas phases are immiscible,
- the water phase is composed only of water,
- gas phase may contain vaporized oil,
- the oil phase is assumed to be a mixture of gas and oil components.

The densities of the phases are determined by so-called formation volume factors, which is the ratio of the phase volume measured at reservoir conditions  $\rho_\alpha$  to the phase volume measured at standard conditions  $\rho_{asc}$ :

$$B_\alpha := \frac{\rho_\alpha}{\rho_{asc}}. \quad (19)$$

Here and later on, we will use subscript *sc* to indicate that quantity was measured at standard condition.

The water density can be computed using just the water volume formation factor. In order to compute oil and gas densities, we need to introduce gas solubility  $R_s$  (also called dissolved gas-oil ratio), which is the volume of gas at standard conditions  $V_{Gsc}$  divided by the volume of oil at standard conditions  $V_{Osc}$  given that they were both obtained from some amount of oil phase at reservoir conditions:

$$R_s := \frac{V_{Gsc}}{V_{Osc}}, \quad (20)$$

and oil volatility  $R_v$ , which is the volume of oil at standard conditions  $V_{Osc}$  divided by the volume of gas at standard conditions  $V_{Gsc}$  given that they were both obtained from some amount of gas phase at reservoir conditions:

$$R_v := \frac{V_{Osc}}{V_{Gsc}}. \quad (21)$$

This allows for calculating all quantities required for the mass-conservation equations for each component. All three equations share the same structure:

$$\frac{\partial}{\partial t}(\phi A_\kappa) + \nabla \cdot v_\kappa = Q_\kappa, \quad (22)$$

where  $\phi$  is the porosity,  $A_\kappa$ —the accumulation term,  $v_\kappa$  and  $Q_\kappa$  is the mass flux and source or sink term of the phase  $\kappa$  respectively. For each component  $\kappa$  the accumulation term and the mass flux are:

$$\begin{aligned}
 A_W &= \frac{S_w}{B_w}, & v_W &= \frac{u_w}{B_w}, \\
 A_O &= \frac{S_o}{B_o} + \frac{R_v S_g}{B_g}, & v_O &= \frac{u_o}{B_o} + \frac{R_v u_g}{B_g}, \\
 A_G &= \frac{S_g}{B_g} + \frac{R_s S_o}{B_o}, & v_G &= \frac{u_g}{B_g} + \frac{R_s u_o}{B_o}.
 \end{aligned}
 \tag{23}$$

The phase flux  $u_\alpha$  is determined by the standard multi-phase Darcy equation (Nordbotten and Celia 2011), i.e.

$$u_\alpha = -\frac{k_{r,\alpha}}{\mu_\alpha} k (\nabla p_\alpha - \rho_\alpha g),
 \tag{24}$$

where  $k$  is the permeability,  $g$  - gravity, and  $k_{r,\alpha}$ ,  $\mu_\alpha$ ,  $\rho_\alpha$ ,  $p_\alpha$  are relative permeability, viscosity, density and pressure of the phase  $\alpha$  respectively.

### The Model with Solvent

In this study, we use the black-oil model extended with the solvent equation instead of a fully compositional model. This choice has been made in OPM to reduce the computational time and to ease the implementation of the CO<sub>2</sub>-EOR injection scenario (Sandve et al. 2018). However, the proposed second-order methods are not limited to this model and in principle can be used in any other formulation. To extend the black-oil model we add the following equation:

$$A_s = \frac{S_s}{B_s}, \quad v_s = \frac{u_s}{B_s}.
 \tag{25}$$

Additionally, the presence of solvent influences relative permeability and viscosity, and these become effective relative permeability  $k_{rae}$  and effective viscosity  $\mu_{ae}$ . The effective properties depend on the miscibility factor  $M$  and are defined as follows:

$$k_{roe} = M \cdot \frac{S_o - S_{or}}{S_n - S_{gc} - S_{or}} \cdot k_{rm} + (1 - M) \cdot k_{ro},
 \tag{26}$$

$$k_{rge} = M \cdot \frac{S_g + S_s - S_{gc}}{S_n - S_{gc} - S_{or}} \cdot k_{rm} + (1 - M) \cdot \frac{S_g}{S_g + S_s} k_{rgt},
 \tag{27}$$

$$k_{rse} = M \cdot \frac{S_g + S_s - S_{gc}}{S_n - S_{gc} - S_{or}} \cdot k_{rm} + (1 - M) \cdot \frac{S_s}{S_g + S_s} k_{rgt}.
 \tag{28}$$

Here  $k_{rgt}$  is the total relative permeability of the gas phase,  $S_n$  is the total hydrocarbon saturation and  $S_{or}$  and  $S_{gc}$  is the residual oil saturation and the critical gas saturation, respectively.

