ANALYSIS OF UPWIND AND HIGH-RESOLUTION SCHEMES FOR SOLVING CONVECTION DOMINATED PROBLEMS IN POROUS MEDIA

V. STARIKOVIČIUS, R. ČIEGIS and A. JAKUŠEV

Vilnius Gediminas Technical University

Saulėtekio 11, LT-10223 Vilnius, Lithuania

E-mail: `vs@sc.vtu.lt`; `{rc, alexj}@fm.vtu.lt`

Abstract. The conservation laws governing the multiphase flows in porous media are often convection-dominated and have a steep fronts that require accurate resolution. Standard discretization methods of the convection terms do not perform well for such problems. The main aim of this work is to analyze the use of upwind and high- resolution schemes in such cases. First, we use a first differential approximation method to perform a theoretical analysis of a standard upwind approximation and different time stepping schemes for the linear hyperbolic equations in 1- and 2D. Next, we present a popular approach to reduce the amount of numerical diffusion introduced by upwind approximation - high-resolution schemes. We compare our implementation of one of the recently proposed central-upwind schemes against the upwind schemes on several test problems based on Buckley-Leverett equation and discuss the results. Finally, a parallel version of central-upwind scheme in 2D is presented. It was implemented using our C++ library of parallel arrays - *ParSol*.

Key words: hyperbolic systems of conservation laws, upwind and high resolution schemes, multiphase flows in porous media

1. Introduction

The multiphase flow in porous media has gained recently a lot of attention. This is due to the fact that problems involving the multiphase flow, heat transfer, and multicomponent mass transport in porous media arise in a broad spectrum of engineering disciplines. Important technological applications include the drying of porous solids and soils, subsurface contamination and remediation, thermally enhanced oil recovery, geothermal energy production, porous heat pipes, nuclear reactor safety analysis, high-level radioactive waste repositories, paper machines.

A standard modeling approach for multiphase flow in porous media are macroscopic models obtained by volume averaging or homogenization methods from microscopic equations. The transport phenomena are mathematically described by the basic principles of conservation for each phase separately and by appropriate interfacial conditions between various phases. Unfortunately, the resulting models are difficult to solve due to the large number of a strongly coupled nonlinear differential equations in the systems. Moreover, the transport equations are often degenerating and changing their type from parabolic to hyperbolic.

The mathematical models for multiphase flow and heat transfer in porous media were described in many papers and books (see [4, 5, 7] and references cited in these works). Our group is working on a software tool for solution of multiphase flow problems - *MfsolverC++* [5, 6]. We are using an object-oriented approach and robust numerical methods to create an extensible and flexible toolbox. Some of the adopted strategies are used in Diffpack [12] and OpenFOAM [1] packages.

Many problems describing multiphase flow in porous media are convection (advection) dominated parabolic problems. If diffusive fluxes (capillary forces) are very small compared to the convective fluxes, the transport equations become hyperbolic. Sharp fronts of the solutions appear often in various applications. They should be resolved accurately by the discrete schemes. We are using a finite volume method for the discretization of model equations in our tool. This method is very popular in CFD, because it ensures a local mass conservation. This is very important for the multiphase models in order to get correct shock positions and to be able to follow small saturations and concentrations. However, as in finite difference or finite element methods, the discretization of the convective term is still quite an open issue. Standard methods of the discretization do not perform well for convection-dominated or purely hyperbolic problems.

It is well know that central differences are second order accurate, but produce nonphysical oscillations in the solutions. The monotonicity property (non-oscillating solutions) is crucial for the desired approximations, since the governing nonlinearities in multiphase models are only defined for saturation values between zero and one. Otherwise, the upwind approximations are monotone, but only first order accurate. They introduce a numerical diffusion, which is reducing the accuracy of the scheme, particularly in the regions of high gradients, and is smearing the sharp fronts of the solution profile. In multidimensional cases, the upwind approximations show a grid orientation sensitivity. In cases of high streamline-to-grid skewness, the degradation of accuracy becomes almost unacceptable, since it requires very fine multidimensional grid resolution.

It is clear that a good approximation scheme should find a balance between monotonicity (boundedness) and accuracy. During the 50 years history of developments, different approaches and concepts have been proposed and used. For historical review on different methods such as Riemann-solver-based upwind schemes, switching schemes based on Peclet number, blended schemes, TVD, ENO, WENO schemes we refer to [16, 13] and references therein. The

most promising approach at this stage combines a higher-order scheme with upwind differencing on a face-by-face basis, based on different boundedness criteria (limiters). So called "High-resolution schemes" are at least second order accurate in regions where the solution is smooth, while capturing fronts without numerical oscillations in regions of strongly varying solution gradients.

Developing *MfsolverC++* toolbox, we want to study the performance of recently emerging high-resolution schemes for convection-dominated multiphase flow problems. Our goal is to test, whether the these new approaches are efficient and robust enough to be implemented in our tool. In this study we limit our selves to purely hyperbolic test cases. The important specific feature of hyperbolic conservation laws is the existence of discontinuous weak solutions and the fact that they are often not unique. Additional conditions are required to pick up the correct physical solution from all possible weak solutions of the problem. Obviously, this requires from the numerical schemes not only the accurate representation of the shocks, but also the convergence to the correct physical solution.

In this study our main benchmark problem is the Buckley-Leverett equation. It describes the displacement of one incompressible fluid by the other (e.g. oil by the water) in a horizontal reservoir when externally applied driving forces are large in relation to the gradient of capillary pressure. In 1D case, the pressure equation of a two-phase model gives a constant total Darcy velocity $v = v_1 + v_2$. Neglecting the diffusive term of capillary effects, the transport equation is reduced to

$$\frac{\partial S_1}{\partial t} + \frac{\partial}{\partial x} f(S_1) = 0, \tag{1.1}$$

$$f(S_1) = \frac{v}{\varepsilon} \lambda_1(S_1) = \frac{v}{\varepsilon} \frac{k_{r1}(S_1)}{k_{r1}(S_1) + \frac{\mu_1}{\mu_2} k_{r2}(1 - S_1)},$$

where ε is the porosity of the porous medium, S_i is the saturation of phase (fluid) i, obviously $S_1 + S_2 = 1$, $\lambda_i = \lambda_i(S_i)$ is the fractional flow function of phase i, $\lambda_1 + \lambda_2 = 1$, $k_{ri} = k_{ri}(S_i)$ is the relative permeability function of phase i, μ_i is the dynamic viscosity of phase i. The Buckley-Leverett equation (1.1) is a very popular validation test for the multiphase flow solvers and schemes, because it can be solved analytically [4, 7, 13].

The rest of this paper is organized as follows. Section 2 introduces the upwind approximation and different time discretization approaches in a case of linear hyperbolic equation. We use *a first differential approximation* (or *modified equations*) to perform a theoretical analysis of different schemes in 1- and 2D. It will help us to understand and explain the behaviour of upwind approximations which we will observe later in the numerical experiments. Section 3 presents a recently proposed way of constructing of high resolution schemes – so called central upwind schemes. In Section 4 we compare our implementation of such scheme against the upwind schemes on several test problems and discuss the results. Section 5 presents a parallel version of 2D

central-upwind scheme implemented using our C++ library of parallel arrays *ParSol*. Finally, we conclude in Section 6.

2. Linear Hyperbolic Equation

2.1. 1D case

In this section we consider an initial-boundary value problem for one-dimensional linear transport equation

$$
\begin{cases}
\dfrac{\partial u}{\partial t} + v\dfrac{\partial u}{\partial x} = 0, \;\; v > 0, \\[2mm]
u(x,0) = u_0(x), \;\; 0 \le x \le 1, \\[2mm]
u(0,t) = \mu(t), \;\; 0 < t \le T,
\end{cases}
\tag{2.1}
$$

where v is a constant. The exact solution of this problem is well known and it describes a wave advanced along the characteristic

$$
u(x,t) = \varphi(x - vt).
$$

Problem (2.1) is used as a simple benchmark to introduce the discrete approximations and to test their accuracy by investigating the *modified equations* of difference schemes.

