A STUDY OF THE MODELLING ERROR IN TWO OPERATOR SPLITTING ALGORITHMS FOR POROUS MEDIA FLOW

K. BRUSDAL, H. K. DAHLE, K. HVISTENDAHL KARLSEN, T. MANNSETH

ABSTRACT. Operator splitting methods are often used to solve convection-diffusion problems of convection dominated nature. However, it is well known that such methods can produce significant (splitting) errors in regions containing self sharpening fronts. To amend this shortcoming, corrected operator splitting methods have been developed. These approaches use the wave structure from the convection step to identify the splitting error. This error is then compensated for in the diffusion step. The main purpose of the present work is to illustrate the importance of the correction step in the context of an inverse problem. The inverse problem will consist of estimating the fractional flow function in a one-dimensional saturation equation.

1. Introduction

We are here interested in the initial-boundary value problem associated with nonlinear convection-diffusion equations of the form

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(f(u) - \varepsilon d(u)\frac{\partial u}{\partial x}\right) = 0, \qquad (x,t) \in (a,b) \times (0,T), \tag{1}$$

where $u = u(x,t)$ denotes the unknown, $f = f(u)$ is the flux function, $d = d(u)$ is the diffusion function and $\varepsilon > 0$ is a scaling parameter. Convection-diffusion equations arise in a variety of applications. We here consider two-phase, immiscible and incompressible flow of oil and water in porous media. In this context, u is the water saturation, $f(u)$ is the fractional flow function and $d(u)$ is the capillary diffusion function.

Due to the nonlinear nature of the differential equation (1), there will in general be no analytical expression for $u(x,t)$. Hence we must rely on numerically techniques to obtain the solution. When (1) is convection dominated, that is, when ε is small compared with

Key words and phrases. Operator splitting, two-phase flow, saturation equation, modelling error, inverse problem.

Brusdal and Mannseth have been supported by the Research Council of Norway under grant 107615/420. Karlsen has been supported by VISTA, a research cooperation between the Norwegian Academy of Science and Letters and Den norske stats oljeselskap a.s. (Statoil).

other scales in (1), conventional methods usually exhibit some combination of difficulties, ranging from non-physical oscillations to severe numerical diffusion at the trailing end of moving fronts. To overcome such difficulties we make use of *operator splitting*. In the present context, operator splitting means to split the convection-diffusion equation (1) into a hyperbolic equation and a parabolic equation, each of which is solved separately in an alternate fashion. Furthermore, each step of such an algorithm is fully discretized by accurate and efficient numerical schemes developed especially for hyperbolic conservation laws and parabolic heat type equations, respectively. The operator splitting approach has been taken by many authors, [2, 4, 5, 6, 7, 8, 9, 10, 12, 15, 16, 17] (see the references cited therein for a more complete list of relevant papers).

In the case where $d(u) \equiv 1$, the physical front widths should be of order $\mathcal{O}(\varepsilon)$. However, it is easy to see that diffusive splitting errors will lead to front widths of size $\mathcal{O}(\sqrt{\varepsilon \Delta t})$ when the splitting time step, Δt, is large ($\gg \varepsilon$). This type of splitting error comes from the fact that the entropy condition forces the hyperbolic solver to throw away certain "information" that controls the structure of the self-sharpening fronts. It is however possible to compensate for this loss of information, which manifests itself in the form of residual flux terms. For example, assume that the solution of (1) is a moving front. If the time step is large enough, the convection step will generate a shock with left and right limits, say, u_l and u_r, respectively. We can then identify a residual flux term associated with this discontinuity; $f_{\text{res}}(u) = f(u) - f_c(u)$, where $f_c(u)$ denotes the correct envelope (dictated by the entropy condition) of $f(u)$ on the interval bounded by u_l and u_r. There are several ways to take the residual flux term $f_{\text{res}}(u)$ into account. We can perform a separate correction step after the diffusion step. Correction is then calculated by solving the "residual" conservation law $v_t + f_{\text{res}}(v)_x = 0$ over a time interval $(0, \tau]$, where $\tau > 0$ is some parameter that has to be chosen (see [16] for details). Another approach is to include the residual term in the equation modelling diffusion, that is, instead of solving the equation $w_t - \varepsilon(d(w)w_x)_x = 0$, we solve $w_t + (f_{\text{res}}(w) - \varepsilon d(w)w_x)_x = 0$. This equation contains the necessary information needed to produce the correct structure of the front. We will here rely on the latter approach, which we shall refer to as *corrected operator splitting* (COS). Splitting methods which do not account for residual flux terms will be denoted by OS.

The idea of using a residual flux term in the diffusion step was introduced in [10] (and further developed in [4, 6, 7, 8]) in the context of two-phase flow with an established front as initial data. In this setting the residual term can be determined a priori, which means that the convection step is to solve $v_t + f_c(v)_x = 0$ (where f_c is linear in the front region). This fact was exploited in [4, 6, 7, 8, 10] in the sense that the modified method of characteristics was used for the convection step. For general initial data this approach may put severe restrictions on the time step. However, in [15, 16] it was observed that by using front tracking [14] to solve the nonlinear conservation law $v_t + f(v)_x = 0$ it is possible to dynamically construct residual flux terms. Consequently, the COS approach makes sense in general. The calculation of these residual terms is a direct consequence of the fact that front tracking is based on solving Riemann problems. Instead of front tracking, a second order Godunov method will here be used in the convection step to illustrate that other techniques can be utilised in the construction of residual flux terms, see also [15].