Effective viscosities are calculated using the Todd-Longstaff mixing parameter  $w$  (Todd and Longstaff 1972):

$$\mu_{oe} = \mu_o^{1-w} \cdot \mu_{mos}^w, \quad \mu_{ge} = \mu_g^{1-w} \cdot \mu_{msg}^w, \quad \mu_{se} = \mu_s^{1-w} \cdot \mu_m^w.
 \tag{29}$$

The fully mixed viscosities for the oil and solvent mixture  $\mu_{mos}$ , the gas and solvent mixture  $\mu_{msg}$ , and the oil, gas, and solvent mixture  $\mu_m$  are computed using the standard mixing rule (Todd and Longstaff 1972).

**Acknowledgements** Anna Kvashchuk acknowledges Alexey Khrulenko for help with the refined Norne model and many fruitful discussions. Anna Kvashchuk also acknowledges Equinor ASA for support. Robert Klöforn acknowledges the INTPART project INSPIRE (274883). Anna Kvashchuk and Robert Klöforn acknowledge the Research Council of Norway and the industry partners, ConocoPhillips Skandinavia AS, Aker BP ASA, Eni Norge AS, Equinor ASA, Neptune Energy Norge AS, Lundin Norway AS, Halliburton AS, Schlumberger Norge AS, Wintershall Norge AS, and DEA Norge AS, of The National IOR Centre of Norway for support. Tor Harald Sandve acknowledges the CLIMIT program (617115) and Equinor ASA for support. The authors thank three anonymous reviewers for thoughtful comments that helped improve the manuscript.

**Funding** Open access funding provided by University of Stavanger & Stavanger University Hospital. Authors Anna Kvashchuk and Robert Klöforn received support from the Research Council of Norway and the industry partners ConocoPhillips Skandinavia AS, Aker BP ASA, Eni Norge AS, Equinor ASA, Neptune Energy Norge AS, Lundin Norway AS, Halliburton AS, Schlumberger Norge AS, Wintershall Norge AS, and DEA Norge AS, of The National IOR Centre of Norway (230303). Author Anna Kvashchuk has received research support from Equinor ASA. The contribution of Robert Klöforn was also supported by the INTPART project INSPIRE (274883). The contribution of Tor Harald Sandve was supported by the CLIMIT program (617115). The authors have no relevant financial or non-financial interests to disclose. All authors contributed to the study's conception and design. Data collection and analysis were performed by Anna Kvashchuk, Robert Klöforn, and Tor Harald Sandve. The first draft of the manuscript was written by Anna Kvashchuk, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript. The datasets generated during and/or analysed during the current study are available in the second-order-opm-tests repository, <https://github.com/kvashchuka/second-order-opm-tests>.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Aavatsmark, I.: Interpretation of a two-point flux stencil for skew parallelogram grids. *Comput. Geosci.* **11**(3), 199–206 (2007)
- Alyae, S., Keilegavlen, E., Nordbotten, J.M.: Analysis of control volume heterogeneous multiscale methods for single phase flow in porous media. *Multiscale Model. Simul.* **12**(1), 335–363 (2014)
- Baxendale, D.: OPM flow reference manual (2022-10). Open Porous Media Initiative (2022)
- Bell, J., Shubin, G.: Higher-order godunov methods for reducing numerical dispersion in reservoir simulation. In: *SPE Reservoir Simulation Symposium*. OnePetro (1985)
- Blunt, M., Rubin, B.: Implicit flux limiting schemes for petroleum reservoir simulation. *J. Comput. Phys.* **102**(1), 194–210 (1992)
- Celia, M.A., Nordbotten, J.M.: Practical modeling approaches for geological storage of carbon dioxide. *Groundwater* **47**(5), 627–638 (2009)
- Chen, L., Li, R.: An integrated linear reconstruction for finite volume scheme on unstructured grids. *J. Sci. Comput.* **68**(3), 1172–1197 (2016)
- Chen, W., Durlofsky, L., Engquist, B., Osher, S.: Minimization of grid orientation effects through use of higher order finite difference methods. *SPE Adv. Technol. Ser.* **1**(02), 43–52 (1993)
- Chen, Z., Huan, G., Ma, Y.: *Computational Methods for Multiphase Flows in Porous Media*, vol. 2. SIAM, Philadelphia (2006)

