

ANGEWANDTE MATHEMATIK
UND
INFORMATIK

A New Inflow-Implicit/Outflow-Explicit Finite
Volume Method for Solving Variable Velocity
Advection Equations

Karol Mikula and Mario Ohlberger

01/10 - N

UNIVERSITÄT MÜNSTER

A NEW INFLOW-IMPLICIT/OUTFLOW-EXPLICIT FINITE VOLUME METHOD FOR SOLVING VARIABLE VELOCITY ADVECTION EQUATIONS

KAROL MIKULA * AND MARIO OHLBERGER †

Abstract. We introduce a new method for solving non-stationary advection equations. The new method is based on finite volume space discretization and a semi-implicit discretization in time. Its basic idea is that outflow from a cell is treated explicitly while inflow is treated implicitly. This is natural, since we know what is outflowing from a cell at the old time step but we leave the method to resolve a system of equations determined by the inflows to a cell to obtain the solution values at the new time step. Our new method is exact for constant velocity transport of quadratic functions in any dimension and for any length of a time step and it is second order accurate for smooth solutions in general. The matrix of the system is determined by the inflow fluxes which results in a M-matrix yielding favourable stability properties for the scheme. The method allows large time steps at a fixed spatial grid without losing stability and not deteriorating precision. This makes the new method attractive for practical applications. The scheme is well suited for variable velocity vector fields in higher dimensions and for nonlinear advection problems which is documented by a series of numerical experiments.

Key words. advection equations, variable velocity, nonlinear conservation laws, level-set method, finite volume method, semi-implicit scheme.

AMS subject classifications. 35K20, 35L04, 35L60, 76M12, 76R50, 65M08.

1. Introduction. In this article we propose a new inflow-implicit/outflow-explicit (I²OE) method for solving general time dependent variable velocity advection equations of the form

$$u_t + \mathbf{v} \cdot \nabla u = 0 \tag{1.1}$$

where $u \in \mathbb{R}^d \times [0, T]$ is the unknown function and \mathbf{v} is a vector field which may vary in space and/or it may depend on the solution or its gradient itself, i.e. $\mathbf{v} = \mathbf{v}(x, u, \nabla u)$. Variable velocity vector fields arise in many applications, e.g. in transport with non-divergence free velocity [5, 7] or in image segmentation by active contours in the form of the generalized subjective surface method [1]. Typical nonlinear models are given e.g. by the one dimensional Burgers equation for $\mathbf{v} = u$ [5], representing a generic nonlinearity in computational fluid dynamics problems, or by the motion of level sets in normal direction for $\mathbf{v} = F(x) \frac{\nabla u}{|\nabla u|}$. Such equations are used in the level-set formulation of interface motion, in free boundary problems of multiphase dynamics and many other applications [7]. In the special case of level sets for motion in normal direction, the new method coincides with the semi-implicit forward-backward diffusion approach recently presented in [6]. Hence, the new method can be seen as a generalization of the latter approach to arbitrary variable velocity advection equations.

The basic idea of our new method is that outflow from a cell is treated explicitly while inflow is treated implicitly. Such an approach is natural, since we know what is flowing out from a cell at an old time step $n - 1$ but we leave the method to resolve a system of equations determined by the inflows to obtain a new value in the cell

*Department of Mathematics, Faculty of Civil Engineering, Slovak University of Technology, Radlinského 11, 81368 Bratislava, Slovakia (karol.mikula@stuba.sk).

† Institut für Numerische und Angewandte Mathematik, Universität Münster, Einsteinstr. 62, D-48149 Münster, Germany (mario.ohlberger@uni-muenster.de).

at time step n . Since the matrix of the system is determined by the inflow fluxes it is an M-matrix for Voronoi like grids and thus it has favourable discrete minimum-maximum properties. Consequently, the method allows large time steps at a fixed spatial grid without losing stability. Interestingly, numerical experiments show that the I²OE scheme is exact for constant velocity transport of any quadratic function in any dimension and for any length of a time step. In general, it is second order accurate for smooth solutions without stability restrictions on the time step, both for variable velocity and nonlinear advection problems. A comparison with the second order Lax-Wendroff method for variable velocity shows good properties of our new scheme with respect to precision and CPU times. Combining the new scheme with the Crank-Nicolson scheme for an additional diffusion term, we get a new stable implicit second-order method for general advection-diffusion equations such as the viscous Burgers problem.

The rest of the article is organized as follows. In Section 2 we introduce the general formulation of the I²OE method on unstructured grids in several space dimensions. We discuss special cases and implementation aspects in Section 3. Finally, in Section 4 we present several representative numerical experiments. They reveal the very nice approximation and stability properties of the new method and demonstrate the efficiency and applicability to a wide range of variable velocity transport equations.

2. The new inflow-implicit/outflow-explicit scheme. In this section we derive our new scheme in a general higher dimensional setting. Several special cases in one space dimension will be treated in Section 3.

Let us consider equation (1.1) in a bounded polygonal domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$, and time interval $[0, T]$. Let \mathcal{Q}_h denote a primal polygonal partition of Ω . Let p be a finite volume (cell) of a corresponding dual Voronoi tessellation \mathcal{T}_h with measure m_p and let e_{pq} be an edge between p and q , $q \in N(p)$, where $N(p)$ is a set of neighbouring finite volumes (i.e. $\bar{p} \cap \bar{q}$ has nonzero $(d-1)$ -dimensional measure). Let c_{pq} be the length of e_{pq} and n_{pq} be the unit outer normal vector to e_{pq} with respect to p . We shall consider \mathcal{T}_h to be an admissible mesh in the sense of [2], i.e., there exists a representative point x_p in the interior of every finite volume p such that the joining line between x_p and x_q , $q \in N(p)$, is orthogonal to e_{pq} . We denote by x_{pq} the intersection of this line segment with the edge e_{pq} . The length of this line segment is denoted by d_{pq} , i.e. $d_{pq} := |x_q - x_p|$. As we have build \mathcal{T}_h based on the primal mesh \mathcal{Q}_h , we assume that the points x_p coincide with the vertices of \mathcal{Q}_h . Let us denote by u_p a (constant) value of the solution in a finite volume p computed by the scheme. For the solution representation inside the finite volume p we use either this value u_p or a reconstructed (but again constant) value denoted by \bar{u}_p . A constant value of the solution assigned to the edge e_{pq} (given again by a reconstruction) is denoted by \bar{u}_{pq} .

In order to motivate our new scheme, let us rewrite (1.1) in the formally equivalent form with conserving and non-conserving parts [3, 4]

$$u_t + \nabla \cdot (\mathbf{v}u) - u \nabla \cdot \mathbf{v} = 0. \quad (2.1)$$

Integrating (2.1) over a finite volume p then yields

$$\int_p u_t \, dx + \int_p \nabla \cdot (\mathbf{v}u) \, dx - \int_p u \nabla \cdot \mathbf{v} \, dx = 0.$$

Using a constant representation of the solution on the cell p denoted by \bar{u}_p and

applying the divergence theorem we get

$$\int_p u_t \, dx + \sum_{q \in N(p)} \int_{e_{pq}} u \mathbf{v} \cdot \mathbf{n}_{pq} \, ds - \bar{u}_p \sum_{q \in N(p)} \int_{e_{pq}} \mathbf{v} \cdot \mathbf{n}_{pq} \, ds = 0.$$

Denoting by \bar{u}_{pq} another representative constant value of the solution on the interface e_{pq} , we further get

$$\int_p u_t \, dx + \sum_{q \in N(p)} \bar{u}_{pq} \int_{e_{pq}} \mathbf{v} \cdot \mathbf{n}_{pq} \, ds - \bar{u}_p \sum_{q \in N(p)} \int_{e_{pq}} \mathbf{v} \cdot \mathbf{n}_{pq} \, ds = 0.$$

If we denote the integrated fluxes in the inward normal direction to the finite volume p by

$$\bar{v}_{pq} = - \int_{e_{pq}} \mathbf{v} \cdot \mathbf{n}_{pq} \, ds, \quad (2.2)$$

we finally arrive at the balance law

$$\int_p u_t \, dx + \sum_{q \in N(p)} \bar{v}_{pq} (\bar{u}_p - \bar{u}_{pq}) = 0. \quad (2.3)$$

The major new idea of our scheme is to split the resulting fluxes into the corresponding inflow and outflow parts to the cell p . This is done by defining

$$a_{pq}^{in} = \max(\bar{v}_{pq}, 0), \quad a_{pq}^{out} = \min(\bar{v}_{pq}, 0). \quad (2.4)$$

We then approximate u_t by the time difference $\frac{u_p^n - u_p^{n-1}}{\tau}$, where τ is a uniform time step size, and take the inflow parts implicitly and the outflow parts explicitly in (2.3). This yields the following system of equations for the finite volume solution $u_p^n, p \in \mathcal{T}_h$ at the n -th discrete time step

$$m_p u_p^n + \tau \sum_{q \in N(p)} a_{pq}^{in} (\bar{u}_p^n - \bar{u}_{pq}^n) = m_p u_p^{n-1} - \tau \sum_{q \in N(p)} a_{pq}^{out} (\bar{u}_p^{n-1} - \bar{u}_{pq}^{n-1}) \quad (2.5)$$

for all $p \in \mathcal{T}_h$.

The most natural choice for reconstructions \bar{u}_p^n and \bar{u}_{pq}^n at any time step n (i.e. old and new time steps) is given by

$$\bar{u}_p^n = u_p^n, \quad \bar{u}_{pq}^n = \frac{1}{2}(u_p^n + u_q^n) \quad (2.6)$$

and leads to the basic **I²OE scheme**:

$$m_p u_p^n + \frac{\tau}{2} \sum_{q \in N(p)} a_{pq}^{in} (u_p^n - u_q^n) = m_p u_p^{n-1} - \frac{\tau}{2} \sum_{q \in N(p)} a_{pq}^{out} (u_p^{n-1} - u_q^{n-1}) \quad (2.7)$$

which for a uniform squared grids in two space dimensions with a finite volume side width h reduces to the following simple system

$$u_p^n + \frac{\tau}{2h^2} \sum_{q \in N(p)} a_{pq}^{in} (u_p^n - u_q^n) = u_p^{n-1} - \frac{\tau}{2h^2} \sum_{q \in N(p)} a_{pq}^{out} (u_p^{n-1} - u_q^{n-1}). \quad (2.8)$$

We summarize our new inflow-implicit/outflow-explicit method in the following definition.

