Weierstraß-Institut
für Angewandte Analysis und Stochastik
Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 0946 – 8633

On (essentially) non-oscillatory discretizations of evolutionary convection-diffusion equations

Volker John[1], Julia Novo[2]

submitted: October 26, 2011

[1] Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
email: volker.john@wias-berlin.de

[2] Universidad Autónoma de Madrid
Departamento de Matemáticas
CSIC-UAM-UC3M-UCM
Spain
email: julia.novo@uam.es

No. 1656
Berlin 2011

2010 *Mathematics Subject Classification.* 65M06, 65M60.

Key words and phrases. Time-dependent convection-diffusion-reaction equations, under- and overshoots, FEM-FCT schemes, ENO schemes, WENO schemes.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract

Finite element and finite difference discretizations for evolutionary convection-diffusion-reaction equations in two and three dimensions are studied which give solutions without or with small under- and overshoots. The studied methods include a linear and a non-linear FEM-FCT scheme, simple upwinding, an ENO scheme of order 3, and a fifth order WENO scheme. Both finite element methods are combined with the Crank–Nicolson scheme and the finite difference discretizations are coupled with explicit total variation diminishing Runge–Kutta methods. An assessment of the methods with respect to accuracy, size of under- and overshoots, and efficiency is presented, in the situation of a domain which is a tensor product of intervals and of uniform grids in time and space. Some comments to the aspects of adaptivity and more complicated domains are given. The obtained results lead to recommendations concerning the use of the methods.

1 Introduction

Evolutionary convection-diffusion-reaction equations are contained in many models of processes from applications, since these equations describe, e.g., chemical reactions or the conservation of concentrations or energy (temperature). Often, the convection is the main mechanism in these processes. In general, the equations have to be solved numerically, which poses difficulties in the convection-dominated regime.

This paper studies numerical methods for linear time-dependent convection-diffusion-reaction equations

$$\begin{aligned} u_t - \varepsilon\Delta u + \mathbf{b}\cdot\nabla u + cu &= f \qquad \text{in } (0,T]\times\Omega, \\ u &= 0 \qquad \text{on } [0,T]\times\partial\Omega, \\ u(0,\mathbf{x}) &= u_0(\mathbf{x}) \text{ in } \Omega. \end{aligned} \tag{1}$$

In (1), $\Omega\subset\mathbb{R}^d$, $d\in\{1,2,3\}$, is a bounded domain, $[0,T]$ is a time interval, $\varepsilon>0$ denotes the diffusion parameter, $\mathbf{b}(t,\mathbf{x})$ is the convection field, $c(t,\mathbf{x})$ describes the reaction, $f(t,\mathbf{x})$ models sources and sinks, and $u_0(\mathbf{x})$ is the initial condition. The use of homogeneous Dirichlet boundary conditions in (1) is just for simplicity of presentation. The numerical studies will consider also other boundary conditions.

Solutions of (1) possess in general a distinct feature, namely layers. In these layers, the values of the solution change rapidly in a very small subdomain, which is too small to be resolved by the underlying grids. This causes the difficulties in the numerical simulations. So-called stabilized methods have to be used. But even with stabilization, under- and overshoots in a vicinity of the layers might occur or the layers might be smeared to a width which is larger by magnitudes than in reality. Both features are unwelcome in simulations of applications, but in particular the under- and overshoots might be perilous since they describe non-physical situations like negative concentrations. Unphysical values might become even the reason for instabilities in simulations of strongly coupled systems, if solutions obtained with such values serve as input to other equations in the model and the unphysical values lead to incorrect model parameters [15]. There has been a lot of research to construct methods that give numerical solutions with sharp layers and without under- and overshoots [27]. In this paper, two classes of such methods will be compared numerically. Only such classes will be considered, where the numerical solutions do not possess under- or overshoots or where they are at least small.

A possible approach for discretizing (1) consists in using stabilized finite element methods in space which are combined with simple implicit time stepping schemes. Several papers appeared in the last years which studied this approach by providing error estimates [6,14] or by comparing them numerically with respect to accuracy [16,17]. In the latter studies, it was found that the only methods of this approach which provide solutions without large under- and overshoots are Finite Element Flux Corrected Transport Methods (FEM-FCT) from [21,20,19]. For this reason, only such finite element schemes will be considered here. One of the schemes is linear and the other one is nonlinear. This raises another issue besides accuracy, namely the efficiency of the schemes. Both issues will be included in the assessment of the methods. Two ways of obtaining the system matrices will be studied: assembling in each discrete time and the so-called group finite element approach [8].