In what follows, differences in the Godunov based OS and COS methods will be illustrated in the context of an *inverse problem.* The inverse problem will consist of recovering the flux function using COS and OS as forward models in the inversion procedure. Note that in general it is not possible to define the residual flux term *a priori* in the inversion procedure since the representation of the flux function is changing throughout the procedure. This motivates the use of COS where the residual flux terms are constructed dynamically. The main purpose of the present work is to study the influence that the modelling errors in the OS and COS forwards models have on the recovery of the flux function.

The paper is organised as follows. In section 2 we give a description of the forward problem and the operator splitting techniques used to solve it. In section 3 we discuss the setup of the inverse problem. Furthermore, we present and discuss numerical experiments. Finally, in section 4 we make some concluding remarks.

2. The forward problem

2.1. **Numerical algorithms.** In this section we describe in some detail the operator splitting methods that are used to solve the forward equation (1). This equation must be equipped with appropriate initial and boundary conditions conditions

$$u\Big|_{t=0} = u_0(x), \qquad u\Big|_{x=a} = u_a \in \mathbb{R} \qquad u\Big|_{x=b} = u_b \in \mathbb{R},$$

where we assume that u_0 is a piecewise smooth function. Let $u^n(x) \approx u(x, t_n)$ denote the piecewise linear numerical solution of (1) at time level t_n, where $0 = t_0 < t_1 < \cdots t_N = T$ is a time discretization of $[0, T]$. Here the function $u^n = u^n(x)$ is piecewise linear with respect to a uniform grid with grid cell size $\Delta x > 0$. We determine u^{n+1} from u^n via the following OS algorithm.

Convection: Let $v(x, \Delta t_n)$, $\Delta t_n = t_{n+1} - t_n > 0$, denote the (entropy weak) solution of the hyperbolic conservation law

$$\frac{\partial v}{\partial t} + \frac{\partial}{\partial x} f(v) = 0, \qquad v(x, 0) = u^n(x). \tag{2}$$

The second order Godunov (slope limiter) method described in [13] is used to calculate this solution. The Godunov method uses a uniform (local) grid with grid cell size Δx_c and a time step $0 < \Delta t_c \leq \Delta t_n$ chosen according to the CFL condition, i.e., Δt_c is chosen so that $\max |f'| \frac{\Delta t_c}{\Delta x_c} = 1$.

Diffusion: Let $w(x, \Delta t_n)$ be the solution of the parabolic heat type problem

$$\frac{\partial w}{\partial t} - \varepsilon \frac{\partial}{\partial x} \left(d(w) \frac{\partial w}{\partial x} \right) = 0, \qquad w(x, 0) = v(x, \Delta t_n) \tag{3}$$

with the Dirichlet boundary data imposed at $x = a, b$. Numerically this solution is found using the Galerkin method on a uniform grid with grid cell $\Delta x_d \equiv \Delta x$ and the usual (piecewise linear) “hat” basis functions. The time discretization is done by the backward Euler method with a single time step of length Δt_n. Finally, we set

$$u^{n+1} = w(x, \Delta t_n).$$

In the constant diffusion case, $d(u) \equiv 1$, convergence of the OS algorithm is proved in [17]. Convergence results in the case where $d(u)$ is nonlinear and possibly strongly degenerate are obtained in [11]. Although this algorithm converges as various discretization parameters tend to zero, it is not difficult to construct an explicit example (see [16]) which shows that OS is too diffusive in regions containing self-sharpening fronts, at least when the time step is large ($\gg \varepsilon$). This potential shortcoming has motivated the corrected operator splitting (COS) algorithm, which differs from OS in the way that the diffusion step is carried out.

Diffusion revisited (residual flux terms): To obtain correct structure of fronts, a residual flux term has to be constructed. To this end, we introduce the envelope function

$$f_c(v\,; v_l, v_r) = \begin{cases} \text{the lower convex envelope of } f \text{ between } v_l \text{ and } v_r \text{ if } v_l < v_r, \\ \text{the upper concave envelope of } f \text{ between } v_r \text{ and } v_l \text{ if } v_l > v_r. \end{cases} \tag{4}$$

Let $v = v(x, \Delta t_n)$ be the true solution of (2) at time $t = \Delta t_n$, which is known to be piecewise smooth. Let $\{y_i\}$ denote the discontinuity points of v. For a fixed i, let $\tilde{v}_i$ and

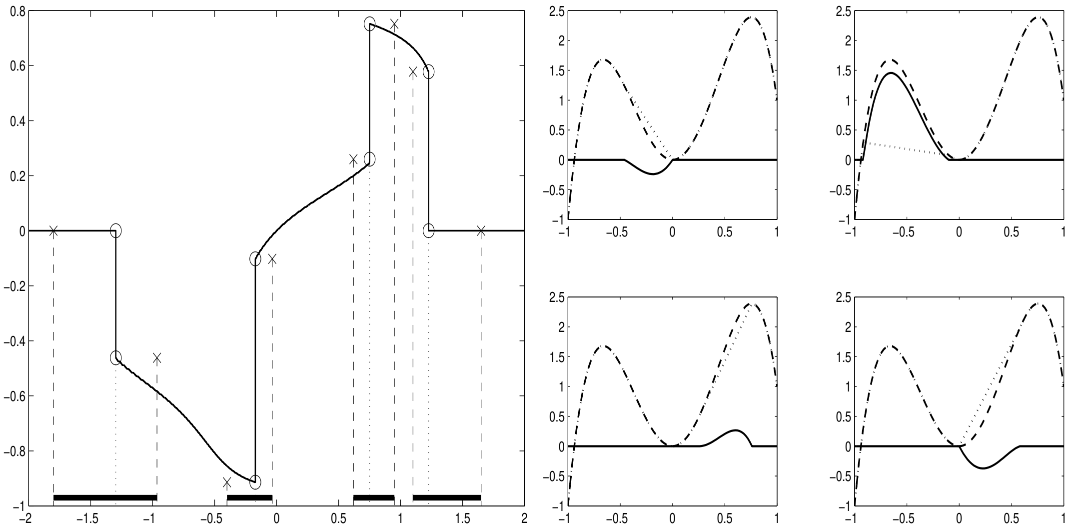

FIGURE 1. The left plot shows an exact solution (solid line) from the convection step, which contains four discontinuities (left and right shock states are shown by circles). The thick solid lines indicate where the corresponding residual flux terms are defined when used in the diffusion step. The right plots show the residual flux terms (solid line) associated with the four discontinuities (the upper left plot corresponds to the first discontinuity, the upper right plot to the second and so on). The flux function is shown as the thick dashed line and the envelope function as dotted.