Definition 2.1 (The new I²OE scheme) Let initial data $u_0 \in C^0(\Omega)$ and Dirichlet boundary data $u_D \in C^0(\partial\Omega \times [0, T])$ be given. Furthermore, let R_p and R_{pq} denote suitable local reconstructions of the solution on the cell p and the interface e_{pq} , respectively and g_{pq} denote a suitable numerical flux that approximates $-\int_{e_{pq}} \mathbf{v} \cdot \mathbf{n}_{pq} ds$. These operators may differ depending on the inflow or outflow character of the cell interface (such dependence will be denoted by superscripts *in* and *out*).

Then the general inflow-implicit/ outflow-explicit method (I²OE) is defined as follows:

Initial data: For $n = 0$ define the piecewise constant approximation u_h^0 through

$$u_h^0|_p(x) := u_p^0 := \pi_p(u_0), \quad \forall x \in p, p \in \mathcal{T}_h, \quad (2.9)$$

where $\pi_p : C^0(p) \rightarrow \mathbb{P}_0(p)$ is a suitable local projection to a constant.

Time step ($\mathbf{n} - 1$) \rightarrow \mathbf{n} : For $n > 0$ we define u_h^n through $u_p^n, p \in \mathcal{T}_h$ as follows

a) Definition of boundary values at time t^n : For all $x_p \in \partial\Omega$ we set

$$u_p^n := u_D(x_p, t^n). \quad (2.10)$$

b) Definition of the interior values at time t^n :

i) Inflow/outflow splitting of the fluxes. For all interfaces e_{pq} we define

$$a_{pq}^{in,n} = \max(g_{pq}^{in}(\mathbf{v}, u_h^n), 0), \quad a_{pq}^{out,n-1} = \min(g_{pq}^{out}(\mathbf{v}, u_h^{n-1}), 0). \quad (2.11)$$

ii) For all $x_p \in \Omega \setminus \partial\Omega$ we define u_p^n as the solution of the following linear system

$$\begin{aligned} u_p^n + \frac{\tau}{m_p} \sum_{q \in N(p)} a_{pq}^{in,n} (R_p^{in}(u_h^n) - R_{pq}^{in}(u_h^n)) \\ = u_p^{n-1} + \frac{\tau}{m_p} \sum_{q \in N(p)} a_{pq}^{out,n-1} (R_p^{out}(u_h^{n-1}) - R_{pq}^{out}(u_h^{n-1})). \end{aligned} \quad (2.12)$$

Note that (2.11) and (2.12) may result in a non-linear system of equations for u_h^n , if the velocity field depends on the solution.

b) Definition of u_h^n :

We define the piecewise constant approximation u_h^n as

$$u_h^n|_p(x) := u_p^n, \quad \forall x \in p, p \in \mathcal{T}_h.$$

A specific I²OE scheme is obtained by specifying the reconstruction operators $R_p^{in}, R_{pq}^{in}, R_p^{out}, R_{pq}^{out}$ and the numerical flux function $g_{pq}^{in}, g_{pq}^{out}$. The most natural choice of reconstruction operators are given in (2.6) and (2.2), which are used in all computations presented in section 4, but more sophisticated choices are possible, cf. [6].

Few additional remarks on the new scheme should be made. Equation (2.3) has the form of a discretization of a diffusion equation, where \bar{v}_{pq} would represent the so-called transmissive coefficients (integrated diffusion fluxes divided by distances between cell centers). In standard forward diffusion all these coefficients are strictly positive which leads to a weighted averaging of the solution and the implicit schemes are natural in this case. On the other hand the negative coefficients would correspond to backward diffusion in which case information propagates outside the cell and explicit schemes are thus natural. In our case the sign of the coefficients is given by

the inflow or outflow character of the cell boundary and the inflow-implicit/outflow-explicit approach is thus natural. This forward-backward diffusion relationship with the inflow-outflow fluxes comes from [6] where the forward-backward diffusion formulation was used for solving advective motion of level sets in normal direction. Let us also note that the scaling h^{d-1} is included in the definition of the numerical fluxes \bar{v}_{pq} . Thus, after dividing by this scaling, we would have a τ/h term in front of the sums in (2.8) which is the standard term in advection discretization schemes. It is also well-known that in the second order schemes for solving advection problems one can identify the "forward diffusion" part (like the first order upwinding) and the "backward diffusion" part (additional sharpening terms coming (sometimes surprisingly) from the second order Taylor's expansions [5]), see e.g. the Lax-Wendroff scheme in Section 3. In our method this splitting arises naturally, gives second order accuracy and when treating it semi-implicitly it brings stability of the computations. Since the "backward diffusion" part of the scheme is clearly given, one can also use limiters straightforwardly if it is necessary to suppress some oscillations.

3. Special cases of the I²OE scheme.

3.1. I²OE scheme for 1D variable velocity case. Let us derive the scheme for the one-dimensional equation

$$u_t + vu_x = 0, \quad (3.1)$$

where $v = v(x)$. Again (3.1) is written in the form

$$u_t + (vu)_x - uv_x = 0, \quad (3.2)$$

and integrated in p_i , the cell with the spatial index i , length h , center point x_i , left border $x_{i-\frac{1}{2}}$ and right border $x_{i+\frac{1}{2}}$. Let us denote $v_i = v(x_i)$, $v_{i-\frac{1}{2}} = v(x_{i-\frac{1}{2}})$, $v_{i+\frac{1}{2}} = v(x_{i+\frac{1}{2}})$, u_i^n the value of the numerical solution at time step n and $\bar{u}_i^n, \bar{u}_{i-\frac{1}{2}}^n$ the reconstructed values. Using the Newton-Leibniz formula we get

$$\int_{p_i} u_t dx + v_{i+\frac{1}{2}} \bar{u}_{i+\frac{1}{2}} - v_{i-\frac{1}{2}} \bar{u}_{i-\frac{1}{2}} - \bar{u}_i (v_{i+\frac{1}{2}} - v_{i-\frac{1}{2}}) = 0,$$

which can be rewritten as

$$\int_{p_i} u_t dx + v_{i-\frac{1}{2}} (\bar{u}_i - \bar{u}_{i-\frac{1}{2}}) + (-v_{i+\frac{1}{2}}) (\bar{u}_i - \bar{u}_{i+\frac{1}{2}}) = 0.$$

If $v_{i-\frac{1}{2}} > 0$ it represents inflow from the left to the cell and if $(-v_{i+\frac{1}{2}}) > 0$ it represents inflow from the right to the cell. If the signs are opposite it represents outflows. Thus, we define

$$\begin{aligned} a_{i-\frac{1}{2}}^{in} &= \max(v_{i-\frac{1}{2}}, 0), & a_{i-\frac{1}{2}}^{out} &= \min(v_{i-\frac{1}{2}}, 0), \\ a_{i+\frac{1}{2}}^{in} &= \max(-v_{i+\frac{1}{2}}, 0), & a_{i+\frac{1}{2}}^{out} &= \min(-v_{i+\frac{1}{2}}, 0). \end{aligned}$$

If we use a finite difference approximation of the time derivative, take inflow implicitly and outflow explicitly and use the simple reconstructions $\bar{u}_i^n = u_i^n$, $\bar{u}_{i-\frac{1}{2}}^n = \frac{1}{2}(u_i^n + u_{i-1}^n)$ in both time steps, we end up with the basic **one-dimensional I²OE scheme**:

$$\begin{aligned} u_i^n + \frac{\tau}{2h} a_{i-\frac{1}{2}}^{in} (u_i^n - u_{i-1}^n) + \frac{\tau}{2h} a_{i+\frac{1}{2}}^{in} (u_i^n - u_{i+1}^n) = \\ u_i^{n-1} - \frac{\tau}{2h} \left(a_{i-\frac{1}{2}}^{out} (u_i^{n-1} - u_{i-1}^{n-1}) + a_{i+\frac{1}{2}}^{out} (u_i^{n-1} - u_{i+1}^{n-1}) \right). \end{aligned} \quad (3.3)$$

The scheme (3.3) requires to solve a tridiagonal system in every time step which is very fast using the standard tridiagonal solver (also called the Thomas algorithm).

Remark. The I²OE scheme allows to use much larger time steps without losing L_∞ -stability than given by a standard CFL condition for explicit schemes, cf. Section 4. However, the "backward diffusion" (outflow) explicit part is not necessarily always dominated by the implicit part in the basic form of the scheme (3.3). Some oscillations may arise e.g. on coarse grids in nonlinear problems or in solutions tending to a shock. The oscillations are usually not unboundedly growing in time and thus we can leave the method with oscillations and remove them at the end of computations using e.g. edge preserving filters. Another approach is to suppress any oscillation during the computation. In our scheme it can be done in two ways. The first approach is to use an averaging in the reconstruction of \bar{u}_i^n , e.g. by setting $\bar{u}_i^n = 0.5(\bar{u}_{i-\frac{1}{2}}^n + \bar{u}_{i+\frac{1}{2}}^n)$ where the simple reconstruction (2.6) or a more complicated reconstruction of $\bar{u}_{i-\frac{1}{2}}^n$ and $\bar{u}_{i+\frac{1}{2}}^n$ can be used, cf. FBD schemes from [6].