If Ω is a simple domain, like a tensor product of intervals, the use of finite difference schemes for the discretization in space is an attractive option. The simplest approach for dealing with convection-dominated problems is the application of simple upwinding [22]. However, it is well known that with this method in general rather smeared solutions are obtained. In the last decades, essentially non-oscillatory (ENO) and weighted ENO (WENO) schemes have been developed in the context of hyperbolic partial differential equations. These schemes use a wider stencil than the simple upwinding. ENO schemes were proposed the first time in [11]. The ENO strategy consists in computing several finite difference approximations of the convective term and to choose the smoothest of these approximations. With this approach, solutions of hyperbolic partial differential equations with high order accuracy in smooth regions and with sharp shocks are obtained. In the WENO methodology, which was proposed the first time in [23] and reviewed recently in [29], a convex combination of several finite difference approximations is used to approximate the convective term. In this way, higher order methods can be constructed on the same stencils as the ENO schemes. The WENO methodology was combined already with discontinuous finite elements in [25]. All finite difference approaches (simple upwinding, ENO, WENO) will be included in the numerical studies. They will be combined with total variation diminishing (TVD) explicit Runge–Kutta schemes as temporal discretization. To our best knowledge, comprehensive numerical studies of ENO and WENO schemes for convection-diffusion-reaction equations in multiple dimensions are not yet available in the literature.

It might be already expected at this point that the explicit schemes will be faster than the implicit methods. Also, that the more complicated nonlinear FEM-FCT scheme will be in general more accurate than the linear methods. However, to our best knowledge, there is no quantification of the gains and losses of the individual schemes compared with the other schemes available in the literature. The main goal of this paper consists in providing such a quantification. It will be concentrated on the simplest situation, i.e., the domain Ω is a tensor product of intervals and uniform grids in time and space are used.

There are applications where this situation occurs, even with $d > 3$, like in the simulation of population balance systems [15,3]. The application of the methods to problems defined on more complicated domains and adaptivity in time and space will be also discussed to some extent.

The paper is organized as follows. Section 2 introduces the FEM-FCT schemes and Section 3 the finite difference methods. Numerical studies are presented in Section 4. A discussion on adaptivity and more complicated domains follows in Section 5. The conclusions of the numerical studies are summarized in Section 6.

2 The FEM-FCT Schemes

The used FEM-FCT schemes are basically the same as in [16,17]. A detailed description of the schemes can be found already in [16]. To keep this paper self-containing, a short presentation of the schemes, which provides the basic ideas, will be given here. In addition to [16,17], prelimiting was applied and the group finite element version of the schemes was used.

Consider continuous piecewise linear or piecewise d-linear finite elements. Applying the Crank–Nicolson scheme and the standard Galerkin finite element method for the discretization of (1) gives at time t_n an algebraic equation of the form

$$\left(M_C + \frac{\Delta t_n}{2} A\right) \mathbf{u}_n = \left(M_C - \frac{\Delta t_n}{2} A\right) \mathbf{u}_{n-1} + \frac{\Delta t_n}{2} \mathbf{f}_{n-1} + \frac{\Delta t_n}{2} \mathbf{f}_n, \qquad (2)$$

where $(M_C)_{ij} = (\varphi_j, \varphi_i)$ is the consistent mass matrix, the matrix A is the stiffness matrix containing the sum of diffusion, convection, and reaction, and Δt_n is the current length of the time step. The vectors of the coefficients of the finite element functions are denoted by $\mathbf{u}_n, \mathbf{f}_n$ etc. and the length of these vectors (number of degrees of freedom) is denoted by N.

At the beginning, (2) is modified such that the matrix on the left hand side gets properties of an M-matrix. To this end, define

$$\begin{aligned} L &= A + D, \\ D &= (d_{ij}), \quad d_{ij} = -\max\{0, a_{ij}, a_{ji}\} = \min\{0, -a_{ij}, -a_{ji}\} \text{ for } i \neq j, \\ &\qquad\qquad d_{ii} = -\sum_{j=1, j\neq i}^{N} d_{ij}, \end{aligned} \qquad (3)$$

$$M_L = \operatorname{diag}(m_i), \quad m_i = \sum_{j=1}^{N} m_{ij}. \qquad (4)$$