$\tilde{v}_{i+1}$ denote the left and right limits of v at $x = y_i$, respectively. Choose positions x_i and x_{i+1} such that y_i is located somewhere in the interval (x_i, x_{i+1}). Then define the residual flux function associated with this (ith) discontinuity by (see Figure 1)

$$f^i_{\mathrm{res}}(v; t_n) = \begin{cases} f(v) - f_c(v; \tilde{v}_i, \tilde{v}_{i+1}), & \text{for } v \in [\tilde{v}_i, \tilde{v}_{i+1}], \\ 0, & \text{for } v \notin [\tilde{v}_i, \tilde{v}_{i+1}]. \end{cases}$$

For convenience, introduce the globally defined residual flux term $f_{\mathrm{res}}(x, v; t_n) = f^i_{\mathrm{res}}(v; t_n)$, $x \in [x_i, x_{i+1})$ for some i. Let now $w(x, \Delta t_n)$ be the weak solution of the following parabolic equation

$$\frac{\partial w}{\partial t} + \frac{\partial}{\partial x}\left(f_{\mathrm{res}}(x, w; t_n) - \varepsilon d(w) \frac{\partial w}{\partial x} \right) = 0, \qquad w(x, 0) = v(x, \Delta t_n),$$

with the Dirichlet boundary data imposed at $x = a, b$. Numerically this solution is found by a Petrov-Galerkin method with quadratic test functions, see e.g. [4, 6, 7, 10, 15] for details.

To find a residual flux term, we need to locate the associated discontinuities (shocks) based on the approximate solution generated by the Godunov method. This is done by introducing a suitable parameter $\rho > 0$ and identify a shock whenever the distance between two subsequent saturation values in the Godunov solution is greater than ρ, see [15] for details.

Remark. The loss of symmetry due to the residual flux term does not affect the overrall computational time for the COS method since the tridiagonal linear systems are solved by direct methods. This claim is confirmed in [15] where the numerical experiments show that COS' runtime (with one or a few time steps) is less than that of OS (with many time steps). In higher dimensions this loss of symmetry may become an issue since iterative methods have to be used to solve the resulting linear systems. However, since the equations are (partially) symmetrized by the use of a Petrov-Galerkin method this issue seems to be of much less consern than the restriction on the time step imposed for the OS method.

Remark. A suitable value of ρ is problem dependent and the parameter is adjusted by considering the number of grid blocks used in the numerical computations. Therefore, changing ρ could result in a less accurate construction of the residual flux function. However, since the number of grid blocks is the same in each numerical experiment, the parameter ρ is kept fixed.

2.2. **Numerical experiments.** For two phase flow in porous media [3] the advective flux function in (1) is given by

$$f(u) = \frac{\lambda_w(u)}{\lambda_w(u) + \lambda_o(u)}, \tag{5}$$

where $\lambda_i = \frac{k_{ri}}{\mu_i}$, is the mobility of phase i, $i = w, o$. Here, k_{ri}, denotes the relative permeability and μ_i the viscosity. The relative permeabilities are represented by third order normalised B-spline expansions [19] in which the coefficient vectors are given by $\vec{c}_w = (0.0, 0.1, 0.52, 1.0)$ and $\vec{c}_o = (1.0, 0.25, 0.02, 0.0)$. The knot vector for the B-spline basis is $\vec{y} = (0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0)$. Furthermore, we set $d(u) \equiv 1$, $\rho = 0.05$ and $\varepsilon = 0.01$.

The saturation profiles are simulated using a flux function, $f^*(u)$, which corresponds to the B-spline expansion given above (see Figure 2a)). The initial saturation (see Figure 2b)) is

$$u_0(x) = \begin{cases} 1.0 - x, & 0.0 \leq x \leq 0.3, \\ 0.7 - 70 \cdot (x - 0.3), & 0.3 \leq x \leq 0.31, \\ 0.0, & 0.31 < x \leq 1.0. \end{cases}$$

The Dirichlet boundary data are $u = 1$ at $x = 0$ and $u = 0$ at $x = 1$. In the numerical experiments the size of the grid blocks is given by $\Delta x = \frac{1}{100}$ and $\Delta x_c = \frac{1}{200}$. The grids are fixed since we here want to study the effect of varying the time step. Only one shock front is present in the solution, therefore we let the corresponding residual flux term be defined on the whole interval $[0, 1]$. In the COS simulation we have used 1 time step to reach $T = 0.2$ since with the COS method the correct width of the front is obtained independently of the time step. In the OS simulations the number of time steps is 1 and 10. We refer to these simulations as COS(1), OS(1) and OS(10), respectively. In Figure 3 the COS and OS solutions are shown together with the reference solution, which is generated by a finite difference scheme with very fine space and time discretization ($\Delta t = \frac{T}{4080}$ and $\Delta x = \frac{1}{5000}$). To measure the relative errors in the (C)OS saturation profiles we define

$$r_{\text{(C)OS}} = \frac{||\vec{u}_{\text{ref}} - \vec{u}_{\text{(C)OS}}||_2}{||\vec{u}_{\text{ref}}||_2}. \tag{6}$$

We have $r_{\text{COS(1)}} = 0.016$, $r_{\text{OS(1)}} = 0.051$ and $r_{\text{OS(10)}} = 0.0136$. Thus the relative errors measured in the L_2-norm are about equally large for the COS(1) and OS(10) solutions. From Figure 3 and the relative errors it is seen that COS is the most efficient and accurate algorithm for the problem under consideration. We refer to [15, 16] for a more detailed comparison of COS and OS for the forward problem.