The second approach is to use standard limiters. The "backward diffusion" part on the right hand side of (3.3) can be modified by using e.g. the superbee limiter in the following straightforward manner. First we compute, see e.g. [5],

$$\theta_{i-\frac{1}{2}}^{n-1} = \frac{u_{i-1}^{n-1} - u_{i-2}^{n-1}}{u_i^{n-1} - u_{i-1}^{n-1}}, \text{ if } v_{i-\frac{1}{2}} \geq 0, \quad \theta_{i-\frac{1}{2}}^{n-1} = \frac{u_{i+1}^{n-1} - u_i^{n-1}}{u_i^{n-1} - u_{i-1}^{n-1}}, \text{ if } v_{i-\frac{1}{2}} < 0$$

and then we modify (3.3) to its limited version

$$\begin{aligned} & u_i^n + \frac{\tau}{2h} a_{i-\frac{1}{2}}^{in} (u_i^n - u_{i-1}^n) + \frac{\tau}{2h} a_{i+\frac{1}{2}}^{in} (u_i^n - u_{i+1}^n) = \\ & u_i^{n-1} - \frac{\tau}{2h} (a_{i-\frac{1}{2}}^{out} (u_i^{n-1} - u_{i-1}^{n-1}) \phi(\theta_{i-\frac{1}{2}}) + a_{i+\frac{1}{2}}^{out} (u_i^{n-1} - u_{i+1}^{n-1}) \phi(\theta_{i+\frac{1}{2}})) \end{aligned} \quad (3.4)$$

where the limiter function may be chosen as $\phi(\theta) = \max(0, \min(1, 2\theta), \min(2, \theta))$. Also other limiter functions ϕ are possible, leading to a whole class of limited schemes. Using $\phi(\theta) \equiv 1$ gives the basic scheme (3.3).

To complete this section, we present the standard second order Lax-Wendroff explicit finite difference scheme for variable velocity advection equation (1.1) which is used for comparisons in Section 4. If we denote $v_i^+ = \max(v_i, 0)$, $v_i^- = \min(v_i, 0)$, it can be written as follows:

$$\begin{aligned} & f_{i-\frac{1}{2}}^n = (1 - \frac{\tau}{h} |v_{i-\frac{1}{2}}|) (u_i^n - u_{i-1}^n) \phi(\theta_{i-\frac{1}{2}})^n, \\ & u_i^{n+1} = u_i^n - \frac{\tau}{h} (v_i^+ (u_i^n - u_{i-1}^n) + v_i^- (u_{i+1}^n - u_i^n)) - \frac{\tau}{2h} |v_i| (f_{i+\frac{1}{2}}^n - f_{i-\frac{1}{2}}^n). \end{aligned} \quad (3.5)$$

3.2. I²OE scheme for 1D viscous Burgers' equation. Let us now consider the non-linear equation

$$u_t + u u_x = \sigma u_{xx} \quad (3.6)$$

where the velocity $v = u$ in (3.1) and σ is a (positive) diffusion coefficient. For the advective part we use the same derivation as in (3.3) but considering the time dependent velocities

$$\bar{v}_{i-\frac{1}{2}}^n = (u_i^n + u_{i-1}^n)/2, \quad \bar{v}_{i+\frac{1}{2}}^n = (u_i^n + u_{i+1}^n)/2,$$

and also time dependent splitting to inflows and outflows

$$\begin{aligned} a_{i-\frac{1}{2}}^{in,n} &= \max(\bar{v}_{i-\frac{1}{2}}^n, 0), & a_{i-\frac{1}{2}}^{out,n} &= \min(\bar{v}_{i-\frac{1}{2}}^n, 0), \\ a_{i+\frac{1}{2}}^{in,n} &= \max(-\bar{v}_{i+\frac{1}{2}}^n, 0), & a_{i+\frac{1}{2}}^{out,n} &= \min(-\bar{v}_{i+\frac{1}{2}}^n, 0). \end{aligned}$$

In order to keep second order accuracy of I²OE scheme, the diffusion part is treated by the Crank-Nicolson approach, and we end up with the following (nonlinear) system

$$\begin{aligned} u_i^n + \frac{\tau}{2h} (a_{i-\frac{1}{2}}^{in,n} + \frac{\sigma}{h}) (u_i^n - u_{i-1}^n) + \frac{\tau}{2h} (a_{i+\frac{1}{2}}^{in,n} + \frac{\sigma}{h}) (u_i^n - u_{i+1}^n) = & (3.7) \\ u_i^{n-1} - \frac{\tau}{2h} \left((a_{i-\frac{1}{2}}^{out,n-1} + \frac{\sigma}{h}) (u_i^{n-1} - u_{i-1}^{n-1}) + (a_{i+\frac{1}{2}}^{out,n-1} + \frac{\sigma}{h}) (u_i^{n-1} - u_{i+1}^{n-1}) \right). \end{aligned}$$

This system is solved iteratively updating $a_{i-\frac{1}{2}}^{in,n}$ and $a_{i+\frac{1}{2}}^{in,n}$ using subsequent values of the iterative solution, starting iterations by u^{n-1} . It turns out that 5 nonlinear iterations inside the scheme (3.7) are sufficient to get overall second order precision. It also means that we have to solve few times a tridiagonal system in every time step of the I²OE scheme in case of nonlinear advection problems.

3.3. I²OE scheme for motion of level sets in normal direction. In this case we consider equation (1.1) with the velocity depending on the gradient of solution, i.e. $\mathbf{v} = F \frac{\nabla u}{|\nabla u|}$. Let us consider uniform grid with finite volume side width h and define

$$\bar{v}_{pq} = Fh \frac{(\bar{u}_p^{n-1} - \bar{u}_{pq}^{n-1})}{(h/2)|\nabla u_{pq}^{n-1}|} = \frac{2F(\bar{u}_p^{n-1} - \bar{u}_{pq}^{n-1})}{|\nabla u_{pq}^{n-1}|} \quad (3.8)$$

where $|\nabla u_{pq}^{n-1}|$ is computed by the diamond-cell strategy as in [6]. Plugging expression (3.8) for integrated inward fluxes into the basic I²OE scheme (2.8) and comparing it with the scheme (2.5) in [6] we see that they form exactly the same system. One can also see the correspondence of the forward diffusion coefficients in [6] and inflow coefficients in I²OE scheme (2.8) respectively the backward diffusion coefficients in [6] and outflow coefficients in I²OE scheme (2.8). The FBD and FBD2 schemes from [6] are obtained using their reconstruction stencils (Definitions 2.3 and 3.1), and the role of the forward diffusion contribution D_p^f respectively the backward diffusion contribution D_p^b in general Definition 2.1 of [6] is played by the total inflow D_p^{in} respectively the total outflow D_p^{out} defined by

$$D_p^{in} = \sum_{q \in N(p)} a_{pq}^{in}, \quad D_p^{out} = \sum_{q \in N(p)} a_{pq}^{out}. \quad (3.9)$$

The numerical experiments for such I²OE scheme are thus already presented in [6] and are not included in the next section.

4. Numerical experiments.

4.1. 1D advection with constant velocity. Let us consider equation (3.1) with $v(x) \equiv 1$ in space interval $\Omega = (-1, 1)$ and time interval $I = (0, T)$, $T = 1$. Let the initial condition u_0 be given by a quadratic polynomial, e.g. by $u_0(x) = 1 - \frac{1}{2}(x^2 - x)$, cf. Figure 4.1. The exact solution is given by shifting in time the initial profile, i.e. $u(x, t) = u_0(x - vt)$. We solve this problem numerically by the I²OE method (3.3) using the exact Dirichlet boundary conditions and compare it with the

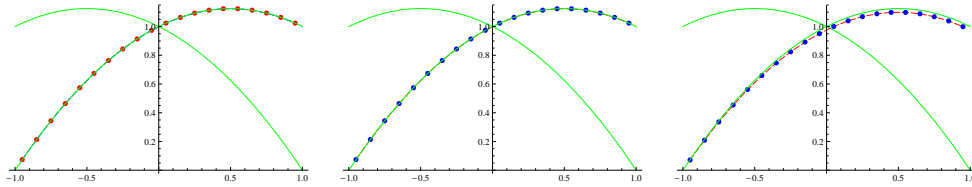


FIG. 4.1. Comparison of the exact and numerical solutions in case of quadratic initial function, I²OE scheme (left), the Lax-Wendroff scheme (middle) and explicit up-wind scheme (right). Here $n = 20, h = 0.1$ and $\tau = h/2$, cf. Tables 4.1 for other cases. The green solid curves represent the initial condition and the exact solution at time $T = 1$, respectively. The red points in the left subfigure represent the exact result of the I²OE scheme, the blue points in the middle subfigure represent also the exact result of the Lax-Wendroff scheme and the blue points and red dashed line in the right subfigure represent approximate result of the explicit up-wind scheme.

exact solution. We also compute the same problem using the standard Lax-Wendroff and explicit up-wind schemes [5] for the constant velocity advection equation. In all comparisons included in this section we used increasing number n of finite volumes discretizing Ω such that the spatial grid size $h = 2/n$. We consider different choices of time steps τ , namely $\tau = h$, $\tau = \frac{h}{2}$, $\tau = 2h$ (or even bigger τ for I²OE scheme) and corresponding number of time steps NTS. In Table 4.1 we report the errors in $L_2(I, L_2)$ norm for all the methods.

As one can see from Table 4.1, the I²OE method is exact for any relation between space and time step, and one can use extremely large (e.g. just one time step $\tau = T$) without any deterioration of the numerical result. Here the errors are comparable to machine precision, they are not exact zeroes because we have to solve a tridiagonal system in every time step yielding some rounding errors which, however, do not propagate even in a long run. The Lax-Wendroff method, as the second order, is exact for any quadratic initial function whenever it is stable. This is of course not the case when $\tau > h$, i.e. for Courant numbers larger than 1. One can see instabilities in the third and 4th rows of Table 4.1, when $\tau = 2h$ and grid is refined. The explicit upwind scheme is the first order, but exact for any initial data if the relation $\tau = h$ is fulfilled, cf. Table 4.1. Its first order accuracy can be seen for $\tau = h/2$ and oscillations occur soon for $\tau > h$ as documented in Table 4.1.

4.2. 1D advection with variable velocity. Let us consider an example with non divergence free velocity field $v(x) = -\sin(x)$ in (3.1) and let the initial profile be given by $u_0(x) = \sin(x)$, $\Omega = (-1, 1)$ and $I = (0, T)$. The exact solution can be derived by the method of characteristics and is given as $u(x, t) = u_0(\frac{2}{\pi} \arctg(e^{\pi t} \tg(\frac{\pi x}{2})))$. We solve this problem numerically for two test cases with $T = 1$ and $T = 2$ and we compare the precision and CPU-time of the I²OE and the Lax-Wendroff schemes. In the first case the non-limited versions of (3.4) and (3.5) are used while in the second case the schemes with the superbee limiter are used.