The diagonal matrix M_L is the lumped mass matrix. Instead of (2), now the equation

$$\left(M_L + \frac{\Delta t_n}{2} L\right) \mathbf{u}_n = \left(M_L - \frac{\Delta t_n}{2} L\right) \mathbf{u}_{n-1} + \frac{\Delta t_n}{2} \mathbf{f}_{n-1} + \frac{\Delta t_n}{2} \mathbf{f}_n \tag{5}$$

is considered. This is the algebraic representation of a stable low order scheme, whose solution does not possess under- and overshoots but widely smeared layers. In the next step, the smearing of the layers will be reduced by a modification of the right hand side of (5)

$$\left(M_L + \frac{\Delta t_n}{2} L\right) \mathbf{u}_n = \left(M_L - \frac{\Delta t_n}{2} L\right) \mathbf{u}_{n-1} + \frac{\Delta t_n}{2} \mathbf{f}_{n-1} + \frac{\Delta t_n}{2} \mathbf{f}_n + \mathbf{f}^* \left(\mathbf{u}_n, \mathbf{u}_{n-1}\right).$$

An appropriate ansatz for $\mathbf{f}^*(\mathbf{u}_n, \mathbf{u}_{n-1})$ is needed. To this end, consider the residual vector defined by the difference of (5) and (2)

$$\begin{aligned}
\mathbf{r} &= \left(M_L + \frac{\Delta t_n}{2} L - \left(M_C + \frac{\Delta t_n}{2} A\right)\right) \mathbf{u}_n \\
&\quad - \left(M_L - \frac{\Delta t_n}{2} L - \left(M_C - \frac{\Delta t_n}{2} A\right)\right) \mathbf{u}_{n-1} \\
&= \left(M_L - M_C\right) \left(\mathbf{u}_n - \mathbf{u}_{n-1}\right) + \frac{\Delta t_n}{2} D(\mathbf{u}_n + \mathbf{u}_{n-1}).
\end{aligned}$$

Now, the modification of the right hand side of (5) is defined to be

$$\mathbf{f}_i^*(\mathbf{u}_n, \mathbf{u}_{n-1}) = \sum_{j=1}^{N} \alpha_{ij} r_{ij}, \quad i = 1, \ldots, N,$$

with the weights $\alpha_{ij} \in [0, 1]$. FEM-FCT methods determine these weights in such a way that they are close to one in smooth regions (this recovers the Galerkin finite element method) and close to zero at layers (this recovers the stable low order scheme). The other contribution to $\mathbf{f}_i^*(\mathbf{u}_n, \mathbf{u}_{n-1})$ stems from a decomposition of the residual vector

$$\begin{aligned}
r_i = \sum_{j=1}^{N} r_{ij} = \sum_{j=1}^{N} \Big[& m_{ij}(u_{n,i} - u_{n,j}) - m_{ij}(u_{n-1,i} - u_{n-1,j}) \\
& - \frac{\Delta t_n}{2} d_{ij}(u_{n,i} - u_{n,j}) - \frac{\Delta t_n}{2} d_{ij}(u_{n-1,i} - u_{n-1,j}) \Big],
\end{aligned}$$

$i = 1, \ldots, N$. The derivation of this representation uses (3) and (4). The numbers r_{ij} are called fluxes.

The nonlinear FEM-FCT scheme from [20] utilizes an explicit solution $\tilde{\mathbf{u}}$ with the forward Euler scheme at the time $t_n - \Delta t_n/2$

$$\tilde{\mathbf{u}} = \mathbf{u}_{n-1} - \frac{\Delta t_n}{2} M_L^{-1} \left(L\mathbf{u}_{n-1} - \mathbf{f}_{n-1} \right). \tag{6}$$

Using $\tilde{\mathbf{u}}$, a prelimiting of the fluxes is applied in the nonlinear scheme

$$\text{if } r_{ij} \left(\tilde{u}_i - \tilde{u}_j \right) < 0 \text{ then set } r_{ij} = 0,$$

which is recommended in [20,19].

The linear FEM-FCT scheme, which will be used, is a special case of one of the schemes presented in [19]. In this scheme, the vector $\mathbf{u}_n$ in the flux r_{ij} is replaced by an approximation which can be obtain with an explicit scheme. Defining $\mathbf{u}_{n-1/2} = (\mathbf{u}_n + \mathbf{u}_{n-1})/2$ and inserting this expression into r_{ij} leads to

$$\begin{aligned} r_{ij} = & 2m_{ij}(u_{n-1/2,i} - u_{n-1,i}) - 2m_{ij}(u_{n-1/2,j} - u_{n-1,j}) \\ & -\Delta t_n d_{ij}(u_{n-1/2,i} - u_{n-1/2,j}). \end{aligned}$$

An approximation of $\mathbf{u}_{n-1/2}$ can be obtained with the forward Euler scheme applied in the low order method (5) with time step $\Delta t_n/2$, see (6) for the solution $\tilde{\mathbf{u}}$. Inserting this approximation gives

$$r_{ij} = \Delta t_n \Big[m_{ij}(v_{n-1/2,i} - v_{n-1/2,j}) - d_{ij}(\tilde{u}_i - \tilde{u}_j) \Big]$$

with

$$v_{n-1/2,i} = \Big(M_L^{-1}(\mathbf{f}_{n-1} - L\mathbf{u}_{n-1}) \Big)_i.$$

Note that both FEM-FCT schemes use an explicit method as predictor, which results in a CFL condition for these methods, see [20,19].