In $Q_T = [0,1] \times [0,T]$ we define a uniform grid $Q_{h\tau} = \omega_h \times \omega_\tau$:

$$
\omega_h = \big\{ x_j : \;\; x_j = jh, \;\; 0 < j < J, \;\; x_J = 1 \big\},
$$
$$
\omega_\tau = \big\{ t^n : \;\; t^n = n\tau, \;\; n = 1, 2, \ldots, N, \;\; N\tau = T \big\}.
$$

Let $U_j^n = U(x_j, t^n)$ be a discrete approximation to the exact solution of differential problem (2.1).

By using the FV method for approximation of differential equation and applying simple approximations of the obtained integrals we get the following discrete scheme

$$
\begin{cases}
\dfrac{U_j^{n+1} - U_j^n}{\tau} + \sigma v \partial_{\bar{x}} U_j^{n+1} + (1-\sigma) v \partial_{\bar{x}} U_j^n = 0, \;\; x \in \omega_h, \\[2mm]
U_j^0 = u_0(x_j), \;\; 0 \le j \le J, \\[2mm]
U_0^{n+1} = \mu(t^{n+1}).
\end{cases}
\tag{2.2}
$$

Here $\partial_{\bar{x}} U$ defines a first order upwind approximation of the convection term

$$
\partial_{\bar{x}} U_j = \frac{U_j - U_{j-1}}{h},
$$

a value $\sigma = 0$ corresponds with the first-order explicit Euler method, $\sigma = 1$ corresponds with the fully implicit Euler method, and the value $\sigma = \frac{1}{2}$ corresponds with the second-order Crank-Nicolson method.

2.1.1. Accuracy of the explicit Euler scheme

Let us define the residual of the discrete scheme

$$R_{EES}(u) := \frac{u_j^{n+1} - u_j^n}{\tau} + v\partial_{\bar{x}} u_j^n,$$

where $u(x,t)$ is a solution of problem (2.1). By using Taylor series we get the following estimate of the residual:

$$R_{EES}(u) = \frac{\tau}{2}\frac{\partial^2}{\partial t^2}u(x_j, t^n + \theta_1\tau) - \frac{vh}{2}\frac{\partial^2}{\partial x^2}u(x_j - \theta_2 h, t^n), \ \ 0 < \theta_1, \theta_2 < 1.$$

Thus the approximation of the explicit Euler scheme is first order in time and space.

By differentiating differential equation (2.1) with respect to time we obtain the equality

$$\frac{\partial^2 u}{\partial t^2} = v^2\frac{\partial^2 u}{\partial x^2},$$

thus both terms in formula of the residual introduce a numerical diffusion. Assuming some additional smoothness of the solution, the residual can be written as

$$R_{EES}(u) = \frac{v}{2}\big(v\tau - h\big)\frac{\partial^2}{\partial x^2}u(x_j, t^n) + \mathcal{O}(\tau^2 + h^2).$$

It is well known that one and the same finite difference scheme can approximate many differential equations. First we prove that the explicit Euler scheme approximates the following equation

$$\frac{\partial w}{\partial t} + v\frac{\partial w}{\partial x} = \frac{v}{2}\big(h - v\tau\big)\frac{\partial^2 w}{\partial x^2} \tag{2.3}$$

with the second order in time and space. The proof of this statement follows after simple computations:

$$R_{EES}(w) = \frac{\partial w}{\partial t} + v\frac{\partial w}{\partial x} + \frac{\tau}{2}\frac{\partial^2 w}{\partial t^2} - \frac{vh}{2}\frac{\partial^2 w}{\partial x^2} + \mathcal{O}(\tau^2 + h^2)$$

$$= \frac{\tau v}{4}\big(h - v\tau\big)\Big(\frac{\partial^3 w}{\partial t\partial x^2} - v\frac{\partial^3 w}{\partial x^3}\Big) + \mathcal{O}(\tau^2 + h^2) = \mathcal{O}(\tau^2 + \tau h + h^2),$$

here $w = w_j^n$. The obtained differential equation is called the *modified equation* (or *the first differential approximation*) of the explicit Euler scheme [13].

Two conclusions follow from equation (2.3).

1. Equation (2.3) is well-defined parabolic equation if time and space steps satisfy the inequality $v\tau \leq h$.

   It becomes ill–posed parabolic equation with inverse direction of time if $v\tau > h$. Thus we obtain the well known stability requirement of the explicit Euler scheme.

2. Numerical diffusions due to time and space discretizations are of opposite signs, thus they compensate each other, e.g. the scheme is exact if $v\tau = h$.

Remark 1. It is well-known that in order to define a generalized (weak) solution of hyperbolic equations for nonsmooth data, the PDE can be modified by adding a small diffusive term [13]. The explicit Euler scheme introduces this artificial diffusion by the numerical diffusion and in this way it defines the "vanishing viscosity" solution, which is a correct physical solution.

We can explicitly calculate the solution of (2.3). For simplicity of presentation let assume that the influence of boundary conditions is neglected, i.e. the problem is solved in an infinite space strip, then we get

$$w(x,t) = \frac{1}{2\sqrt{\pi D t}} \int_{-\infty}^{\infty} u_0(z) e^{-\frac{(z-x+vt)^2}{4Dt}} \, dz, \quad D = \frac{v}{2}\big(h - v\tau\big).$$

2.1.2. Accuracy of the fully implicit Euler scheme

Let us define the residual of the fully implicit Euler scheme

$$R_{IES}(u) := \frac{u_j^{n+1} - u_j^n}{\tau} + v\partial_{\bar{x}} u_j^{n+1}.$$

It can be estimated as

$$R_{IES}(u) = -\frac{\tau}{2}\frac{\partial^2}{\partial t^2} u(x_j, t^{n+1} - \theta_3\tau) - \frac{vh}{2}\frac{\partial^2}{\partial x^2} u(x_j - \theta_2 h, t^{n+1}), \quad 0 < \theta_3 < 1.$$

Thus the approximation of the fully implicit Euler scheme is also first order in time and space.

It is easy to prove that the modified equation for the fully implicit Euler scheme is given by

$$\frac{\partial w}{\partial t} + v\frac{\partial w}{\partial x} = \frac{v}{2}\big(h + v\tau\big)\frac{\partial^2 w}{\partial x^2}. \tag{2.4}$$

Two conclusions can be done from equation (2.4).

1. Equation (2.4) is unconditionally parabolic, thus we again obtain the well known fact that the fully implicit Euler scheme is unconditionally stable.
2. Numerical diffusions due to time and space discretizations are of the same sign, thus they are added and do not compensate each other. Therefore the explicit Euler scheme is more accurate than the implicit scheme for $v\tau < h$, especially for $v\tau \sim h$.

It follows from (2.4) that the discretization modifies the PDE by adding some numerical diffusion, which defines correctly the vanishing viscosity solution.