In the shorter time $T = 1$ the strong peak is already formed but both schemes are stable with slight overshoot and undershoot in the result by the Lax-Wendroff scheme on coarser grids. This is however not observed for the I²OE scheme, even for large time steps several times exceeding standard CFL condition (which is necessary to be respected by the Lax-Wendroff scheme). In Figure 4.2 we plot solutions of two schemes to see the visual comparison of numerical and exact result. The next figure shows a zoom of the solutions in a neighborhood of the origin. Figure 4.4 shows log-log plots of CPU time versus error of the schemes. We can see superior behavior of the

TABLE 4.1

Report on the $L_2(I, L_2)$ errors of the I²OE method, the Lax Wendroff scheme, and the explicit up-wind scheme. for various choices of time step.

n	$\tau = h$	NTS	I ² OE error	Lax Wendroff error	up-wind error
20	0.1	10	$1.8 \cdot 10^{-16}$	0.0	0.0
40	0.05	20	$3.5 \cdot 10^{-16}$	0.0	0.0
80	0.025	40	$7.5 \cdot 10^{-16}$	0.0	0.0
160	0.0125	80	$1.4 \cdot 10^{-15}$	0.0	0.0
n	$\tau = h/2$	NTS	I ² OE	Lax Wendroff	up-wind
20	0.05	20	$3.7 \cdot 10^{-16}$	$5.1 \cdot 10^{-17}$	$1.83 \cdot 10^{-2}$
40	0.025	40	$8.0 \cdot 10^{-16}$	$7.5 \cdot 10^{-17}$	$8.99 \cdot 10^{-3}$
80	0.0125	80	$1.1 \cdot 10^{-15}$	$8.3 \cdot 10^{-17}$	$4.45 \cdot 10^{-3}$
160	0.00625	160	$2.4 \cdot 10^{-15}$	$9.9 \cdot 10^{-17}$	$2.22 \cdot 10^{-3}$
n	$\tau = 2h$	NTS	I ² OE	Lax Wendroff	up-wind
20	0.2	5	$2.1 \cdot 10^{-16}$	$1.1 \cdot 10^{-11}$	$5.02 \cdot 10^{-2}$
40	0.1	10	$2.1 \cdot 10^{-16}$	$1.4 \cdot 10^{-9}$	0.641
80	0.05	20	$3.9 \cdot 10^{-16}$	0.466	$3.8 \cdot 10^{+3}$
160	0.025	40	$5.7 \cdot 10^{-16}$	$1.6 \cdot 10^{+16}$	$1.3 \cdot 10^{+12}$
n	$\tau = 10h$	NTS	I ² OE	Lax Wendroff	up-wind
20	1	1	$4.3 \cdot 10^{-16}$	–	–
40	0.5	2	$1.0 \cdot 10^{-15}$	–	–
80	0.25	4	$1.5 \cdot 10^{-15}$	–	–
160	0.125	8	$2.5 \cdot 10^{-15}$	–	–
160	$\tau = 40h = 0.5$	2	$1.7 \cdot 10^{-15}$	–	–
160	$\tau = 80h = 1$	1	$2.6 \cdot 10^{-15}$	–	–

I²OE scheme in this example with considerable speed-up when using larger time steps up to 4-8 times exceeding the CFL condition. In this case both schemes are second order accurate which holds true for any time step size of the I²OE scheme. Although the Lax-Wendroff scheme gives slightly smaller errors on a given grid when respecting the CFL stability condition ($\tau = h$ in this example), we can gain from higher speed of the I²OE scheme when refining the space discretization and increasing the time step.

In the second experiment, when $T = 2$, the gradient in the exact solution around the origin is very steep and both schemes produce oscillations. They can be limited, e.g. by the super-bee limiter as described in section 3.1. Using the limiter, we can see very precise capturing of the solution profile outside a small neighborhood of zero by both schemes, cf. Figures 4.5-4.6. Figure 4.7 again shows an efficiency/precision comparison of both methods but now when using the limiter. In this case both schemes are first order accurate which holds true for any time step size of the I²OE method. From Figure 4.7 one can also see an interesting behaviour of the I²OE scheme in this example, where no error increase on fixed grid is observed when using time steps up to 16 times exceeding the CFL condition.

The third set of Figures in this subsection represents another possibility for stabilization of the schemes. We constructed grids gradually refined around the origin. The number of finite volumes is the same as in the two previous computations (i.e. $n = 160$) but the size of finite volumes is increasing from 0 to -1 and 1 as a geo-

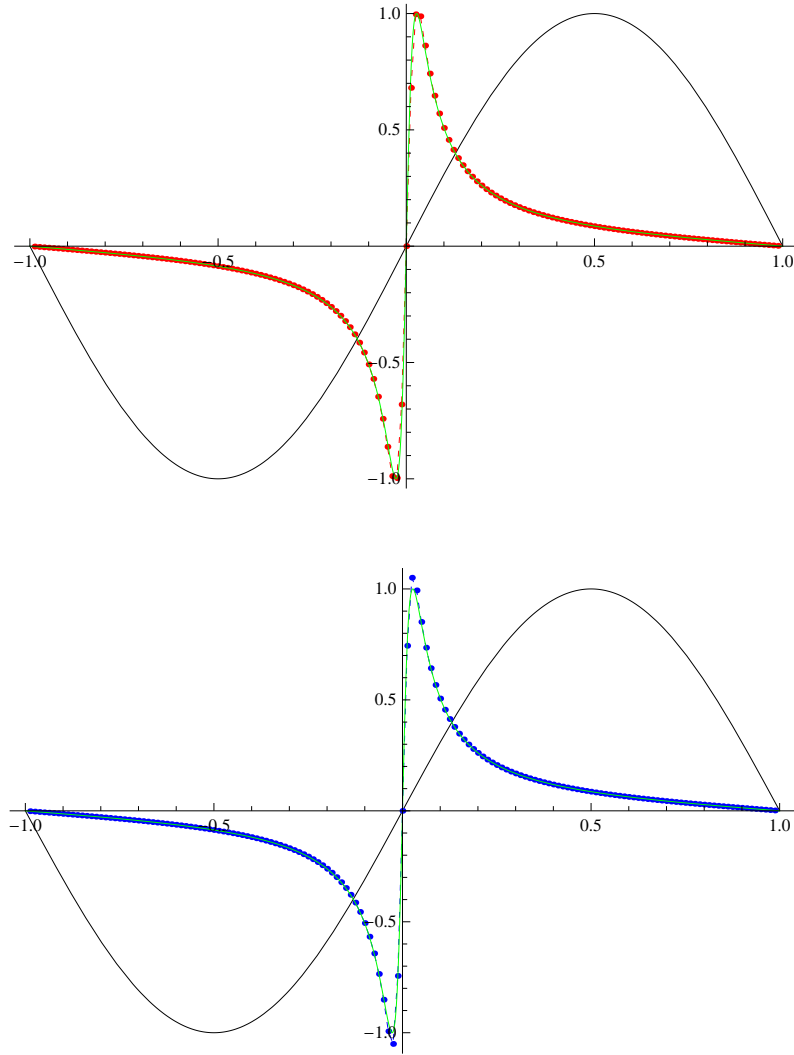


FIG. 4.2. The result of the I²OE scheme (up, red points) and the Lax-Wendroff (LW) method (down, blue points) at time $T = 1$, computed with $n = 160$ and $\tau = h$. By green line we plot the exact solution at T and by black line the initial condition $u_0(x) = \sin(x)$, the velocity in this example is $v(x) = -\sin(x)$.

metrical sequence with quotient 1.07. In Figures 4.8-4.9 we see precise capturing of the solution with shock at time $T = 2$ by both schemes (again with slight overshoot and undershoot in the Lax-Wendroff scheme, similar to the first test case). For the experiments presented in Figures 4.8-4.9 we used the time step $\tau = \tau_{LWCFL}$ fulfilling for the Lax-Wendroff scheme standard CFL stability condition on this graded grid. We note that subsequent uniform refinement of such initially graded grids and halving the time step gives similar plots of CPU time versus error as presented in Figure 4.4. Both schemes are second order accurate and the I²OE method can use large time steps also on this type of grids.

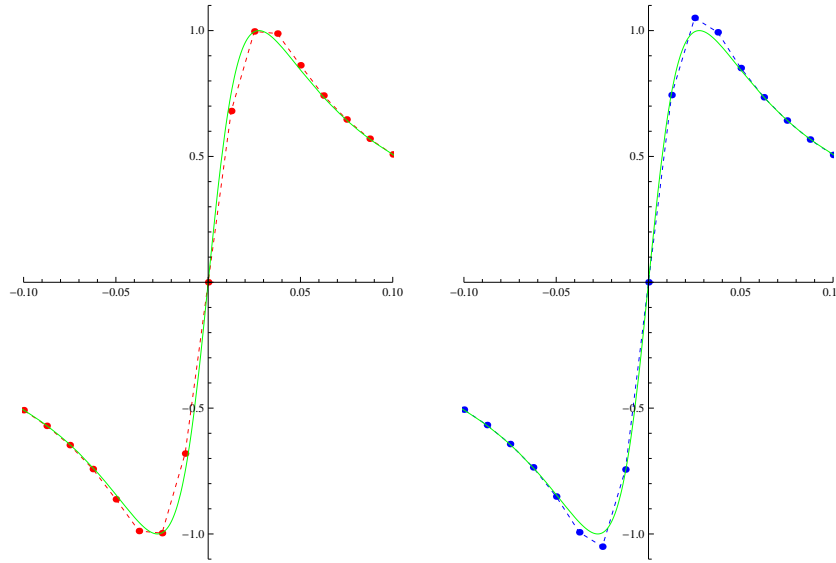


FIG. 4.3. Zooms of the results of the I²OE scheme (left) and LW scheme (right) from Figure 4.2.