The weights are computed as described in [16], using Zalesak's algorithm [34], see also [20] for discussions of this algorithm. The algorithm looks as follows

(1) Compute

$$P_i^+ = \sum_{j=1, j\neq i}^{N} \max\{0, r_{ij}\}, \quad P_i^- = \sum_{j=1, j\neq i}^{N} \min\{0, r_{ij}\}.$$

(2) Compute

$$Q_i^+ = \max\{0, \max_{i=1,\ldots,N, j\neq i} (\tilde{u}_j - \tilde{u}_i)\}, \quad Q_i^- = \min\{0, \min_{i=1,\ldots,N, j\neq i} (\tilde{u}_j - \tilde{u}_i)\}.$$

(3) Compute

$$R_i^+ = \min\left\{1, \frac{m_i Q_i^+}{P_i^+}\right\}, \quad R_i^- = \min\left\{1, \frac{m_i Q_i^-}{P_i^-}\right\}.$$

If the denominator is zero, set the value equal to 1.

(4) Compute

$$\alpha_{ij} = \begin{cases} \min\{R_i^+, R_j^-\} \text{ if } r_{ij} > 0, \\ \min\{R_i^-, R_j^+\} \text{ otherwise.} \end{cases}$$

This presentation followed [20]. Note that there is a different scaling in the definition of R_i^+ and R_i^- with respect to the length to the time step in [19], which comes from a different scaling of the fluxes r_{ij} compared with formula (39) in [19].

In all simulations presented in Section 4, it was not exploited that the coefficients of the equations are constant in time such that the computing times correspond to the general situation where the coefficients are time-dependent. This is the case, in particular, if the convection field is a computed flow field. In this situation, the matrix A changes in each discrete time. The standard way to obtain A consists in assembling this matrix every time.

But there is an alternative approach called group finite element method [8]. The group FEM-FCT (GFEM-FCT) was applied already, e.g., in [19,20]. Starting point of the group finite element method is the divergence formulation of the convective term of (1) $\mathbf{b} \cdot \nabla u = \nabla \cdot (\mathbf{b}u)$, where the convection is considered to be divergence-free. The basic idea of this method consists in not only using u, but also the group $(\mathbf{b}u)$, as finite element variable in (1). Let $\{\varphi_j\}_{j=1}^N$ be the finite element basis functions, $\{\mathbf{b}_j\}_{j=1}^N$ be the values of the convection at the nodes, and $\{u_j\}_{j=1}^N$ be the unknown degrees of freedom of the solution. Then, the ansatz for the group finite element method is $(\mathbf{b}u) = \sum_{j=1}^N (\mathbf{b}_j u_j)\varphi_j$, whose insertion leads to the following approximation

$$(\nabla \cdot (\mathbf{b}u), \varphi_i) \approx \sum_{k=1}^{d} \left(\sum_{j=1}^{N} (\partial_k \varphi_j, \varphi_i)(\mathbf{b}_j)_k u_j \right). \tag{7}$$

The matrices $C_k = (\partial_k \varphi_j, \varphi_i)_{i,j=1}^N$, $k = 1, \ldots, d$, have to be assembled only once. To obtain the approximation of the convection matrix from (7), C_k has to be multiplied with the k-th component of the convection. In this way, the group finite element method obtains an approximation of the convection matrix by some multiplications of pre-computed matrices and the current convection vector, instead of applying numerical quadrature. Comparing (7) with the standard assembling procedure, one can see that in the group finite element method the value of the convection at the node j is used instead of the values at the quadrature points around the node j as in the standard approach. Hence, both approaches are not identical but the differences can be expected to be small, in particular on fine grids. Diffusion and reaction are assumed to be independent of time such that the corresponding matrices needed to be assembled also only once.

To our best knowledge, numerical analysis for the FEM-FCT schemes is not available.

3 ENO and WENO Finite-Difference Schemes

On simple domains, finite difference methods are always attractive for discretizing partial differential equations. In this section, some methods based on the ENO and WENO interpolation procedure are described, which are combined with explicit total variation diminishing Runge–Kutta methods. It was noted in the review [29] that for problems in more than one dimension, finite difference methods should be preferred to finite volume approaches if ENO and WENO schemes are used.