2.1.3. Accuracy of the Crank-Nicolson scheme

Let us define the residual of the Crank-Nicolson scheme

$$R_{CN}(u) := \frac{u_j^{n+1} - u_j^n}{\tau} + v\partial_{\bar{x}}\left(\frac{u_j^{n+1} + u_j^n}{2}\right).$$

It can be estimated as

$$R_{CN}(u) = \frac{\tau^2}{24}\frac{\partial^3}{\partial t^3}u(x_j, t^{n+1/2}) + v\frac{\tau^2}{8}\frac{\partial^3}{\partial t^2\partial x}u(x_j, t^{n+1/2})$$

$$- \frac{vh}{2}\frac{\partial^2}{\partial x^2}u(x_j, t^{n+1/2}) + \mathcal{O}(\tau^3 + h^2).$$

The approximation of the Crank-Nicolson scheme is second order in time and first order in space. By using the following consequences from the differential equation

$$\frac{\partial^3 u}{\partial t^3} = -v^3\frac{\partial^3 u}{\partial x^3}, \qquad \frac{\partial^3 u}{\partial t^2\partial x} = v^2\frac{\partial^3 u}{\partial x^3},$$

we write the residual as:

$$R_{CN}(u) = \frac{v^3\tau^2}{12}\frac{\partial^3}{\partial x^3}u(x_j, t^{n+1/2}) - \frac{vh}{2}\frac{\partial^2}{\partial x^2}u(x_j, t^{n+1/2}) + \mathcal{O}(\tau^3 + h^2).$$

The modified equation of the Crank-Nicolson scheme is given by

$$\frac{\partial w}{\partial t} + v\frac{\partial w}{\partial x} = \frac{vh}{2}\frac{\partial^2 w}{\partial x^2} - \frac{v^3\tau^2}{12}\frac{\partial^3 w}{\partial x^3}. \tag{2.5}$$

2.2. 2D case

Let us consider an initial-boundary value problem for two-dimensional linear hyperbolic equation

$$\begin{cases} \dfrac{\partial u}{\partial t} + v_1\dfrac{\partial u}{\partial x_1} + v_2\dfrac{\partial u}{\partial x_2} = 0, \quad v_1, v_2 > 0, \\[2mm] u(x_1, x_2, 0) = u_0(x_1, x_2), \quad 0 \le x_1, x_2 \le 1, \\[2mm] u(0, x_2, t) = \mu_1(x_2, t), \quad 0 < x_2 < 1, \, 0 < t \le T, \\[2mm] u(x_1, 0, t) = \mu_2(x_1, t), \quad 0 < x_1 < 1, \, 0 < t \le T, \end{cases} \tag{2.6}$$

where v_j are constants.

In $Q_T = [0,1] \times [0,1] \times [0,T]$ we define a uniform grid $Q_{h\tau} = \omega_h \times \omega_\tau$:

$$\omega_h = \big\{(x_{1j}, x_{2k}): \ x_{ij} = jh_i, \ \ 0 < j < J_i, \ x_{J_i} = 1, \ i = 1, 2\big\},$$

$$\omega_\tau = \big\{t^n: \ t^n = n\tau, \ n = 1, 2, \ldots, N, \ N\tau = T\big\}.$$

Let $U_{jk}^n = U(X_{jk}, t^n)$, $X_{jk} = (x_{1j}, x_{2k})$ be a discrete approximation to the exact solution of differential problem (2.6).

By using the FV method for approximation of differential equation and applying simple approximations of the obtained integrals we get the following discrete scheme

$$\frac{U_{jk}^{n+1} - U_{jk}^n}{\tau} + \sigma A U_{jk}^{n+1} + (1-\sigma)AU_{jk}^n = 0, \tag{2.7}$$

where we denoted by AU the two-point upwind approximation of convection terms

$$AU_{jk} = v_1 \partial_{\bar{x}_1} U_{jk} + v_2 \partial_{\bar{x}_2} U_{jk}.$$

2.2.1. Accuracy of the explicit Euler scheme

The residual of the discrete scheme is given by

$$R_{EES}(u) := \frac{u_{jk}^{n+1} - u_{jk}^n}{\tau} + Au_{jk}^n,$$

where $u(x_1, x_2, t)$ is a solution of (2.6). By using Taylor series we get the following estimate of the residual:

$$R_{EES}(u) = \frac{\tau}{2}\frac{\partial^2}{\partial t^2}u(X_{jk}, t^n + \theta_0\tau) - \frac{v_1 h_1}{2}\frac{\partial^2}{\partial x_1^2}u(x_{1j} - \theta_1 h_1, x_{2k}, t^n)$$

$$- \frac{v_2 h_2}{2}\frac{\partial^2}{\partial x_2^2}u(x_{1j}, x_{2k} - \theta_2 h_2, t^n), \quad 0 < \theta_i < 1, \ i = 0, 1, 2.$$

Thus the approximation of the explicit Euler scheme is first order in time and space. Applying the maximum principle it is easy to prove that this scheme is stable if time step τ satisfies the inequality

$$\tau\left(\frac{v_1}{h_1} + \frac{v_2}{h_2}\right) \le 1. \tag{2.8}$$

The modified equation for the two-dimensional explicit Euler scheme is given as

$$\frac{\partial w}{\partial t} + v_1\frac{\partial w}{\partial x_1} + v_2\frac{\partial w}{\partial x_2} = a\frac{\partial^2 w}{\partial x_1^2} + b\frac{\partial^2 w}{\partial x_1 \partial x_2} + c\frac{\partial^2 w}{\partial x_2^2}, \tag{2.9}$$

$$a = \frac{v_1}{2}\left(h_1 - v_1\tau\right), \quad b = -v_1 v_2 \tau, \quad c = \frac{v_2}{2}\left(h_2 - v_2\tau\right).$$

The right-hand side of this equation is a positively defined quadratic form if

$$4ac - b^2 = v_1 v_2\left(h_1 h_2 - (h_1 v_2 + h_2 v_1)\tau\right) > 0.$$

We see that this condition coincides with the stability requirement (2.8).

By rewriting the quadratic form defined by the right-hand side of equation (2.9) in a canonical form, we can estimate qualitatively the influence of the

numerical diffusion. Let us consider the following example of oblique linear hyperbolic equation and the discrete grid:

$$v_1 = v_2 = v, \quad h_1 = h_2 = h.$$

We introduce a new system of coordinates

$$y_1 = \frac{\sqrt{2}}{2}x_1 + \frac{\sqrt{2}}{2}x_2, \quad y_2 = -\frac{\sqrt{2}}{2}x_1 + \frac{\sqrt{2}}{2}x_2.$$

Then we rewrite the modified equation for the two-dimensional explicit Euler scheme as

$$\frac{\partial w}{\partial t} + \sqrt{2}v\frac{\partial w}{\partial y_1} = \frac{v}{2}(h - 2v\tau)\frac{\partial^2 w}{\partial y_1^2} + \frac{v}{2}h\frac{\partial^2 w}{\partial y_2^2}. \tag{2.10}$$

It follows from this equation that for $2v\tau \sim h$ the numerical diffusion is much larger in the direction orthogonal to the characteristic. If $2v\tau \ll h$, then the ellipse defined by the quadratic form on the right-hand side of (2.10) is close to a circle.

We can explicitly calculate the solution of (2.10). For simplicity of presentation let us assume that the influence of boundary conditions is neglected, i.e. the problem is solved in a plane, then we get

$$w(y_1, y_2, t) = \frac{1}{4\pi t\sqrt{D_1 D_2}} \int\limits_{-\infty}^{\infty} \int\limits_{-\infty}^{\infty} \tilde{u}_0(z, s)\, e^{-\frac{(z - y_1 + \sqrt{2}vt)^2}{4D_1 t}} e^{-\frac{(s - y_2)^2}{4D_2 t}}\, dzds,$$

$$D_1 = \frac{v}{2}(h - 2v\tau), \quad D_2 = \frac{v}{2}h, \quad \tilde{u}_0(y_1, y_2) = u_0\left(\frac{y_1 - y_2}{\sqrt{2}}, \frac{y_1 + y_2}{\sqrt{2}}\right).$$

2.2.2. Accuracy of the fully implicit Euler scheme

The residual of the discrete scheme is given by

$$R_{IES}(u) := \frac{u_{jk}^{n+1} - u_{jk}^n}{\tau} + Au_{jk}^{n+1}.$$

By using the Taylor series and taking the solution of differential problem (2.7) we get the following estimate of the residual:

$$R_{IES}(u) = -\frac{\tau}{2}\frac{\partial^2}{\partial t^2}u(X_{jk}, t^{n+1} - \theta_0\tau) - \frac{v_1 h_1}{2}\frac{\partial^2}{\partial x_1^2}u(x_{1j} - \theta_1 h_1, x_{2k}, t^{n+1})$$

$$- \frac{v_2 h_2}{2}\frac{\partial^2}{\partial x_2^2}u(x_{1j}, x_{2k} - \theta_2 h_2, t^{n+1}), \quad 0 < \theta_i < 1, \ i = 0, 1, 2.$$

Thus the approximation of the fully implicit Euler scheme is first order in time and space. Applying the maximum principle it is easy to prove that this scheme is unconditionally stable.