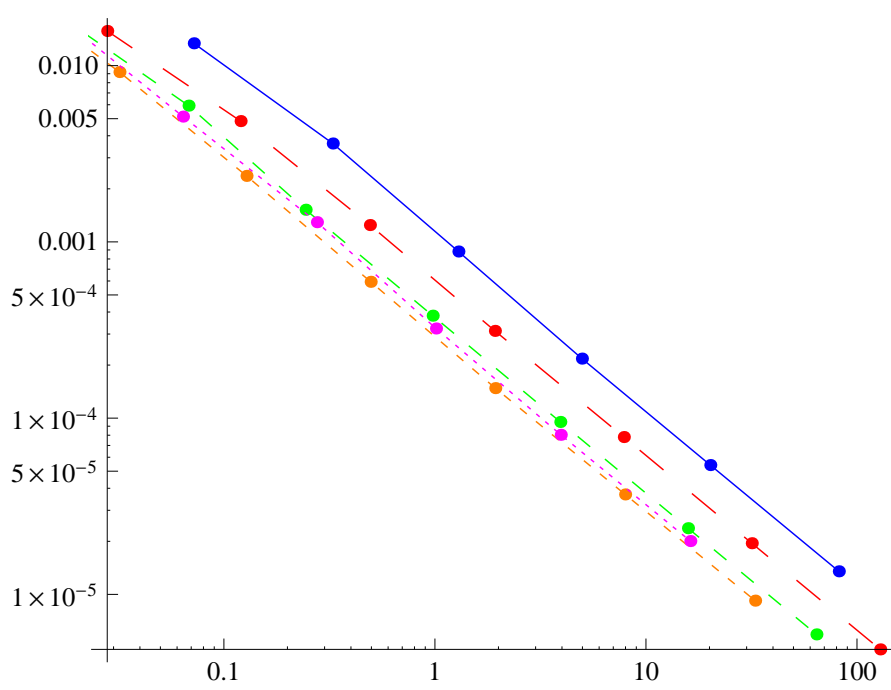


FIG. 4.4. CPU versus $L_2(I, L_2)$ -error for the Lax-Wendroff method (blue solid line) and for the I²OE scheme with CFL=1 (red large dashed, CFL condition is satisfied, i.e. $\tau = h$), CFL=2 (green medium dashed, $\tau = 2h$), CFL=4 (orange small dashed, $\tau = 4h$) and CFL=8 (magenta tiny dashed, $\tau = 8h$) for the experiment without limiter, $T = 1$. The plots indicate that I²OE scheme is about 4 times faster in order to get the same $L_2(I, L_2)$ -error.

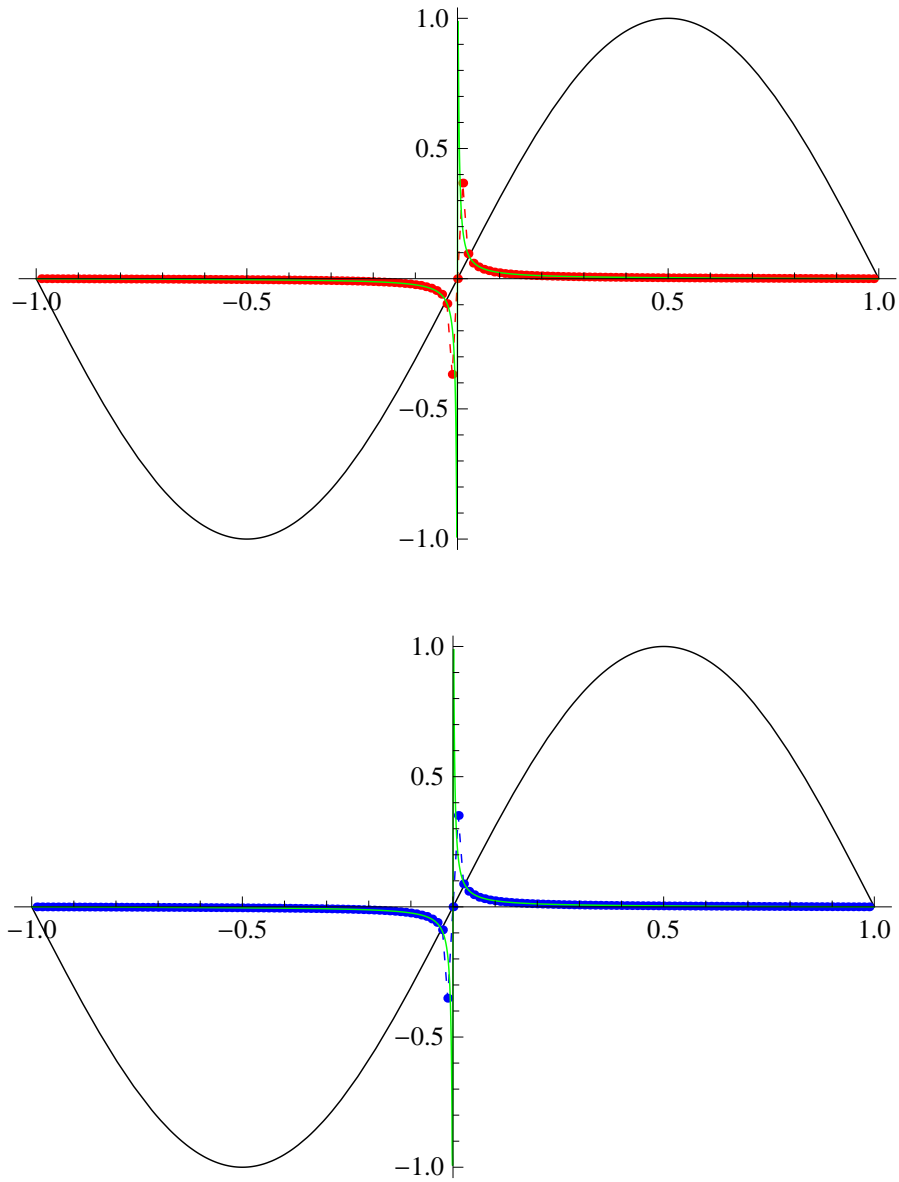


FIG. 4.5. The result of the I^2OE scheme (up, red points) and the Lax-Wendroff method (down, blue points) at time $T = 2$ using limited version of the schemes, computed with $n = 160$ and $\tau = h$ (CFL=1). By green line we plot the exact solution at T and by black line the initial condition $u_0(x) = \sin(x)$, the velocity $v(x) = -\sin(x)$.

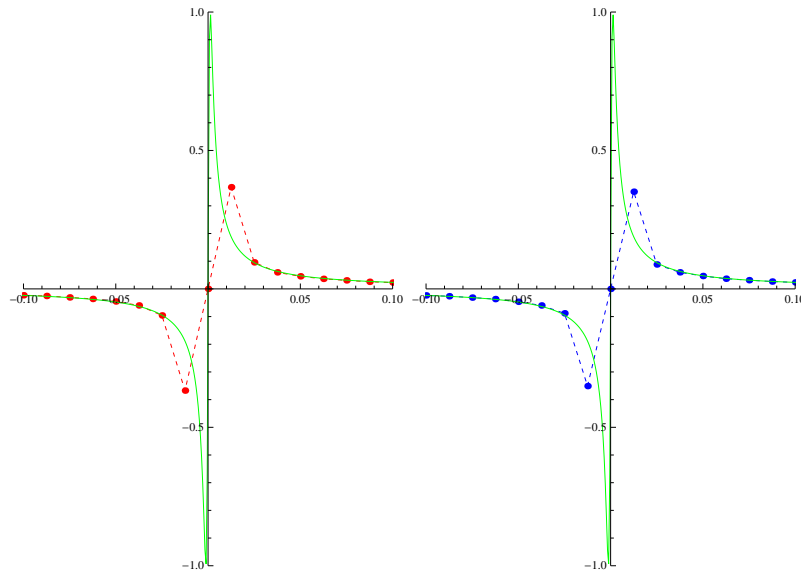


FIG. 4.6. Zooms of the results of the I²OE scheme (left) and LW scheme (right) from Figure 4.5.

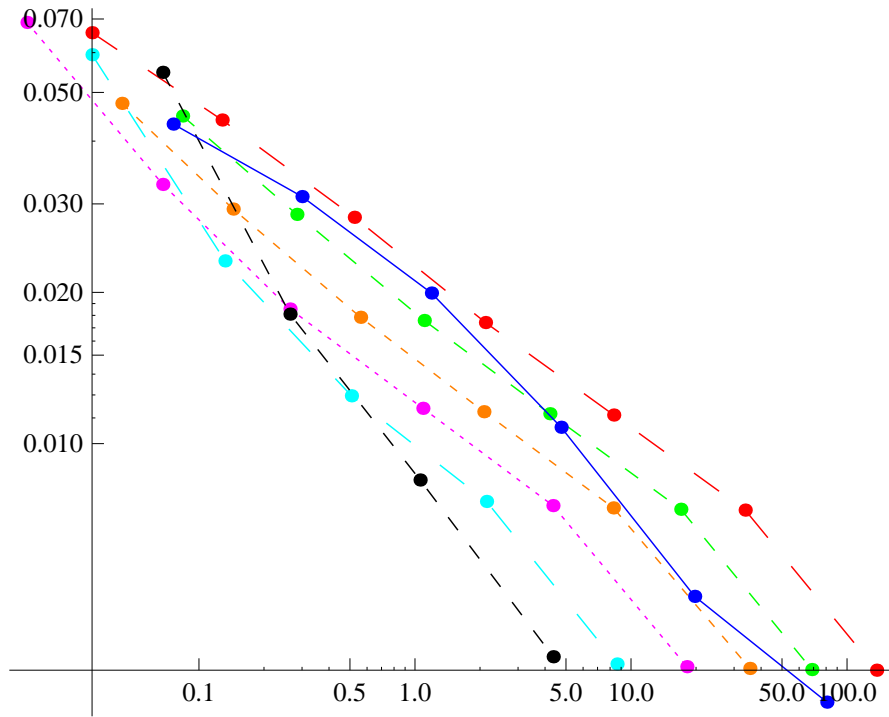


FIG. 4.7. CPU versus $L_2(I, L_2)$ -error for the Lax-Wendroff method (blue) and for the I²OE scheme with CFL=1 (red large dashed, CFL condition is satisfied, i.e. $\tau = h$), CFL=2 (green medium dashed, $\tau = 2h$), CFL=4 (orange small dashed, $\tau = 4h$) and CFL=8 (magenta tiny dashed, $\tau = 8h$), CFL=16 (cyan large dashed, $\tau = 16h$) and CFL=32 (black medium dashed, $\tau = 32h$) for the experiment with limiter, $T = 2$.