One may ask whether the relative merits of OS and COS will be similar also when the models are applied to solve an associated inverse problem. The importance of neglecting some specific physics may be different for the inverse problem than for the forward problem.

3. The inverse problem

3.1. **Problem formulation.** We will attempt to recover the flux function, $f(u)$, from discrete values of u, which are denoted by $\vec{u}_{\text{obs}}$. The forward model is given by equation (1) (recall that $d(u) \equiv 1$) with an appropriate solution algorithm for $u(x, t)$. The OS(10)

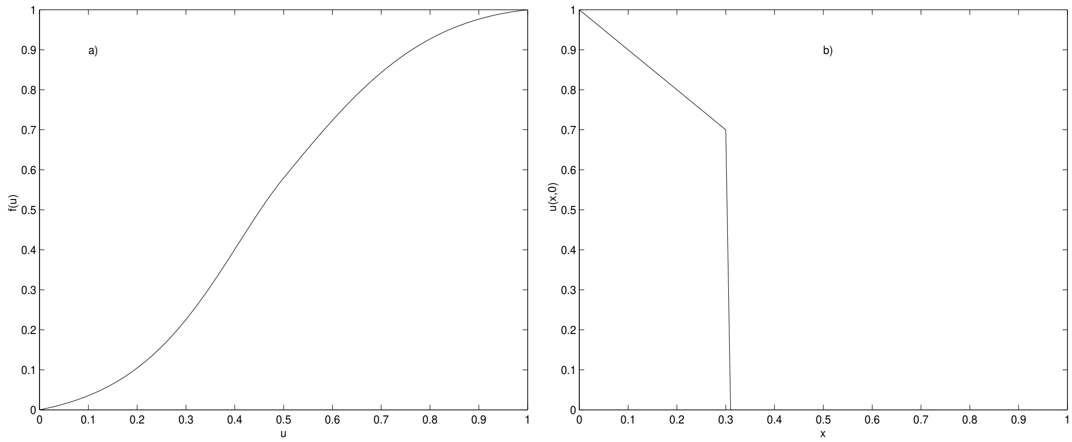

FIGURE 2. The flux function, $f^*(u)$, in plot a) and the initial saturation, $u_0(x)$, in plot b).

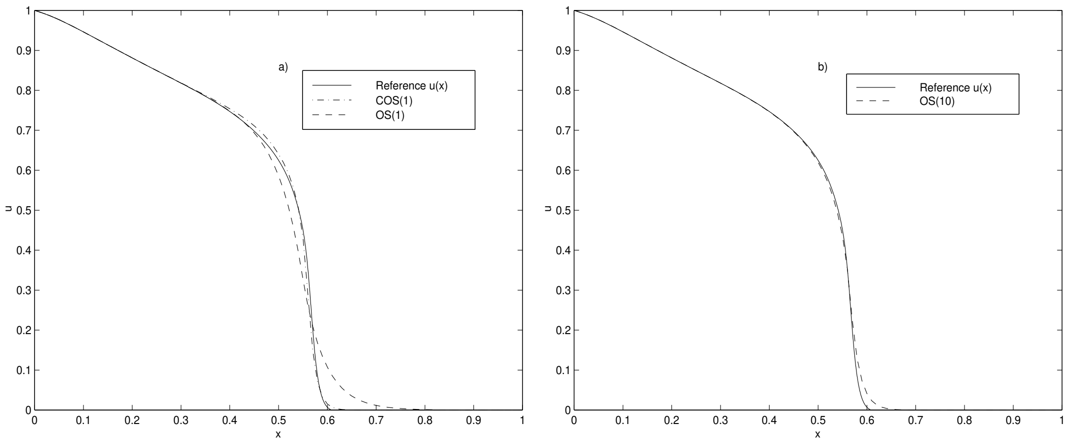

FIGURE 3. The reference, COS(1) and OS(1) solutions of the forward problem in plot a). The reference solution and OS(10) solution in plot b).

and COS(1) solution algorithms are selected since these gave similar accuracy when applied to the forward problem.

The discrete observed values will be sampled from the fine-grid solution of the forward problem at time $T = 0.2$. The flux function used when calculating the fine-grid solution is $f^*(u)$ (see section 2.2).

Objective functions, which measure the least squares distance between observed and corresponding calculated values of $\vec{u}$, are formed; $J_{(\mathrm{C})\mathrm{OS}}(f) = \|\vec{u}_{\mathrm{obs}} - \vec{u}_{(\mathrm{C})\mathrm{OS}}(f)\|_2^2$. Recovery of $f^*(u)$ is carried out through minimisation of the objective functions. Representing the flux function by a B-spline expansion leads to a finite dimensional parameter estimation problem for the coefficients in the expansion. This is solved by a Levenberg-Marquardt algorithm [18].

3.2. **Solution criteria.** Different parametrisations of the flux function lead to different parameter estimation problems with solutions corresponding to different functional spaces for the estimated flux function. Hence, having solved the parameter estimation problem does not necessarily mean that one has solved the underlying inverse problem. Furthermore, inverse problems are known to be inherently ill-posed. Solution non-existence, solution non-uniqueness and instability of the solution with respect to data may occur. This shows the need for solution criteria to decide whether a solution to the parameter estimation problem is also a solution to the original inverse problem.