The modified equation of the two-dimensional fully implicit Euler scheme is defined as

$$\frac{\partial w}{\partial t} + v_1 \frac{\partial w}{\partial x_1} + v_2 \frac{\partial w}{\partial x_2} = a \frac{\partial^2 w}{\partial x_1^2} + b \frac{\partial^2 w}{\partial x_1 \partial x_2} + c \frac{\partial^2 w}{\partial x_2^2}, \qquad (2.11)$$

$$a = \frac{v_1}{2}\big(h_1 + v_1 \tau\big), \quad b = v_1 v_2 \tau, \quad c = \frac{v_2}{2}\big(h_2 + v_2 \tau\big).$$

The quadratic form on the right-hand side of this equation is positively defined since

$$4ac - b^2 = v_1 v_2 \big(h_1 h_2 + (h_1 v_2 + h_2 v_1)\tau\big) > 0.$$

The unconditional parabolicity of the modified equation coincides with the fact that the fully implicit Euler scheme is unconditionally stable.

Let us consider the following example of oblique linear hyperbolic equation and the discrete grid:

$$v_1 = v_2 = v, \quad h_1 = h_2 = h.$$

Then we rewrite the modified equation for the two-dimensional fully implicit Euler scheme as

$$\frac{\partial w}{\partial t} + \sqrt{2} v \frac{\partial w}{\partial y_1} = \frac{v}{2}\big(h + 2v\tau\big)\frac{\partial^2 w}{\partial y_1^2} + \frac{v}{2}h\frac{\partial^2 w}{\partial y_2^2}.$$

It follows that for $2v\tau \sim h$ numerical diffusion is larger in the direction of characteristics y_1 than in the orthogonal direction y_2. We also note that numerical diffusion in the direction y_2 is the same for both the explicit and implicit Euler schemes. If $2v\tau \ll h$, then the ellipse defined by the quadratic form again is close to a circle.

3. Central–Upwind High Resolution Schemes

In this section we describe the construction of new second-order central upwind schemes for solution of systems of hyperbolic conservation laws. They were first introduced in papers of Tadmor and Jiang [8] and modified in many papers (see [9, 10, 11] and references given in these papers). We consider the scheme from [9] and present it for 1D system of hyperbolic equations (here u is a vector):

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0.$$

We are interested in these schemes, because they are simple, robust and Riemann-problem-solver free methods. Such schemes can be used as a "black-box" solvers in general tools for simulating flows in porous media.

We consider the following semi-discrete central-upwind scheme [9, 10]:

$$\frac{d}{dt}U_j(t) = -\frac{H_{j+1/2}(t) - H_{j-1/2}(t)}{h}, \qquad (3.1)$$

where the numerical fluxes and cell averages of the solution are defined as:

$$H_{j+1/2}(t) = \frac{a_{j+1/2}^{+} f(U_{j+1/2}^{-}) - a_{j+1/2}^{-} f(U_{j+1/2}^{+})}{a_{j+1/2}^{+} - a_{j+1/2}^{-}} \tag{3.2}$$

$$+ \frac{a_{j+1/2}^{+} a_{j+1/2}^{-}}{a_{j+1/2}^{+} - a_{j+1/2}^{-}} (U_{j+1/2}^{+} - U_{j+1/2}^{-}), \quad U_j = \frac{1}{h} \int_{x_{j-1/2}}^{x_{j+1/2}} u(x,t)\, dx,$$

with slight modifications at the boundary cells. Here $a_{j+1/2}^{\pm}$ define the right- and left- sided local speeds of propagation of the corresponding Riemann fans. In the scalar case they can be evaluated exactly:

$$a_{j+1/2}^{+} = \max_{u \in [u_{j+1/2}^{min}, u_{j+1/2}^{max}]} \big(f'(u), 0\big), \quad a_{j+1/2}^{-} = \min_{u \in [u_{j+1/2}^{min}, u_{j+1/2}^{max}]} \big(f'(u), 0\big),$$

where $u_{j+1/2}^{min} = \min(u_{j+1/2}^{-}, u_{j+1/2}^{+})$ and $u_{j+1/2}^{max} = \max(u_{j+1/2}^{-}, u_{j+1/2}^{+})$.

Let us assume that at a time level $t = t^n$ the computed cell averages of the solution U_j^n are available. Then the evolution of the solution to the next time level $t = t^{n+1}$ is done in three steps. We present all formulas, because there are a few misprints in [9].

Reconstruction

Starting with a piecewise–constant solution and using the cell averages $\{U_j^n\}$ one reconstructs a second–order piecewise linear interpolant

$$U(x, t^n) = \sum_{j=0}^{J} \big(U_j^n + S_j^n(x - x_j)\big) \chi_j(x), \tag{3.3}$$

here S_j^n is a slope and $\chi_j(x)$ is the characteristic function over the j-th cell. The piecewise polynomial reconstruction should be non-oscillatory, this is achieved with the help of nonlinear limiters. In all numerical experiments we have used the generalized *minmod* limiter

$$s_j = \text{minmod}\Big(\theta \frac{u_j - u_{j-1}}{h}, \frac{u_{j+1} - u_{j-1}}{2h}, \theta \frac{u_{j+1} - u_j}{h}\Big),$$

$$\text{minmod}(v_1, v_2, \ldots, v_k) = \begin{cases} \min_j v_j, & \text{if } v_j > 0 \ \forall j, \\ \max_j v_j, & \text{if } v_j < 0 \ \forall j, \\ 0, & \text{otherwise.} \end{cases}$$

This limiter with $\theta = 1$ is dissipative and will be referred to as the MM1 limiter. The case $\theta = 2$ corresponds to the most compressive limiter and it will be denoted by MM2.

Evolution

A new evolution operator is defined in [9], where differential equation is integrated separately over nonsmooth (those that include Riemann fans) $[x_{j+1/2,l}, x_{j+1/2,r}]$ and smooth $[x_{j-1/2,r}, x_{j+1/2,l}]$ control volumes, where

$$x_{j+1/2,l} = x_{j+1/2} + a_{j+1/2}^{-}\tau, \quad x_{j+1/2,r} = x_{j+1/2} + a_{j+1/2}^{+}\tau.$$

We obtain the following intermediate cell averages over the nonsmooth areas:

$$V_{j+1/2}^{n+1} = \frac{1}{a_{j+1/2}^{+} - a_{j+1/2}^{-}}\Big(U_{j+1/2,r}^{n}\, a_{j+1/2}^{+} - \frac{S_{j+1}^{n}}{2}(a_{j+1/2}^{+})^2\tau$$

$$- U_{j+1/2,l}^{n}\, a_{j+1/2}^{-} + \frac{S_{j}^{n}}{2}(a_{j+1/2}^{-})^2\tau - f(U_{j+1/2,r}^{n+1/2}) + f(U_{j+1/2,l}^{n+1/2})\Big)$$

and over the smooth areas:

$$V_{j}^{n+1} = U_{j}^{n} + \frac{S_{j}^{n}}{2}(a_{j-1/2}^{+} + a_{j+1/2}^{-})\tau - \frac{\tau\big(f(U_{j+1/2,l}^{n+1/2}) - f(U_{j-1/2,r}^{n+1/2})\big)}{h - (a_{j-1/2}^{+} - a_{j+1/2}^{-})\tau}.$$

The midpoint values $U_{j+1/2,l,r}^{n+1/2}$ are obtained using the Taylor expansions about the corresponding points.