- Class, H., Ebigo, A., Helmig, R., Dahle, H.K., Nordbotten, J.M., Celia, M.A., Audigane, P., Darcis, M., Ennis-King, J., Fan, Y., et al.: A benchmark study on problems related to CO<sub>2</sub> storage in geologic formations. *Comput. Geosci.* **13**(4), 409–434 (2009)
- Coats, K.H., Thomas, L., Pierson, R.: Compositional and black oil reservoir simulation. In: *SPE Reservoir Simulation Symposium*. OnePetro (1995)
- Cockburn, B., Shu, C.-W.: TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. II. general framework. *Math. Comput.* **52**(186), 411–435 (1989)
- Contreras, F., Lyra, P., Souza, M., Carvalho, Dd.: A cell-centered multipoint flux approximation method with a diamond stencil coupled with a higher order finite volume method for the simulation of oil-water displacements in heterogeneous and anisotropic petroleum reservoirs. *Comput. Fluids* **127**, 1–16 (2016)
- Durlofsky, L.J., Engquist, B., Osher, S.: Triangle based adaptive stencils for the solution of hyperbolic conservation laws. *J. Comput. Phys.* **98**(1), 64–73 (1992)
- ECLIPSE™ 2014.2 (2014). <http://www.software.slb.com/products/eclipse>
- Eymard, R., Gallouët, T., Herbin, R.: Finite volume methods. In: *Handbook of Numerical Analysis*, vol. 7, pp. 713–1018 (2000)
- Flemisch, B., Darcis, M., Erbertseder, K., Faigle, B., Lauser, A., Mosthaf, K., Müthing, S., Nuske, P., Tatomir, A., Wolff, M., et al.: Dumux: Dune for multi-phase, component, scale, physics, flow and transport in porous media. *Adv. Water Resour.* **34**(9), 1102–1112 (2011)
- Geiger-Boschung, S., Matthäi, S.K., Niessner, J., Helmig, R.: Black-oil simulations for three-component, three-phase flow in fractured porous media. *SPE J.* **14**(02), 338–354 (2009)
- Harten, A.: High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.* **135**(2), 260–278 (1997)
- Jiang, G.-S., Shu, C.-W.: Efficient implementation of weighted ENO schemes. *J. Comput. Phys.* **126**(1), 202–228 (1996)
- Jung, Y., Pau, G.S.H., Finsterle, S., Pollyea, R.M.: Tough3: a new efficient version of the tough suite of multiphase flow and transport simulators. *Comput. Geosci.* **108**, 2–7 (2017)
- Killough, J.E., Kossack, C.A.: Fifth comparative solution project: evaluation of miscible flood simulators. In: *SPE Symposium on Reservoir Simulation* (1987). <https://doi.org/10.2118/16000-MS>
- Klemetsdal, Ø.S., Rasmussen, A.F., Møyner, O., Lie, K.-A.: Efficient reordered nonlinear gauss-seidel solvers with higher order for black-oil models. *Comput. Geosci.* **24**(2), 593–607 (2020). <https://doi.org/10.1007/s10596-019-09844-5>
- Klöforn, R., Kvashchuk, A., Nolte, M.: Comparison of linear reconstructions for second-order finite volume schemes on polyhedral grids. *Comput. Geosci.* (2017). <https://doi.org/10.1007/s10596-017-9658-8>
- Kvashchuk, A., Klöforn, R., Sandve, T.H.: Comparison of higher order schemes on complicated meshes and reservoirs. In: *SPE Reservoir Simulation Conference*. OnePetro (2019)
- Lake, L.W.: *Enhanced oil recovery* (1989)
- Lamine, S., Edwards, M.G.: Multidimensional upwind schemes and higher resolution methods for three-component two-phase systems including gravity driven flow in porous media on unstructured grids. *Comput. Methods Appl. Mech. Eng.* **292**, 171–194 (2015)
- Lauser, A., Rasmussen, A., Sandve, T., Nilsen, H.: Local forward-mode automatic differentiation for high performance parallel pilot-level reservoir simulation. In: *ECMOR XVI-16th European Conference on the Mathematics of Oil Recovery*, vol. 2018, pp. 1–12. European Association of Geoscientists and Engineers (2018)
- Leveque, R.J.: *Finite Volume Methods for Hyperbolic Problems*, vol. 31. Cambridge University Press, Cambridge (2002)
- Lichtner, P.C., Hammond, G.E., Lu, C., Karra, S., Bisht, G., Andre, B., Mills, R.T., Kumar, J., Frederick, J.M.: *PFLORAN user manual*. Technical report (2019). <http://documentation.pfloran.org>
- Lie, K.-A.: *An Introduction to Reservoir Simulation Using MATLAB/GNU Octave: User Guide for the MATLAB Reservoir Simulation Toolbox (MRST)*. Cambridge University Press, Cambridge (2019). <https://doi.org/10.1017/9781108591416>
- Lie, K.-A., Mykkeltvedt, T.S., Møyner, O.: A fully implicit WENO scheme on stratigraphic and unstructured polyhedral grids. *Comput. Geosci.* **24**(2), 405–423 (2020)
- Liu, X.-D., Osher, S., Chan, T.: Weighted essentially non-oscillatory schemes. *J. Comput. Phys.* **115**(1), 200–212 (1994)
- May, S., Berger, M.: Two-dimensional slope limiters for finite volume schemes on non-coordinate-aligned meshes. *SIAM J. Sci. Comput.* **35**(5), 2163–2187 (2013)
- Mykkeltvedt, T.S., Raynaud, X., Lie, K.-A.: Fully implicit higher-order schemes applied to polymer flooding. *Comput. Geosci.* **21**(5), 1245–1266 (2017)