Often, inverse problems where the modelling errors are negligible compared to the error in the observed data are considered. When the noise in the observed data is a result of additive independent random errors (e.g., resulting from independent measurements), it is possible to apply statistically based solution criteria [1, 20] to test if the solution to a parameter estimation problem can be accepted as a solution to the underlying inverse problem. These criteria concern the value of the objective function and the randomness of the elements in the residual vector at solution. Furthermore, if a solution to this kind of inverse problem has been found, it is possible to give approximate error bounds for the solution.

The inverse problem studied here is quite different. We consider a problem where the modelling error dominates the error in the observed data. In fact, the whole purpose is to study the effect of two different modelling errors on the recovery of the flux function. Contrary to measurement errors, modelling errors can not be expected to be additive independent random errors. Hence, neither the above solution criteria nor the error bounds can be applied. Presently, we do not know enough about the nature of the modelling error to put forth any alternative solution criteria covering this case. Therefore, the estimated flux functions presented later will be solutions to specific parameter estimation problems, while we can not claim that any inverse problem has actually been solved. However, it is still of interest to study the influence of the modelling errors in OS and COS when attempting to recover the flux function.

We will use the same parametrisations of the flux function when running the OS(10) model as when running the COS(1) model, thereby ensuring that corresponding OS and COS estimates belong to the same functional space.

3.3. **Numerical experiments.** The observed data are generated at $T = 0.2$ by a finite difference scheme using the flux function, $f^*(u)$, and the fine grid given in section 2.2. From the resulting saturation profile 101 equally spaced data are sampled. This constitutes the observation data used in the parameter estimation. The size of the grid cells used in OS(10) and COS(1) is fixed; $\Delta x = \frac{1}{100}$.

For a given order, m, of the B-spline basis functions and a given number of internal knots, k, in the knot vector, the number of coefficients in each relative permeability curve is $m + k$. The number of elements in the knot vector is $2m + k$. Here the order, m, is fixed; $m = 3$. The first and last coefficients in the B-spline expansions of each relative permeability curve are also fixed, $c_w^1 = c_o^{m+k} = 0.0$ and $c_w^{m+k} = c_o^1 = 1.0$, so that the total number of parameters to be estimated is $2(m + k) - 4 = 2(k + 1)$.

In the numerical experiments 1, 2, 3 and 4, the flexibility of the estimated fractional flow function is systematically increased by adding knots in the knot vector (increasing k). The number of knots is $k = 1$, $k = 2$, $k = 3$ and $k = 7$, respectively. Increased flexibility in the flux function allows the flux function to compensate for modelling errors in the OS and COS models. Hence, we expect the minimum value of the objective function, $J_{\mathrm{(C)OS}}(f)$, to decrease with increased flexibility in the flux function while the relative errors in the estimated flux functions may increase. The COS and OS estimated flux functions and the corresponding minimum values of $J_{\mathrm{(C)OS}}(f)$ will be compared at each level of flexibility.

The knot vector, initial guess on the unknown parameters, and the estimated parameters from the experiments can be found in Table 1, 2, 3 and 4, respectively. The minimum values of the objective function, $J_{\mathrm{(C)OS}}(f)$, and the relative errors in the estimated flux functions are given in Table 5.

In Figure 4b) the observed saturations and (C)OS simulated saturations corresponding to the minimum value of $J_{\mathrm{(C)OS}}$ are given for experiment 1. We omit plotting the saturation profiles for experiment 2, 3 and 4, since they are visually identical to the plot in Figure 4b). In Figure 5 and 6 the true flux function, $f^*(u)$, and the estimated flux functions, $f_{\mathrm{(C)OS}}(u)$, are plotted.

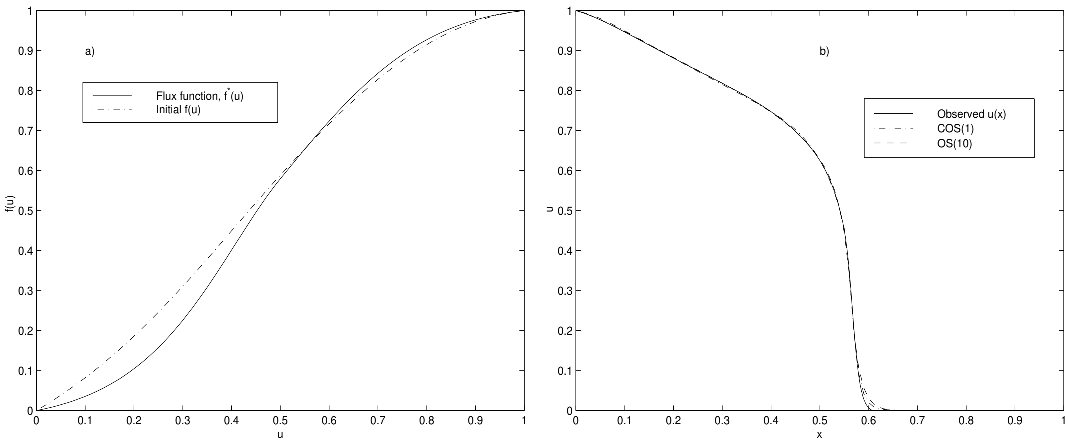

FIGURE 4. The true flux function, $f^*(u)$, and the initial guess on the flux function in plot a). In plot b) the observed saturation, u_{obs}, and simulated saturations, $u_{(\mathrm{C})\mathrm{OS}}$, in experiment 1.