Projection

At this step a piecewise linear interpolant, reconstructed from the computed intermediate cell averages $V_{j+1/2}^{n+1}$, V_{j}^{n+1}, is projected back onto the original grid by exactly integrating over the intervals $[x_{j-1/2}, x_{j+1/2}]$:

$$U_{j}^{n+1} = \frac{\tau}{h}\big(a_{j-1/2}^{+}W_{j-1/2}^{n+1} - a_{j+1/2}^{-}W_{j+1/2}^{n+1}\big) + \Big(1 + \frac{\tau}{h}\big(a_{j+1/2}^{-} - a_{j-1/2}^{+}\big)\Big)W_{j}^{n+1}$$

$$+ \frac{\tau^2}{2h}\Big(s_{j+1/2}^{n+1}a_{j+1/2}^{+}a_{j+1/2}^{-} - s_{j-1/2}^{n+1}a_{j-1/2}^{+}a_{j-1/2}^{-}\Big).$$

We note that in our numerical experiments only scalar equations are solved. In this case the central-upwind scheme (3.1) reduces to a more simple scheme

$$\frac{d}{dt}U_{j}(t) = -\frac{f(U_{j+1/2}^{-}) - f(U_{j-1/2}^{-})}{h}.$$

4. Numerical Experiments

In this section we compare different numerical approximations with analytical solutions for three test hyperbolic problems. Our goal is to determine the experimental order of convergence of different discrete schemes and to investigate their accuracy of shock resolution.

4.1. 1D case

Problem 1. We solve a linear hyperbolic equation

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \qquad (4.1)$$

in the domain $Q_T = (-1, 1) \times (0, 0.5)$. This equation is obtained from the Buckley-Leverett equation (1.1) in the case of linear relative permeability functions $(k_{ri}(S_i) = S_i)$ and viscosity ratio equal to one [7]. We note that linear relative permeabilities are usually employed for the geothermal systems due to their simplicity and sufficient accuracy [14]. The initial condition is defined as

$$u(x, 0) = \begin{cases} 0, & x \in [-1, -0.5] \cup [0, 1], \\ 1, & x \in [-0.5, 0]. \end{cases} \qquad (4.2)$$

Problem 2. We solve the nonlinear Buckley-Leverett equation

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \qquad (4.3)$$

in the domain $Q_T = (-1, 1) \times (0, 0.6)$ subject to the initial condition (4.2). The flux in (4.3) is a nonconvex function [2]

$$f(u) = \frac{4u^2}{4u^2 + (1 - u)^2}.$$

Such flux function is obtained in the case of quadratic relative permeabilities: $k_{ri}(S_i) = S_i^2$. Power functions of phase saturations are widely used in petroleum and nuclear safety engineering [7].

Problem 3. We solve in the domain $Q_T = (-1, 1) \times (0, 1)$ the Buckley-Leverett equation, whose flux is a C^1 monotone increasing nonconvex function [10]

$$f(u) = \begin{cases} \dfrac{1}{4}u(1 - u), & 0 \le u < \dfrac{1}{2}, \\ \dfrac{1}{2}u^2 - \dfrac{1}{2}u + \dfrac{3}{16}, & u \ge \dfrac{1}{2}. \end{cases}$$

The domain Q_T is discretized with the uniform grid $Q_{h\tau}$, and a Courant number $C = v\tau/h$. Let define the L_p-norms of the error of the discrete solution

$$\|u - U\|_{L_p} = \left(\int_{-1}^{1} |u(x, t) - U(x, t)|^p \, dx \right)^{1/p},$$

where $p = 1, 2$ and $U(x, t)$ is a second order piecewise linear interpolant of the discrete solution. The convergence rate ρ_h is determined as

$$\rho_h = \log \left(\frac{\|u - U_h\|_{L_p}}{\|u - U_{2h}\|_{L_p}} \right) / \log \frac{1}{2}.$$

Table 1. Error and experimental order of convergence for Problem 1 with upwind approximation and the explicit Euler time-stepping.

J	C	L_1-error	ρ_{L_1}	L_2-error	ρ_{L_2}
40	0.5	0.18457	0.5457	0.23901	0.3106
80	0.5	0.12835	0.5241	0.19668	0.2813
160	0.5	0.08998	0.5123	0.16353	0.2663
320	0.5	0.06335	0.5062	0.13672	0.2583
40	0.1	0.24342	0.5073	0.27273	0.3026
80	0.1	0.17077	0.5114	0.22560	0.2737
160	0.1	0.12022	0.5064	0.18846	0.2595
320	0.1	0.08482	0.5032	0.15796	0.2547

First we consider results for Problem 1. Table 1 presents the error norms of the discrete solutions obtained using upwind space approximation and the explicit Euler time-stepping.

Table 1 shows that the order of convergence is $\mathcal{O}(h^{1/2})$ in the L_1-norm and $\mathcal{O}(h^{1/4})$ in the L_2-norm. The accuracy of the discrete solution decreases when smaller time steps are used for a fixed space step (i.e. when the Courant number is reduced). These observations agree well with the theoretical conclusions obtained from the analysis of the modified equation in Section 2.1.1.

Table 2 presents the error norms of the discrete solutions obtained using upwind space approximation and the fully implicit Euler time-stepping.

Table 2. Error and experimental order of convergence for Problem 1 with upwind approximation and the implicit Euler time-stepping.

J	C	L_1-error	ρ_{L_1}	L_2-error	ρ_{L_2}
40	1.0	0.34708	0.4007	0.33710	0.2855
80	1.0	0.25147	0.4649	0.27596	0.2887
160	1.0	0.17837	0.4955	0.22946	0.2662
320	1.0	0.12615	0.4998	0.19253	0.2532
40	0.5	0.30749	0.4437	0.31120	0.2948
80	0.5	0.21904	0.4894	0.25584	0.2826
160	0.5	0.15474	0.5013	0.21361	0.2603
320	0.5	0.10934	0.5011	0.17925	0.2530
40	0.1	0.26733	0.4855	0.28669	0.3008
80	0.1	0.18834	0.5053	0.23679	0.2759
160	0.1	0.13275	0.5046	0.19796	0.2587
320	0.1	0.09371	0.5024	0.16598	0.2539

Results show that the order of convergence is $\mathcal{O}(h^{1/2})$ in the L_1-norm and $\mathcal{O}(h^{1/4})$ in the L_2-norm. As it follows from the analysis of modified equations the solution of the explicit Euler scheme is more accurate than the solution of the fully implicit Euler scheme. But the implicit Euler scheme is unconditionally stable, thus it can be used for any Courant number.

Table 3 presents the error norms of the discrete solutions obtained using upwind space approximation and the Crank-Nicolson time-stepping.

Table 3. Error and experimental order of convergence for Problem 1 with upwind approximation and the Crank-Nicolson time-stepping.

J	C	L_1-error	ρ_{L_1}	L_2-error	ρ_{L_2}
40	1.0	0.25354	0.4767	0.27864	0.2915
80	1.0	0.17913	0.5012	0.23085	0.2714
160	1.0	0.12641	0.5028	0.19313	0.2574
320	1.0	0.08930	0.5015	0.16202	0.2534
40	0.1	0.25572	0.4963	0.27986	0.3017
80	0.1	0.17977	0.5084	0.23137	0.2745
160	0.1	0.12664	0.5054	0.19336	0.2589
320	0.1	0.08938	0.5027	0.16212	0.2542

As it follows from the analysis of the modified equation the error of the Crank-Nicolson scheme is dominated by the upwind space discretization.