- Mykkeltvedt, T.S., Gasda, S.E., Sandve, T.H.: CO<sub>2</sub> convection in hydrocarbon under flowing conditions. *Transp. Porous Media* **139**(1), 155–170 (2021)
- Nocedal, J., Wright, S.: *Numerical Optimization*. Springer, New York (2006)
- Nordbotten, J.M., Celia, M.A.: *Geological Storage of CO<sub>2</sub>: Modeling Approaches for Large-Scale Simulation*. Wiley, New Jersey (2011)
- Rasmussen, A.F., Sandve, T.H., Bao, K., Lauser, A., Hove, J., Skaflestad, B., Klöfkorn, R., Blatt, M., Rustad, A.B., Sævareid, O., Lie, K.-A., Thune, A.: The open porous media flow reservoir simulator. *Comput. Math. Appl.* **81**, 159–185 (2021). <https://doi.org/10.1016/j.camwa.2020.05.014>
- Rubin, B., Blunt, M.: Higher-order implicit flux limiting schemes for black oil simulation. In: *SPE Symposium on Reservoir Simulation*. OnePetro (1991)
- Rubin, B., Edwards, M.: Extension of the TVD midpoint scheme to higher-order accuracy in time. In: *SPE Symposium on Reservoir Simulation*. OnePetro (1993)
- Sammon, P.H., Kurihara, M., Jialing, L.: Applying high-resolution numerical schemes in reservoirs described by complex corner-point grids. In: *SPE Reservoir Simulation Symposium*. OnePetro (2001)
- Sandve, T.H., Rasmussen, A., Rustad, A.B.: Open reservoir simulator for CO<sub>2</sub> storage and CO<sub>2</sub>-EOR. In: *14th Greenhouse gas control technologies conference Melbourne*, pp. 21–26 (2018)
- Todd, M., Longstaff, W., et al.: The development, testing, and application of a numerical simulator for predicting miscible flood performance. *J. Pet. Technol.* **24**(07), 874–882 (1972)
- Trangenstein, J.A., Bell, J.B.: Mathematical structure of the black-oil model for petroleum reservoir simulation. *SIAM J. Appl. Math.* **49**(3), 749–783 (1989)
- Voskov, D.V., Tchelepi, H.A.: Comparison of nonlinear formulations for two-phase multi-component EoS based simulation. *J. Pet. Sci. Eng.* **82**, 101–111 (2012)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.