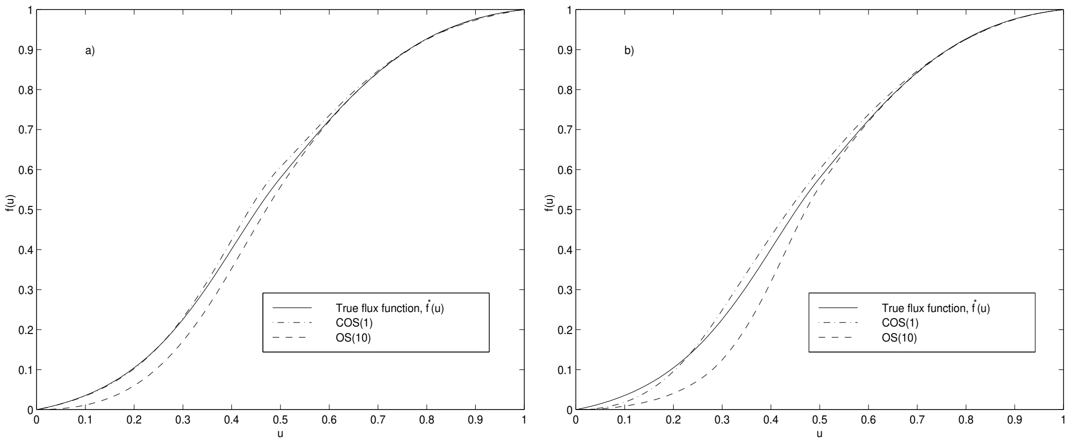

FIGURE 5. The true flux function, $f^*(u)$, and estimated flux functions, $f_{(\mathrm{C})\mathrm{OS}}(u)$, in experiment 1 (plot a)) and experiment 2 (plot b)).

The minimum values of the objective function, $J_{(\mathrm{C})\mathrm{OS}}(f)$, are decreasing with increased flexibility of the fractional flow function, whereas in most cases the relative error of the flux function is increasing. We also have $J_{\mathrm{COS}(1)} < J_{\mathrm{OS}(10)}$ and that the relative error in the flux function is greater with OS(10) than COS(1) at each level of flexibility.

We interpret these results as follows: In the convection step the speed of a point $\hat{u}$ on a saturation curve is given by $f'(\hat{u})$. The OS(10) solution contains numerical diffusion

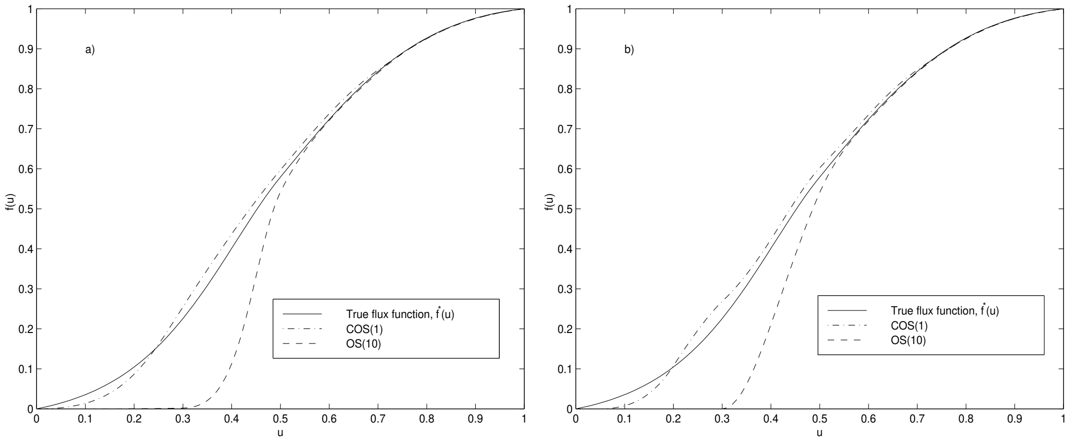

FIGURE 6. The true flux function, $f^*(u)$, and estimated flux functions, $f_{(\mathrm{C})\mathrm{OS}}(u)$, in experiment 3 (plot a)) and experiment 4 (plot b)).

TABLE 1

Experiment no. 1	
Knot vector	$\vec{y} = (0.0, 0.0, 0.0, 0.5, 1.0, 1.0, 1.0)$
Initial parameters	$\vec{c}_w = (0.0, 0.3, 0.82, 1.0)$ $\vec{c}_o = (1.0, 0.45, 0.02, 0.0)$
Estimated parameters COS(1)	$\vec{c}_w = (0.0, 0.098, 0.399, 1.0)$ $\vec{c}_o = (1.0, 0.166, 0.027, 0.0)$
Estimated parameters OS(10)	$\vec{c}_w = (0.0, 0.0, 0.725, 1.0)$ $\vec{c}_o = (1.0, 0.326, 0.019, 0.0)$

such that low-saturation points on the front travel too fast. To compensate for this the inversion algorithm seeks to decrease the speed of these points by deforming the estimated flux function in the lower saturation region, cf. Figure 5 and 6. The ability to do so, without affecting the higher saturation regions too much, increases with increased flexibility in the functional representation of the flux function. Observe that not even the deformed flux function in Figure 6b) did bring the minimum value of $J_{\mathrm{OS}(10)}$ below that of $J_{\mathrm{COS}(1)}$ in experiment 1. Thus, the modelling errors in OS(10) are clearly more serious than those of COS(1), as far as the current inverse problem is concerned.