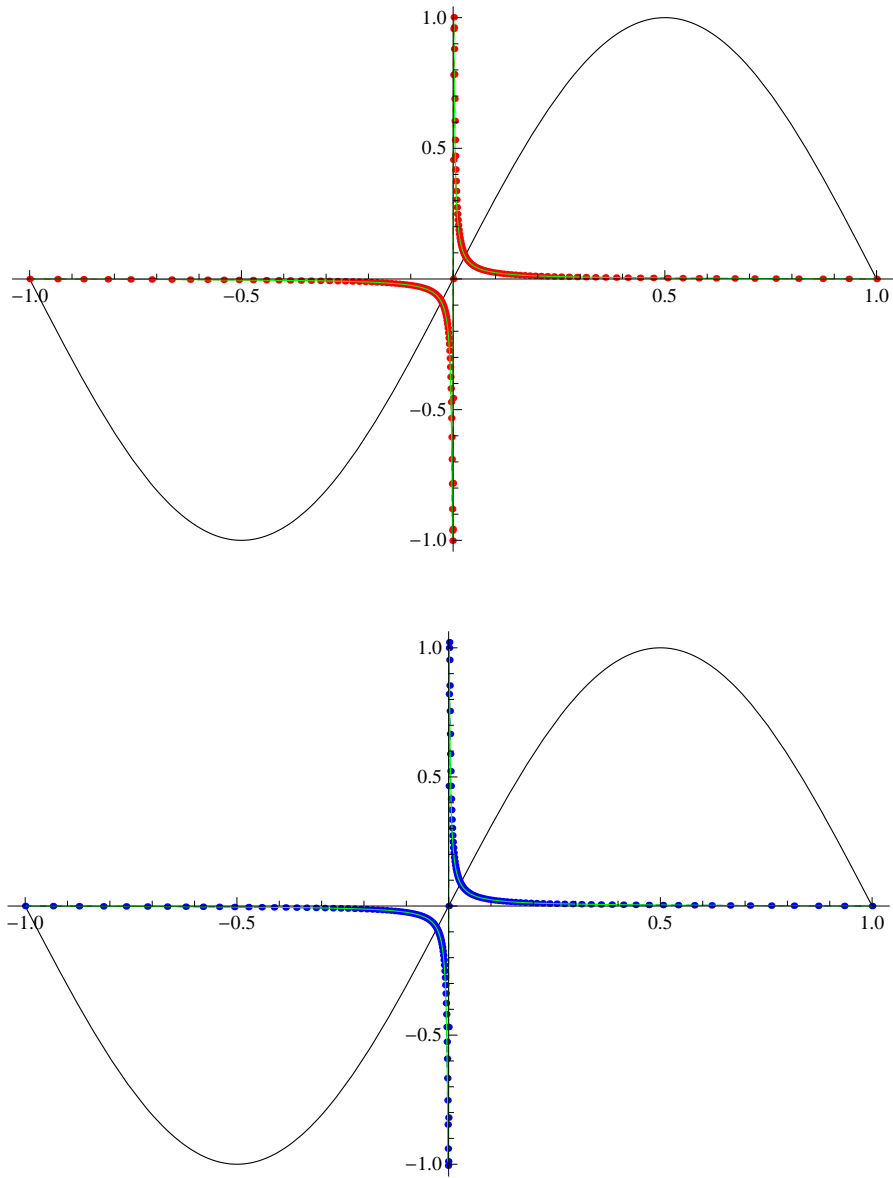


FIG. 4.8. The result of the I^2OE scheme (up, red points) and the Lax-Wendroff method (down, blue points) at time $T = 2$, computed on a grid gradually refined around the origin ($n = 160$, $\tau = \tau_{CFL}$) and without limiter.

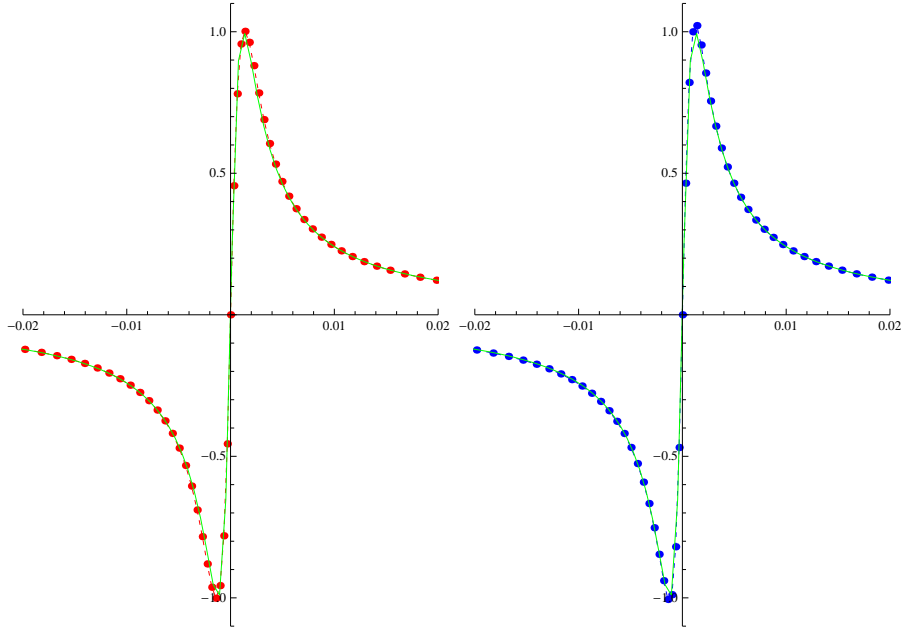


FIG. 4.9. Zooms of the results from Figure 4.5, when using the gradually refined grid around the origin and without limiter, for the I²OE scheme (left) and LW scheme (right).

4.3. 1D viscous Burgers equation. In order to test the scheme in the case of non-linear advection we have chosen the exact solution

$$u(x, t) = u_r + \frac{1}{2}(u_l - u_r) \left(1 - \tanh \left(\frac{(u_l - u_r)(x - st)}{4\sigma} \right) \right), \quad (4.1)$$

$u_l > u_r$, of the viscous Burgers equation (3.6).

We solve the problem (3.6) by the scheme (3.7) on space interval $(-0.5, 0.5)$ and in time interval $(0, 0.48)$, first with $\sigma = 0.01$. Naturally, in order to keep stability of the scheme, the spatial discretization step h should be proportional to σ . Exploring the explicit part of the scheme one can find that when $\tau = 4h^2/(2\sigma - h)$ we get no oscillations. Here it means that for $h = 0.01$ ($n = 100$) we can use time a step $\tau = 0.04$. Then one can refine the time and space step in order to check that the scheme is second order accurate, cf. Table 4.2. The visual comparisons of numerical and exact results for $n = 100$ and $n = 200$ are presented in Figure 4.10. Interestingly, $\tau = 0.04$ represents a stability constraint for the I²OE scheme in this example for any spatial grid size, we document that phenomenon by Figure 4.11 where such a time step was used on the grid with $n = 800$. These considerations may also lead to a suitable stabilization of the I²OE scheme by an artificial viscosity in case of non-viscous Burgers equation, which will be an objective of our further research.

TABLE 4.2

Report on the $L_2(I, L_2)$ errors of Γ^2 OE method for the viscous Burgers equation (3.6) with $\sigma = 0.01$.

n	h	τ	NTS	$L_2(I, L_2)$	EOC
100	0.01	0.04	12	$5.01 \cdot 10^{-3}$	
200	0.005	0.02	24	$1.22 \cdot 10^{-3}$	2.04
400	0.0025	0.01	48	$3.01 \cdot 10^{-4}$	2.02
800	0.00125	0.005	96	$7.51 \cdot 10^{-5}$	2.00

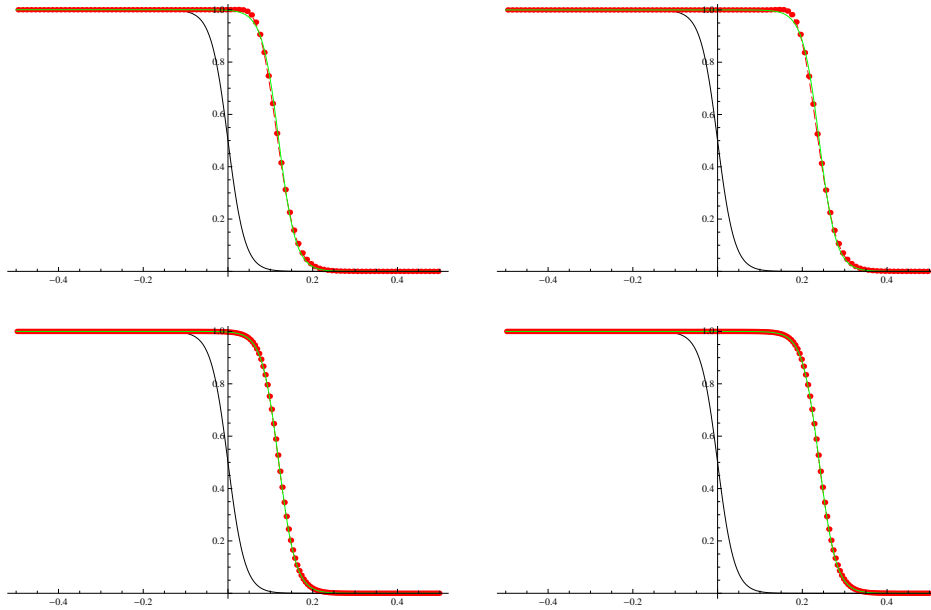


FIG. 4.10. The results of the Γ^2 OE scheme in solving viscous Burgers equation (3.6) with $\sigma = 0.01$. Comparison with travelling wave exact solution (4.1) in time $t = 0.24$ (left) and $t = 0.48$ (right) for $n = 100$ (up) and $n = 200$ (down), $\tau = 4h$.