All three time-stepping schemes give the same accuracy for small Courant numbers.

It is well-known that for Galerkin type schemes the accuracy of a discrete solution is of the same order as optimal approximation order of the exact solution with the given ansatz space (see, e.g. [15]). For a step function with the ansatz space of piecewise linear functions the optimal approximation order is $\mathcal{O}(h)$ in the L_1-norm and $\mathcal{O}(h^{1/2})$ in the L_2-norm. Thus for our finite-volume approximations these orders are not reached and the modified equation explains why the convergence orders are reduced.

Table 4 presents the error norms of the discrete solutions computed using the second-order central-upwind scheme with the compressive *minmod* reconstruction MM2. Results show that the order of convergence is $\mathcal{O}(h^{3/4})$ in the L_1-norm and $\mathcal{O}(h^{2/5})$ in the L_2-norm, thus they are closer to the optimal approximation orders with the given ansatz space.

Next we investigate the shock resolution obtained with the upwind and the second-order central-upwind schemes. Accurate shock resolution is very important for simulation of flows in porous media. The linear hyperbolic equation with a solution defined by step function is a challenging test problem, since for the nonlinear Buckley-Leverett equation the so-called "self-sharpening effect" can improve the shock resolution. Figure 1 shows the numerical solutions in

Table 4. Error and experimental order of convergence for Problem 1 with second-order central-upwind scheme with the *minmod* reconstruction MM2.

J	C	L_1-error	ρ_{L_1}	L_2-error	ρ_{L_2}
40	0.2	0.09874	0.7820	0.18746	0.4314
80	0.2	0.05781	0.7723	0.14056	0.4153
160	0.2	0.03402	0.7650	0.10634	0.4025
320	0.2	0.02016	0.7551	0.08105	0.3918

comparison to the analytical solution. In Figure 1(a) solutions are computed using the upwind space approximation and the explicit Euler time stepping with $J = 40, 80, 160$ and 320, a Courant number $C = 0.5$. In Figure 1(b) numerical solutions are computed using the second-order central-upwind scheme with the *minmod* reconstruction MM2 on the same grids and Courant number $C = 0.25$.
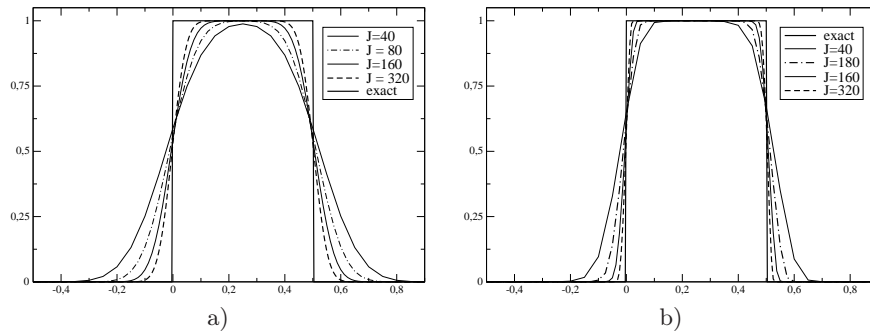
Figure 1. Numerical solution of the linear hyperbolic problem: a) upwind space discretization with the explicit Euler stepping, b) the second-order central-upwind scheme.

It follows from the presented results that the solution profiles obtained by the second-order central-upwind scheme are clearly sharper than the ones computed by the fully upwind scheme.

Next we describe the results obtained for Problem 2. The nonlinear flux in equation (4.3) is a nonconvex function therefore the viscosity solution of the Buckley-Leverett problem consists of two rarefaction waves and two shock fronts, which move with different velocities.

Table 5 presents the error norms of the discrete solutions obtained using upwind space approximation and the explicit Euler time-stepping at time $T = 0.4$.

Table 5. Error and experimental order of convergence for Problem 2 with upwind approximation and the explicit Euler time-stepping.

J	C	L_1-error	ρ_{L_1}	L_2-error	ρ_{L_2}
40	0.25	0.07680	0.8433	0.13210	0.5269
80	0.25	0.04566	0.7501	0.09967	0.4064
160	0.25	0.02624	0.7991	0.07311	0.4471
320	0.25	0.01489	0.8176	0.05358	0.4485

Results show that the order of convergence is $\mathcal{O}(h^{0.8})$ in the L_1-norm and $\mathcal{O}(h^{0.45})$ in the L_2-norm, thus the optimal approximation orders are almost reached.
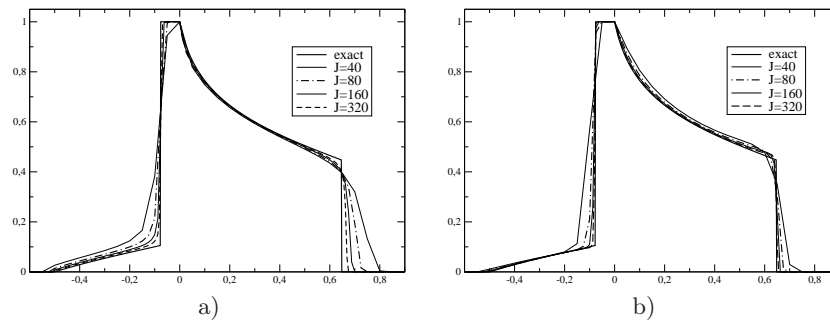

Figure 2. Numerical solution of the Buckley-Leverett problem: a) upwind space discretization with the explicit Euler stepping, b) the second-order central-upwind scheme.

Table 6 presents the error norms of the discrete solutions computed using the second-order central-upwind scheme with the *minmod* reconstruction MM2. Figure 2 shows these numerical solutions in comparison to the analytical solution.

It follows from the analysis of modified equations that the upwind space discretization introduces additional numerical diffusion which helps to pick up the entropy solution from a set of possible solutions [4, 7, 13]. The aim of high-order approximations is to reduce the influence of numerical diffusion, then there is possibility that the solution captured by discrete scheme is not the entropy solution. We apply the upwind and the central-upwind schemes to Problem 3 with non-convex flux function [10]. Figure 3 shows numerical solutions computed with *minmod* reconstruction in comparison to the analytical solution, which is computed by using the upwind space approximation and explicit Euler time stepping with $J = 5000$ grid points. The solution of the central-upwind scheme with compressive *minmod* reconstruction function

Table 6. Error and experimental order of convergence for Problem 2 with second-order central-upwind scheme with the *minmod* reconstruction MM2.

J	C	L_1-error	ρ_{L_1}	L_2-error	ρ_{L_2}
40	0.2	0.05317	0.9842	0.12082	0.6046
80	0.2	0.02713	0.9707	0.09321	0.3743
160	0.2	0.01417	0.9369	0.06410	0.5403
320	0.2	0.00729	0.9590	0.04296	0.5774

MM2 (i.e. $\theta = 2$) is not the entropy solution, but the dissipative MM1 reconstruction (i.e. $\theta = 1$) leads to the convergence towards the unique entropy solution. The grid was refined to reach the convergence.
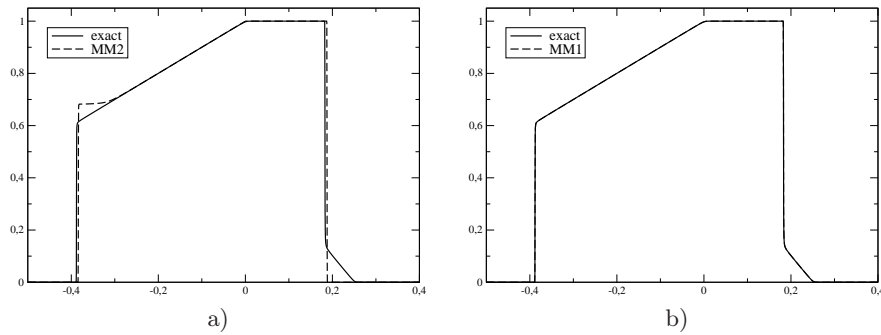
Figure 3. Numerical solution of Problem 3 with the second-order central-upwind scheme a) *minmod* reconstruction function MM2, b) *minmod* reconstruction function MM1.