In an attempt to remove numerical diffusion in the OS simulations, the number of time steps is increased to 20. With 20 time steps the width of the front in the OS solution is of

TABLE 2

Experiment no. 2	
Knot vector	$\vec{y} = (0.0, 0.0, 0.0, 0.3, 0.5, 1.0, 1.0, 1.0)$
Initial parameters	$\vec{c}_w = (0.0, 0.18, 0.456, 0.82, 1.0)$ $\vec{c}_o = (1.0, 0.67, 0.321, 0.02, 0.0)$
Estimated parameters COS(1)	$\vec{c}_w = (0.0, 0.0, 0.350, 0.79, 1.0)$ $\vec{c}_o = (1.0, 0.562, 0.258, 0.0192, 0.0)$
Estimated parameters OS(10)	$\vec{c}_w = (0.0, 0.0, 0.209, 0.671, 1.0)$ $\vec{c}_o = (1.0, 1.0, 0.220, 0.02, 0.0)$

TABLE 3

Experiment no. 3	
Knot vector	$\vec{y} = (0.0, 0.0, 0.0, 0.3, 0.5, 0.7, 1.0, 1.0, 1.0)$
Initial parameters	$\vec{c}_w = (0.0, 0.18, 0.456, 0.664, 0.892, 1.0)$ $\vec{c}_o = (1.0, 0.67, 0.321, 0.149, 0.012, 0.0)$
Estimated parameters COS(1)	$\vec{c}_w = (0.0, 0.0, 0.215, 0.315, 0.520, 1.0)$ $\vec{c}_o = (1.0, 0.338, 0.152, 0.063, 0.014, 0.0)$
Estimated parameters OS(10)	$\vec{c}_w = (0.0, 0.0, 0.002, 0.255, 0.587, 1.0)$ $\vec{c}_o = (1.0, 1.0, 0.066, 0.064, 0.014, 0.0)$

TABLE 4

Experiment no. 4	
Knot vector	$\vec{y} = (0.0, 0.0, 0.0, 0.05, 0.1, 0.2, 0.3, 0.5, 0.6, 0.7, 1.0, 1.0, 1.0)$
Initial parameters	$\vec{c}_w = (0.0, 0.03, 0.089, 0.177, 0.290, 0.456, 0.612, 0.710, 0.892, 1.0)$ $\vec{c}_o = (1.0, 0.945, 0.842, 0.697, 0.530, 0.321, 0.192, 0.122, 0.012, 0.0)$
Estimated parameters COS(1)	$\vec{c}_w = (0.0, 0.0, 0.0, 0.034, 0.189, 0.189, 0.269, 0.319, 0.477, 1.0)$ $\vec{c}_o = (1.0, 1.0, 1.0, 0.711, 0.400, 0.128, 0.08, 0.047, 0.0139, 0.0)$
Estimated parameters OS(10)	$\vec{c}_w = (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.296, 0.352, 0.587, 1.0)$ $\vec{c}_o = (1.0, 1.0, 1.0, 1.0, 0.114, 0.114, 0.094, 0.058, 0.0134, 0.0)$

the order of the physical front, ε. We repeat the numerical experiments 1, 2, 3 and 4 using OS(20). The minimum values of the objective function, $J_{\mathrm{OS}}(f)$, and the relative errors of

Table 5

Experimental results				
Objective functions, $J_{(\mathrm{C})\mathrm{OS}}(f)$				
Experiment no.	1	2	3	4
COS(1)	0.00096	0.00034	0.00027	0.000231
OS(10)	0.00251	0.00245	0.00223	0.00219
OS(20)	0.0012	0.00090	0.00082	0.0008
Relative error in $f_{(\mathrm{C})\mathrm{OS}}(u)$				
Experiment no.	1	2	3	4
COS(1)	0.01889	0.0252	0.0286	0.03094
OS(10)	0.0449	0.0728	0.1853	0.15644
OS(20)	0.0365	0.030	0.0349	0.0545

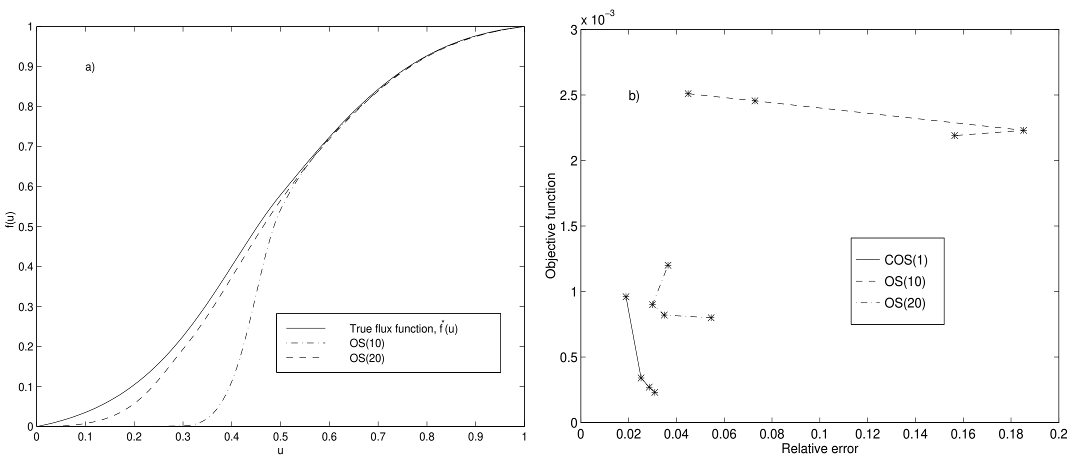

Figure 7. The true flux function, $f^*(u)$, and OS(10) and OS(20) estimated flux functions in plot a). The minimum values of the objective function, $J_{(\mathrm{C})\mathrm{OS}}(f)$, versus the relative errors in the estimated flux function, $f_{(\mathrm{C})\mathrm{OS}}(u)$, in plot b).

the estimated flux functions have decreased compared with the OS(10) results (see Table 5 and Figure 7b)). Figure 7a) shows the true flux function, $f^*(u)$, and the OS(10) and OS(20) estimated flux functions in experiment 3. The OS(20) estimated flux function is considerably better than the one achieved with OS(10). This supports our view that the deformation in the OS(10) estimated flux functions is due to numerical diffusion.