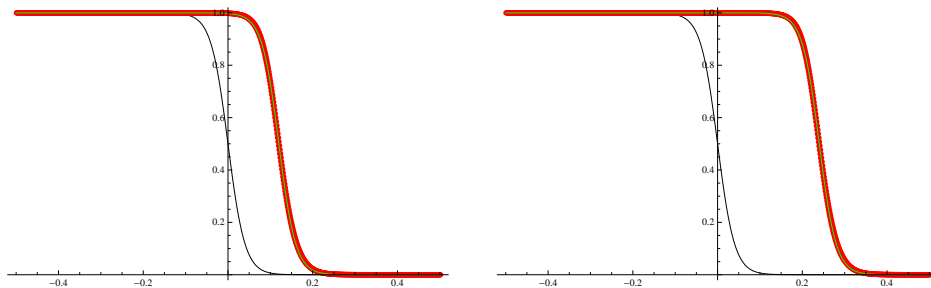


FIG. 4.11. The results of the Γ^2 OE scheme in solving viscous Burgers equation (3.6) with $\sigma = 0.01$. Comparison with travelling wave exact solution (4.1) in time $t = 0.24$ (left) and $t = 0.48$ (right) for $n = 800$ and $\tau = 0.04 = 32h$. There is no losing of stability in the case of such large time step.

TABLE 4.3

Report on the $L_2(I, L_2)$ errors of I²OE method for the viscous Burgers equation (3.6) with $\sigma = 0.001$.

n	h	τ	NTS	$L_2(I, L_2)$	EOC
250	0.004	0.016	30	$2.04 \cdot 10^{-2}$	
500	0.002	0.008	60	$6.97 \cdot 10^{-3}$	1.57
1000	0.001	0.004	120	$1.78 \cdot 10^{-3}$	1.97
2000	0.0005	0.002	240	$4.43 \cdot 10^{-4}$	2.00
4000	0.00025	0.001	480	$1.105 \cdot 10^{-4}$	2.00

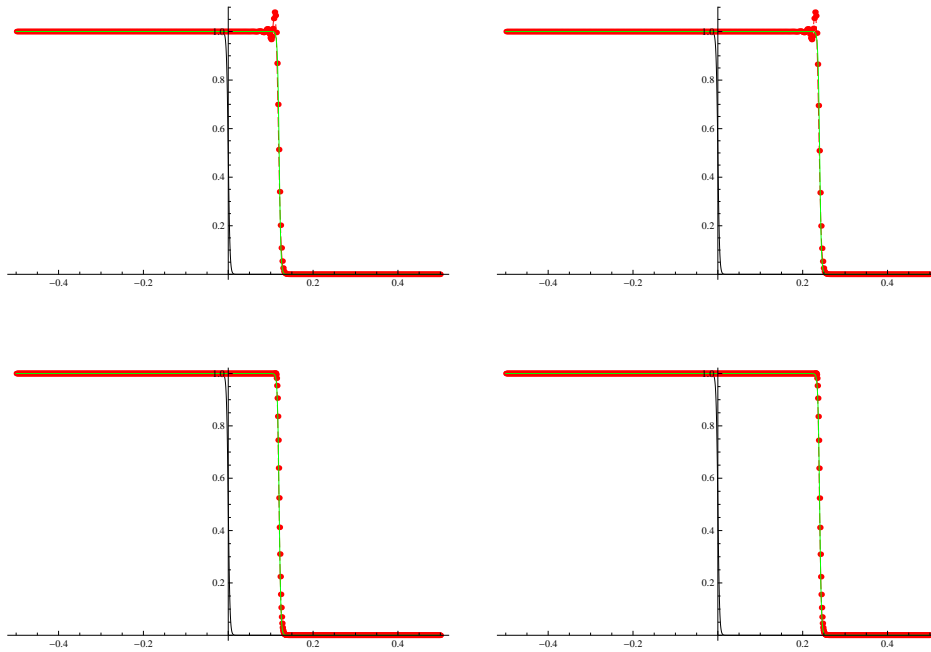


FIG. 4.12. The results of the I²OE scheme in solving viscous Burgers equation (3.6) with $\sigma = 0.001$. Comparison with travelling wave exact solution (4.1) in time $t = 0.24$ (left) and $t = 0.48$ (right) for $n = 500$ (up, small nonincreasingly propagating oscillations) and $n = 1000$ (down, results without any oscillation), $\tau = 4h$.

In the second example we decreased the viscosity ten times and use $\sigma = 0.001$. A similar consideration as above leads to a stable choice of the time step $\tau = 0.004$ for the grid size $h = 0.001$ ($n = 1000$). In Table 4.3 we show errors and EOC for refined and coarsened grids and time step. One can again see that EOC=2 also in this example when refining the grid. The lower convergence rate in the beginning is caused by oscillations when the grid size is not sufficiently fine, but as we can see from Figure 4.12 (top) these oscillations are "stable", they do not increase in time and by refining the spatial resolution they are removed completely. This indicates that the local grid refinement, as presented in subsection 3.1 in Figures 4.8-4.9, would be useful also here provided that the finer mesh is moving together with the shock-like profile.

4.4. 2D advection by a constant velocity vector field and by a rotation.

Let us now consider a second order radially symmetric polynomial initial function in

TABLE 4.4

Report on the $L_2(I, L_2)$ errors of I²OE method for various choices of time step in case of a paraboloid centered and rotated around the origin (4th column) and a general quadratic function transported by a constant vector field (5th column).

n	τ	NTS	$L_2(I, L_2)$	$L_2(I, L_2)$
20	0.1	10	$1.9 \cdot 10^{-16}$	$1.4 \cdot 10^{-15}$
40	0.05	20	$3.7 \cdot 10^{-16}$	$7.1 \cdot 10^{-15}$
80	0.025	40	$6.2 \cdot 10^{-16}$	$1.1 \cdot 10^{-15}$
160	0.0125	80	$8.6 \cdot 10^{-16}$	$1.3 \cdot 10^{-14}$
160	0.00625	160	$1.0 \cdot 10^{-15}$	$1.6 \cdot 10^{-14}$
160	0.05	20	$5.7 \cdot 10^{-16}$	$3.4 \cdot 10^{-14}$
160	1.0	1	$3.2 \cdot 10^{-16}$	$2.2 \cdot 10^{-14}$

TABLE 4.5

Report on the $L_2(I, L_2)$ errors of I²OE method for a rotating paraboloid centered outside the origin.

n	τ	NTS	$L_2(I, L_2)$	EOC
20	0.157	20	$7.585 \cdot 10^{-3}$	
40	0.0785	40	$1.772 \cdot 10^{-3}$	2.09
80	0.03925	80	$4.247 \cdot 10^{-4}$	2.06
160	0.019625	160	$1.035 \cdot 10^{-4}$	2.04

the form of paraboloid $u_0(x_1, x_2) = x_1^2 + x_2^2 - 0.25$ and the vector field $\mathbf{v}(x_1, x_2) = (-x_2, x_1)$ which rotates the initial function around the origin. Since it is radially symmetric the exact solution does not change in time. This test problem as well as all further 2D examples in this section are solved in the spatial domain $\Omega = (-1, 1) \times (-1, 1)$ which is split into $n \times n$ finite volumes. The time interval $(0, T)$ is equal $(0, 1)$ for this example and one can see in the 4th column of Table 4.4 that the exact solution is reproduced numerically up to machine precision for any grid size and time step size. The system matrix and the right hand side of the I²OE method is constructed in such way that it does not touch this exact solution.

In the second example we consider the same paraboloid but shifted to $(0.5, 0)$, i.e. $u_0(x_1, x_2) = (x_1 - 0.5)^2 + x_2^2 - 0.25$ and the same rotational vector field. The problem is solved in time interval $(0, 3.14)$ and the method is second order accurate, as one can see in Table 4.5, where no special attention to a CFL condition was given when choosing τ and h .

The third example represents transport of the quadratic polynomial $u_0(x_1, x_2) = 2x_1^2 + x_2^2 - x_1 + x_2 - 0.25$ by the constant vector field $\mathbf{v}(x_1, x_2) = (-1.0, 0.5)$. We can see in the 5th column of Table 4.4 that the numerical solution by the I²OE scheme is exact for any length of time and space step, the errors are again comparable with machine precision.

4.5. 2D advection with nontrivial variable velocity. In the next two examples we test the I²OE scheme choosing the velocity vector field

$$\mathbf{v}(x) = -\frac{x}{|x|}, x = (x_1, x_2).$$

The exact solution is given by $u(x, t) = u_0(x + \frac{tx}{|x|})$ where $u_0(x)$ is an initial profile.

First we consider the initial function $u_0(x_1, x_2) = -x_1^2 - x_2^2 + 1$ and solve the problem in time interval $(0, 0.6)$ on subsequently refined grids. In this case there is

TABLE 4.6
 Report on the I²OE scheme errors for the variable velocity example (4.2).

n	τ	NTS	$L_1(I, L_1)$	EOC	$L_1(I, L_1)$	EOC	NTS	$L_1(I, L_1)$	EOC
20	0.1	6	$6.57 \cdot 10^{-3}$		$1.05 \cdot 10^{-2}$		25	$1.95 \cdot 10^{-1}$	
40	0.05	12	$1.48 \cdot 10^{-3}$	2.15	$2.41 \cdot 10^{-3}$	2.12	50	$8.01 \cdot 10^{-2}$	1.28
80	0.025	24	$3.50 \cdot 10^{-4}$	2.08	$6.17 \cdot 10^{-4}$	1.96	100	$3.73 \cdot 10^{-2}$	1.10
160	0.0125	48	$8.52 \cdot 10^{-5}$	2.03	$1.76 \cdot 10^{-4}$	1.80	200	$1.83 \cdot 10^{-2}$	1.02
320	0.00625	96	$2.10 \cdot 10^{-5}$	2.02	$5.79 \cdot 10^{-5}$	1.60	400	$9.14 \cdot 10^{-3}$	1.00

a point singularity formed in the origin due to subsequent arrival of initial function values from circular neighborhoods. The numerical method is second order accurate in this case as one can see in the 4th column of Table 4.6.