We have computed solutions of Problem 3 by using various TVD schemes with different limiters. Computations were done with OpenFOAM tool [1]. The following limiters also have captured different non-physical solutions: SuperBee, MUSCL, van Leer.

4.2. Linearization of nonlinear fluxes

Let us consider the nonlinear equation

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0.$$

To solve this equation using programming packages which do not include Newton-type iterative methods (e.g. OpenFOAM tool [1]), the following Picard-type linearization algorithm need to be used:

$$\frac{\partial u^s}{\partial t} + \frac{\partial}{\partial x}\left(\frac{f(u^{s-1})}{u^{s-1}}u^s\right) = 0.$$

A similar situation is obtained when we solve a system of conservation equations. Then the velocity can be computed (e.g. from the Darcy law using the pressure field [4, 5, 7]) and we get a linear hyperbolic equation

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\big(v(x)u\big) = 0. \tag{4.4}$$

Let us assume that $v(x) \geq 0$. Applying the central-upwind discretization to equation (4.4) and the explicit Euler time-stepping we get the following discrete scheme:

$$U_j^{n+1} = U_j^n + \frac{\tau}{h}\Big(v(x_{j+1/2})U_{j+1/2}^{n,-} - v(x_{j-1/2})U_{j-1/2}^{n,-}\Big) = 0. \tag{4.5}$$

Here $U_{j+1/2}^-$ is computed by using a special piecewise polynomial interpolant of an appropriate order. For second-order piecewise linear reconstruction the slope should satisfy the condition $s_j = u_x(x_j,t) + \mathcal{O}(h)$, while to satisfy the non-oscillatory property, the reconstruction should use a nonlinear limiter, e.g. the generalized *minmod* limiter.

Function $v(x)$ is given or values of v_j are known, therefore usually $v(x_{j\pm1/2})$ is computed by a piecewise linear interpolation formula

$$v_{j+1/2} = \frac{1}{2}\big(v_j + v_{j+1}\big).$$

If we solve a nonlinear hyperbolic equation or a system of nonlinear equations, such inconsistency in reconstruction of $v(x_{j\pm1/2})$ and $U_{j\pm1/2}^-$ can lead to discrete solutions which are oscillatory or do not satisfy the entropy condition.
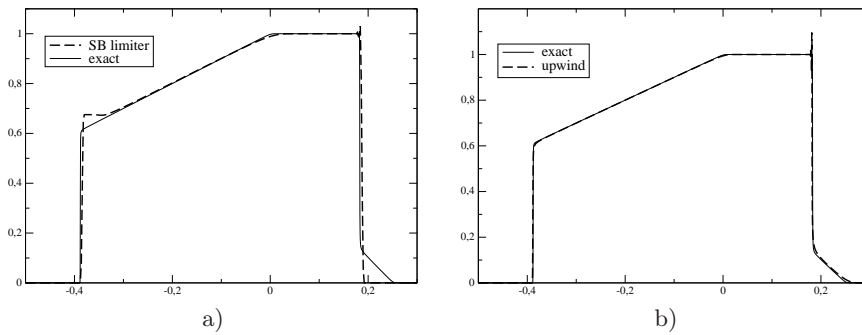
Figure 4. Numerical solution of Problem 3: a) scheme (4.5) with SB limiter and $J = 1000$, b) scheme (4.6) with $J = 5000$.

As an example we solve Problem 3. Figure 4 shows two numerical solutions. The first one is computed with scheme (4.5) and SuperBee limiter

implemented in OpenFOAM tool, the second solution is obtained when the upwind reconstruction is used to compute $U^-_{j\pm1/2}$:

$$U^{n+1}_j = U^n_j + \frac{\tau}{h}\left(v(x_{j+1/2})U^n_j - v(x_{j-1/2})U^n_{j-1}\right) = 0. \qquad (4.6)$$

We see that a spike is formed at the right shock front that cannot be removed by taking smaller time and space discretization steps. For multiphase problems this creates a big problem, because we are getting values bigger than one. If for the functions $v(x_{j\pm1/2})$ and $U^-_{j\pm1/2}$ the same reconstruction interpolant is used then the numerical solution of the first scheme is converging to a non-physical solution (SuperBee limiter!) and solution of the second scheme to the correct physical solution.
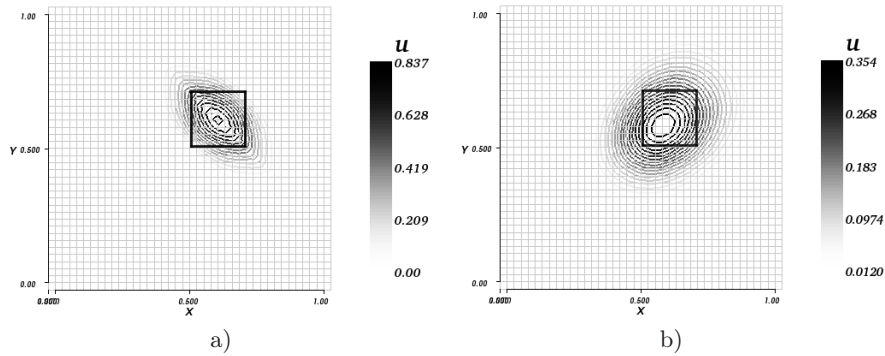
4.3. 2D case

In this section, we numerically solve the 2D linear hyperbolic equation (2.6) with the initial and boundary conditions given by

$$u(x_1, x_2, 0) = \begin{cases} 1, & (x_1, x_2) \in [0.1, 0.3] \times [0.1, 0.3], \\ 0, & \text{otherwise.} \end{cases}$$

$$u(0, x_2, t) = 0, \quad 0 < x_2 < 1,\ 0 < t \le T,$$

$$u(x_1, 0, t) = 0, \quad 0 < x_1 < 1,\ 0 < t \le T.$$

With a constant uniform velocity field $v_1 = v_2 = 1$ this problem describes a 2D transport of initial shape of the solution. We have computed the solution at time $T = 0.4$ on a uniform grid with $h_1 = h_2 = 0.025$ and time step $\tau = 0.0125$ using the explicit and implicit Euler upwind scheme (2.7). Obtained results are presented in Figure 5.

Figure 5. Numerical solution of 2D problem with the upwind scheme: a) explicit Euler scheme, b) implicit Euler scheme.

As we can see, in 2D case such skewed flow-to-mesh alignment causes a big amount of numerical diffusion which is smearing our squared solution profile

to ellipses. The differences between the explicit and implicit schemes can be explained by the analytical conclusions that we have made in Sections 2.2.1 and 2.2.2 considering the first differential approximations of corresponding schemes. As expected, we don't see the numerical diffusion in the direction of characteristic for the explicit Euler scheme and our discretization parameters. On the contrary, for the implicit Euler scheme numerical diffusion in this direction is the biggest.

5. Parallel Algorithm

In this section we consider a parallel version of the central-upwind 2D scheme, which was formulated in Section 3. We restrict to explicit Euler time-stepping. The space discretization is done on staggered grids as in the Tadmor algorithm [8]. The nonlinear hyperbolic equation is solved in 2D region with periodic boundary conditions.

Let us assume that we have p processors, which are connected by two dimensional mesh, i.e. $p = p_1 \times p_2$. The grid ω_h (a data set) is decomposed into a number of 2D subgrids by using a block distribution scheme. Then each subgrid ω_{hp} has

$$\frac{(J+1)}{p_1} \times \frac{(J+1)}{p_2} = \frac{(J+1)^2}{p}$$

computational points of the grid ω_h and it is assigned to one processor, which is responsible for all computations of the local part of vector U.