4. Conclusion

We have compared COS and OS models with respect to accuracy in the context of solving the one-dimensional saturation equation of porous media flow. In line with previous work, the numerical experiments indicate that the COS method is much more accurate than the OS method. For the problem considered here the OS approach needed 10 time steps to achieve the same accuracy in the L_2-norm as COS did with 1 time step.

We have also studied the COS and OS models in the context of an inverse problem where we recover the flux function in the one-dimensional saturation equation from discrete saturation data. The numerical experiments indicate that even though the COS(1) and OS(10) models solve the forward problem with similar accuracy, and thus in that sense contain modelling errors of equal size, the modelling error in OS(10) is the most serious one for this inverse problem.

References

[1] Y. Bard. *Nonlinear Parameter Estimation.* John Wiley and Sons, New York City, 1981.

[2] J. T. Beale and A. Majda. Rates of convergence for viscous splitting of the Navier-Stokes equations. *Math. Comp.*, 37(156):243–259, 1981.

[3] G. Chavent and J. Jaffre. *Mathematical models and finite elements for reservoir simulation*, volume 17 of *Studies in mathematics and it's applications.* North Holland, Amsterdam, 1986.

[4] H. K. Dahle. *Adaptive characteristic operator splitting techniques for convection-dominated diffusion problems in one and two space dimensions.* PhD thesis, Department of Mathematics, University of Bergen, 1988.

[5] H. K. Dahle. ELLAM-based operator splitting for nonlinear advection diffusion equations. Technical Report 98, Department of Mathematics, University of Bergen, 1995.

[6] H. K. Dahle, M. S. Espedal, R. E. Ewing, and O. Sævareid. Characteristic adaptive subdomain methods for reservoir flow problems. *Numerical Methods for PDEs*, 6:279–309, 1990.

[7] H. K. Dahle, M. S. Espedal, and O. Sævareid. Characteristic, local grid refinement techniques for reservoir flow problems. *International J. for Numerical Methods in Engineering*, 34:1051–1069, 1992.

[8] H. K. Dahle, R. E. Ewing, and T. F. Russell. Eulerian-Lagrangian localized adjoint methods for a nonlinear advection-diffusion equation. *Comput. Methods. Appl. Mech. Engrg.*, 122:223–250, 1995.

[9] C. N. Dawson. Godunov-mixed methods for advective flow problems in one space dimension. *SIAM J. Num. Anal.*, 28(5):1282–1309, Oct. 1991.

[10] M. S. Espedal and R. E. Ewing. Characteristic Petrov-Galerkin subdomain methods for two-phase immiscible flow. *Comput. Methods. Appl. Mech. Engrg.*, 64:113–135, 1987.

[11] S. Evje and K. H. Karlsen. Viscous splitting approximation of mixed hyperbolic-parabolic convection-diffusion equations. Report No. 2, 1997/98. Institut Mittag-Leffler. Available at the URL `http://www.math.ntnu.no/conservation/`.

[12] R. E. Ewing. Operator splitting and Eulerian-Lagrangian localized adjoint methods for multiphase flow. In Whiteman, editor, *The Mathematics of Finite Elements and Applications VII MAFELAP*, pages 215–232. Academic press, San Diego, CA, 1991.

[13] J. B. Goodman and R. J. LeVeque. A geometric approach to high resolution tvd schemes. *SIAM J. Num. Anal.*, 25:268–284, 1988.

[14] H. Holden, L. Holden, and R. Høegh-Krohn. A numerical method for first order nonlinear scalar conservation laws in one-dimension. *Comput. Math. Applic.*, 15(6–8):595–602, 1988.

[15] K. H. Karlsen, K. Brusdal, H. K. Dahle, S. Evje, and K.-A. Lie. The corrected operator splitting approach applied to a nonlinear advection-diffusion problem. To appear in Comput. Methods. Appl. Mech. Engrg.

[16] K. H. Karlsen and N. H. Risebro. Corrected operator splitting for nonlinear parabolic equations. Preprint, University of Bergen, Norway, 1997. Available at the URL `http://www.math.ntnu.no/conservation/`.

[17] K. H. Karlsen and N. H. Risebro. An operator splitting method for nonlinear convection-diffusion equations. *Numer. Math.*, 77(3):365–382, 1997.

[18] D. . W. Marquardt. An algorithm for least squares estimation of nonlinear parameters. *SIAM J. Appl. Math.*, 11:431–441, 1963.

[19] L. L. Schumaker. *Spline Functions: Basic Theory.* John Wiley and Sons, New York, 1981.

[20] S. Weisberg. *Applied Linear Regression.* John Wiley and Sons, New York, 1985.

(Kari Brusdal) Department of Mathematics, University of Bergen, Johs. Brunsgt. 12, N-5008 Bergen, Norway

E-mail address: `kari.brusdal@mi.uib.no`

(Helge K. Dahle) Department of Mathematics, University of Bergen, Johs. Brunsgt. 12, N-5008 Bergen, Norway

E-mail address: `helge.dale@mi.uib.no`

(Kenneth Hvistendahl Karlsen) Department of Mathematics, University of Bergen, Johs. Brunsgt. 12, N-5008 Bergen, Norway

E-mail address: `kenneth.karlsen@mi.uib.no`

(Trond Mannseth) RF Rogaland Research,, Thormøhlensgt. 55, N-5008 Bergen, Norway

E-mail address: `trond.mannseth@rf.no`