More complicated and interesting behaviour is observable when $u_0(x_1, x_2) = -2x_1^2 - x_2^2 + 1$, when different values are coming to the origin at the same time from different radial characteristics directions. It causes "hole"-like singularities at the origin. In this case we can (naturally) see a lower than second order accuracy in a short run time interval $(0, 0.6)$, the 6th column of Table 4.6, and the first order precision for long run time interval $(0, 2.5)$, in the 9th column of Table 4.6. In these cases, due to a strongly singular behavior we measure the errors in the $L_1(I, L_1)$ norm. The numerical result at time $T = 2.5$ for the grid $n = 80$ is shown in Figure 4.13 left up in the form of graph and right up in the form of isolines. Comparisons of particular isoline $u(x, T) = -10$ of numerical and exact solutions are shown in the bottom part of the same Figure for various choices of the time step τ . On the bottom left, the isolines of numerical solution (red line) and exact solution (blue line) are plotted for $\tau = 0.025$. Choosing 10 times bigger time step $\tau = 0.25$ we do not see visually (bottom middle) any error increase and the CPU was 4 times shorter (in fact, the error was $4.00 \cdot 10^{-2}$ in comparison with $3.73 \cdot 10^{-2}$ obtained with $\tau = 0.025$, cf. Table 4.6). Interestingly, we can use an extremely large (just one) time step $\tau = 2.5$ without any additional stabilization and obtain the result shown on the bottom right which visually only slightly differs from the exact solution and was obtained in 20 times faster CPU.

5. Conclusions. In this article we introduced a new scheme for solving variable velocity advection equations. By our best knowledge the inflow-implicit/outflow-explicit splitting is a new original approach and we have shown that it brings interesting results. There is only a weak stability constraint when solving general (non-linear) advection and advection-diffusion equations. The method is second order accurate and can be used in any spatial dimension and on unstructured Voronoi-like grids. We presented the scheme in solving 1D and 2D advection equations with divergence and non-divergence free velocities and in solving the viscous Burgers equation. A solution of the level-set problems for motion in normal direction is also a special case of the scheme and was presented in [6].

Together with its interesting behavior, this new semi-implicit approach brings new open questions in solving advection equations which are worth to be studied in the near future. The goal of this paper was to introduce the inflow-implicit/outflow-explicit splitting idea and to outline its basic properties and first computational results.

6. Acknowledgment. The work of the first author was supported by the grants APVV-0351-07 and VEGA 1/0269/09.

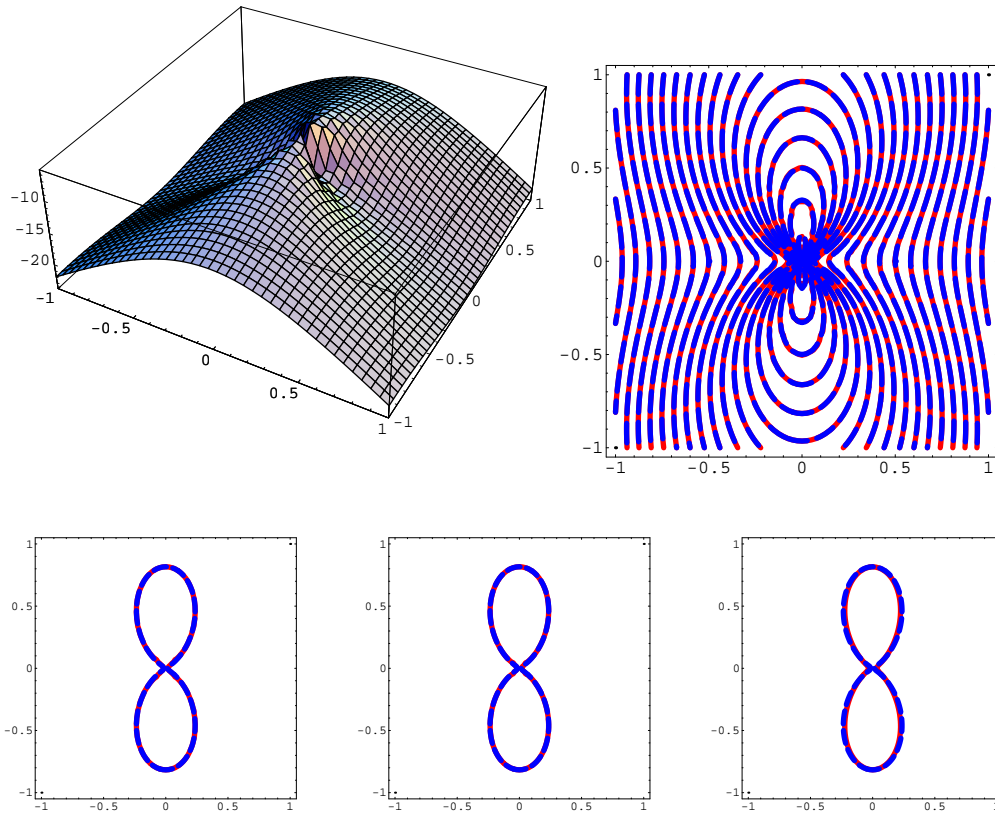


FIG. 4.13. The graph of numerical solution at $T = 2.5$ for experiment with nontrivial velocity (4.2), left up. The comparison of isolines of the numerical solution (red solid lines) and exact solution (blue dashed lines), right up. In the bottom row we present comparison of particular exact and numerical isoline of the result computed with time step τ equals to 0.025 (left), 0.25 (middle) and 2.5 (right).

REFERENCES

- [1] P.Bourgine, R.Čunderlík, O.Drbílková, K.Mikula, N.Peyriéras, M.Remešíková, B.Rizzi, A.Sarti, 4D embryogenesis image analysis using PDE methods of image processing, *Kybernetika*, Vol. 46, No. 2 (2010).
- [2] Eymard, R., Gallouet, T., & Herbin R.: The finite volume methods, *Handbook for Numerical Analysis*, 2000, Vol. 7 (Ph. Ciarlet, J. L. Lions, eds.), Elsevier.
- [3] Frolkovič, P., Mikula, K.: Flux-based level set method: a finite volume method for evolving interfaces, *Applied Numerical Mathematics*, Vol. 57, No. 4 (2007) pp. 436-454.
- [4] Frolkovič, P., Mikula, K.: High-resolution flux-based level set method, *SIAM Journal on Scientific Computing*, Vol. 29, No. 2 (2007) pp. 579-597.
- [5] LeVeque R.J., *Finite Volume Methods for Hyperbolic Problems*, Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002.
- [6] Mikula, K., Ohlberger, M.: A new level set method for motion in normal direction based on a semi-implicit forward-backward diffusion approach, *SIAM J. Scientific Computing*, Vol. 32, No. 3 (2010) pp. 1527-1544.
- [7] Sethian, J. A.: *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Material Science*, Cambridge University Press, New York, 1999.

Preprints
"Angewandte Mathematik und Informatik"

- 01/08 - N P. Henning, M. Ohlberger: The heterogeneous multiscale finite element method for elliptic homogenization problems in perforated domains
- 02/08 - S G. Alsmeyer, A. Iksanov: A log-type moment result for perpetuities and its application to martingales in supercritical branching random walks
- 03/08 - S G. Alsmeyer, M. Meiners: On a Min-Type Stochastic Fixed-Point Equation Related to the Smoothing Transformation
- 04/08 - S G. Alsmeyer, M. Meiners: A Note on the Transience of Critical Branching Random Walks on the Line
- 05/08 - S G. Alsmeyer, G. Hölker: Asymptotic Behavior of Ultimately Contractive Iterated Lipschitz Functions
- 06/08 - N E. Pekalska, B. Haasdonk: Kernel Quadratic Discriminant Analysis with Positive Definite and Indefinite Kernels
- 07/08 - S M. Meiners: Weighted Branching and a Pathwise Renewal Equation
- 08/08 - S M. Ebbers, M. Löwe: Torpid Mixing of the Swapping Chain on Some Simple Spin Glass Models
- 09/08 - I T. Ropinski, I. Viola, M. Biermann, F. Lindemann, R. Leißa, H. Hauser, K. Hinrichs: Multimodal Closeups for Medical Visualization
- 01/09 - I J. Mensmann, T. Ropinski, K. Hinrichs: An Evaluation of the CUDA Architecture for Volume Rendering
- 02/09 - N P. Henning, M. Ohlberger: Advection-diffusion problems with rapidly oscillating coefficients and large expected drift. Part 1: Homogenization – existence, uniqueness and regularity
- 03/09 - N P. Henning, M. Ohlberger: Advection-diffusion problems with rapidly oscillating coefficients and large expected drift. Part 2: The heterogeneous multiscale finite element method
- 04/09 - I J. Meyer-Spradow, T. Ropinski, J. Mensmann, K. Hinrichs: Rapid Prototyping of Volume Visualization in Collaboration with Domain Experts
- 05/09 - N K. Mikula, M. Ohlberger: A New Level Set Method for Motion in Normal Direction Based on a Forward-Backward Diffusion Formulation
- 06/09 - I T. Ropinski, S. Diepenbrock, S. Bruckner, K. Hinrichs, E. Gröller: Volumetric Texturing
- 07/09 - I J.-S. Praßni, J. Mensmann, T. Ropinski, K. Hinrichs: Shape-based Transfer Functions for Volume Visualization
- 08/09 - N A. Dedner, R. Klöfkorn, M. Nolte, M. Ohlberger: A generic interface for parallel and adaptive scientific computing: Abstraction principles and the DUNE-FEM module
- 09/09 - N P. Henning, M. Ohlberger: A-posteriori error estimate for a heterogeneous multiscale finite element method for advection-diffusion problems with rapidly oscillating coefficients and large expected drift
- 01/10 - N K. Mikula, M. Ohlberger: A New Inflow-Implicit/Outflow-Explicit Finite Volume Method for Solving Variable Velocity Advection Equations