Since the sub-domains are connected at their boundaries, processors dealing with neighbouring sub-domains have to exchange boundary information with each other at every time-step. The update of vector U^{n+1} at grid points which lie beside cutting planes (i.e. boundary nodes of the local part of the vector U) needs a special attention, since information from the neighbouring processors is required to compute new values of U^{n+1}. Such information is obtained by exchanging data with neighbour processors in the specified topology of processors. The amount of exchanged data depends on the grid stencil, which is used to discretize the PDE model.

We will estimate the complexity of the sequential and parallel algorithms by counting basic operations at one time step. The total complexity of the serial algorithm is expressed as

$$W = J^3 + \mathcal{O}(J^2).$$

The communication step is implemented before updating vectors U^{n+1} and only neighbouring processors are communicating with each other. Each processor exchanges vector elements corresponding to boundary points of the local subdomain. A total amount of data, exchanged between two processors, is equal to $\mathcal{O}(J/\sqrt{p})$ elements. This can be done in

$$T_{1,p}(J) = \alpha + \beta \frac{J}{\sqrt{p}}$$

time, by using the *odd–even* data exchange algorithm. Here α is the message startup time and β is the time required to send one element of data. The total complexity of the parallel algorithm is given by

$$T_p(J) = \frac{J^2}{p} + \alpha + \beta \frac{J}{\sqrt{p}}\,.$$

According to the definition of the isoefficiency function, we must find the rate at which the problem size W needs to grow with p for a fixed efficiency of the algorithm. Let $H(p, W) = pT_p - W$ be the total overhead of a parallel algorithm. Then the *isoefficiency* function $W = g(p, E)$ is defined by the implicit equation:

$$W = \frac{E}{1 - E}\, H(p, W)\,.$$

After simple computations we get the isoefficiency function $W = \mathcal{O}(p)$.

It should be noted that implementation of the nonlinear limiters is quite costly operation, therefore $\beta \approx 1$ and we expect that the efficiency of parallel algorithm will be close to one even on clusters of PCs.

Special tools are developed to simplify parallelization of sequential algorithms, e.g. *Diffpack* tool [12] and *PETSc* toolkit [3]. We have developed new tool *ParSol* of parallel numerical arrays, which can be used for semi–automatic parallelization of data parallel algorithms, that are implemented in `C++`. Such algorithms are constructed for solving PDEs and systems of PDEs on logically regular rectangular grids. *ParSol* is a library of parallel array objects, a functionality of which is similar to *Distributed Arrays* in *PETSc*. We list the following main features of *ParSol* (see [6]):

a) created for `C++` programming language,

b) based on HPF ideology,

c) the library heavily uses such `C++` features as OOP, templates and expression templates. The efficiency of *ParSol* arrays is very close to native C arrays.

d) MPI 1.1 standard is used to implement parallelization.

ParSol arrays have a number of advantages for programming mathematical algorithms, such as virtual indexing, built-in array operations, automated management of dynamically allocated memory, periodic boundary conditions. A subclass of *ParSol* arrays consists of vectors, which simulate numerical objects of linear algebra. Many useful basic vector operations are supported within the *ParSol* library, e.g. parallel computation of vector norms, the inner product of two vectors, scaling of vectors. A class of banded matrix is also implemented in the tool, this enables efficient realization of parallel iterative algorithms.

Table 7 presents experimental speedup $S_p(J)$ and efficiency $E_p(J)$ values for solving 2D nonlinear hyperbolic problem on PC cluster "Vilkas" in Vilnius Gediminas technical university. Here p is the number of processors,

$$S_p(J) = \frac{T_1(J)}{T_p(J)}, \quad E_p(J) = \frac{S_p(J)}{p}$$

and $T_p(J)$ is CPU time required to solve the problem of size $J \times J$ with p processors.

Table 7. The speedup and efficiency for explicit algorithm on PC cluster.

p	$T_p(640)$	$S_p(640)$	$E_p(640)$
1	111.0	1.00	1.000
2	55.4	2.00	1.000
4	27.5	4.03	1.010
8	13.9	7.98	0.998

Results, presented in Table 7, confirm our theoretical predictions about the speed-up and efficiency of the parallel algorithm.

6. Conclusions

This study confirms that the standard upwind method is very robust. Although the numerical diffusion reduces the convergence rate of a numerical solutions and is heavily smearing sharp solution profiles (especially in linear and multidimensional cases), but it also ensures the convergence to a correct physical solution. High-resolution schemes show a very promising results in our tests: convergence rates are close to optimal, steep fronts are well captured. All this is encouraging a further research in this direction. However, a danger of picking up non-physical solutions appears for many popular and widely used limiters. It will be interesting to see whether the small diffusive terms from capillary effects will ensure the convergence to physical vanishing viscosity solutions and eliminate the need in numerical diffusion. Although the high-resolution methods are more heavy computationally, they seems to be well suited for parallelization.

Acknowledgements

References

[1] Openfoam home page. http://www.opencfd.co.uk/openfoam.

[2] A. Balaguer and C. Conde. Fourth-order nonoscillatory upwind and central schemes for hyperbolic conservation laws. *SIAM J. Numer. Anal.*, **43**(2), 455–473, 2005.

[3] S. Balay, K. Buschelman, V. Eijkhout, W.D. Gropp, D. Kaushik, M.G. Knepley, L. Curfman McInnes, B.F. Smith and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 2.3.0, Argonne National Laboratory, 2005.

[4] P. Bastian. *Numerical Computation of Multiphase Flows in Porous Media.* Habilitation Dissertation, Kiel university, 1999.

[5] R. Čiegis, O. Iliev, V. Starikovičius and K. Steiner. Numerical algorithms for solving problems of multiphase flows in porous media. *Mathematical Modelling and Analysis*, **11**(2), 133–148, 2006.

[6] R. Čiegis, A. Jakušev and V. Starikovičius. Parallel tool for solution of multiphase flow problems. In: *Lecture notes in computer science. Sixth International conference on Parallel Processing and Applied Mathematics*, volume 3911. Springer, 312–319, 2006.

[7] R. Helmig. *Multiphase Flow and Transport Processes in the Subsurface – A Contribution to the Modelling of Hydrosystems.* Springer-Verlag, 1997.

[8] G. S. Jiang and E. Tadmor. Nonoscillatory central schemes for multidimensional hyperbolic conservation laws. *SIAM J. Sci. Comput.*, **19**(6), 1892–1917, 1998.

[9] A. Kurganov and Chi-Tien Lin. On the reduction of numerical dissipation in central-upwind schemes. *Comm. in Comput. Phys.*, **2**(1), 141–163, 2007.

[10] A. Kurganov, G. Petrova and B. Popov. Adaptive semi-discrete central-upwind schemes for nonconvex hyperbolic conservation laws. *SIAM J. Sci. Comput., submitted.*

[11] A. Kurganov and E. Tadmor. New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comput. Phys.*, **160**, 241–282, 2000.

[12] H.P. Langtangen and A. Tveito. *Advanced Topics in Computational Partial Differential Equations. Numerical Methods and Diffpack Programming.* Springer, Berlin, 2003.

[13] R.J. LeVeque. *Numerical Methods for Conservation Laws.* Birkhäuser Verlag, 1991.

[14] C.H. Sondergeld and D.L. Turcotte. An experimental study for two-phase convection in a porous medium with applications to geological problems. *J. Geophys. Res.*, **82**, 2045–2053, 1977.

[15] V. Thomee. *Galerkin Finite Element Methods for Parabolic Problems.* Springer, 1997.

[16] B. van Leer. Upwind and high-resolution methods for compressible flow: From donor cell to residual-distribution schemes. *Commun. Comput. Phys.*, **1**, 192–206, 2006.