Consider first the Runge–Kutta schemes. An optimal second order TVD Runge–Kutta method [30] is the method of Heun, which is given by

$$\begin{aligned}
k_1 &= F(t_{n-1}, u_{n-1}),\\
k_2 &= F\left(t_{n-1} + \Delta t, u_{n-1} + \Delta t k_1\right),\\
u_n &= u_{n-1} + \frac{\Delta t}{2}(k_1 + k_2),
\end{aligned} \tag{8}$$

where $F(t,u) = f + \varepsilon\Delta u - \mathbf{b}\cdot\nabla u - cu$. An optimal third order TVD Runge–Kutta method has the form [30]

$$\begin{aligned}
k_1 &= F(t_{n-1}, u_{n-1}),\\
k_2 &= F(t_{n-1} + \Delta t, u_{n-1} + \Delta t k_1),\\
k_3 &= F\left(t_{n-1} + \frac{\Delta t}{2}, u_{n-1} + \frac{\Delta t}{4}k_1 + \frac{\Delta t}{4}k_2\right),\\
u_n &= u_{n-1} + \Delta t\left(\frac{k_1}{6} + \frac{k_2}{6} + \frac{4k_3}{6}\right).
\end{aligned} \tag{9}$$

The terms on the right hand sides in these schemes will be approximated by finite differences. In the case of dominant convection, the discretization of the first order derivative has to be performed particularly carefully.

Let for simplicity $\Omega = (0,1)^d$. This domain can be triangulated with a grid consisting of grid lines that are parallel to the axes of a Cartesian coordinate system. Thus, it is sufficient to describe the finite difference schemes for a single coordinate, e.g. for x. Consider a partition of $(0,1)$ such that $x_0 = 0 < x_1 < \ldots < x_N = 1$ and set $h_i = x_i - x_{i-1}$, $i = 1,\ldots N$. Fix a discrete time $t = t_{n-1}$ and denote by $u_h^i = u_h(t, x_i)$ the finite difference approximation to

$u(t, x_i)$. For the approximation of the second derivative, the standard central finite difference is used

$$u_{xx}(x_i, t) \approx (u_h^i)_{xx} = 2\frac{(u_h^{i+1} - u_h^i)/h_{i+1} - (u_h^i - u_h^{i-1})/h_i}{h_i + h_{i+1}}.$$

The reactive term $c(t, x_i)u(t, x_i)$ is approximate by $c(t, x_i)u_h^i$.

The simplest stable finite difference discretization of the first order term is the simple upwind scheme, which approximates

$$u_x(t, x_i) \approx (u_h^i)_x = \begin{cases} (u_h^i - u_h^{i-1})/h_i & \text{if } b_1(t, x_i) \geq 0, \\ (u_h^{i+1} - u_h^i)/h_{i+1} & \text{if } b_1(t, x_i) < 0, \end{cases} \tag{10}$$

where $b_1(t, x_i)$ is the first component of the convection vector at (t, x_i).

In the convection-dominated case, the accuracy of the approximation of the convective term is essential for the spatial accuracy of the finite difference method. A more sophisticated choice than simple upwinding consists in approximating $u_x(t, x_i)$ by an ENO interpolation procedure. To obtain a higher order approximation than with the simple upwind scheme, second order information on the numerical solution has to be employed. Consider first the case $b_1(t, x_i) \geq 0$ and define

$$\begin{aligned} a_1 &= \frac{(u_h^{i+1} - u_h^i)/h_{i+1} - (u_h^i - u_h^{i-1})/h_i}{h_i + h_{i+1}} = u[x_{i-1}, x_i, x_{i+1}], \\ a_2 &= \frac{(u_h^i - u_h^{i-1})/h_i - (u_h^{i-1} - u_h^{i-2})/h_{i-1}}{h_{i-1} + h_i} = u[x_{i-2}, x_{i-1}, x_i], \end{aligned}$$

where $u[\cdot]$ denotes divided differences. Dirichlet boundary conditions are extended off Ω to define values outside $[0, 1]$. The basic idea of ENO interpolation consists in using the smoother approximation, where smoothness is measured by the absolute value of the second order divided differences (which is proportional to the curvature). That means, if $|a_1| < |a_2|$, the second degree polynomial based on the nodes $\{x_{i-1}, x_i, x_{i+1}\}$ is applied to approximate the derivative at x_i

$$u_x(t, x_i) \approx (u_h^i)_x = (u_h^i - u_h^{i-1})/h_i + a_1 h_i.$$

Otherwise, the derivative at x_i is approximated using the second degree polynomial based on the nodes $\{x_{i-2}, x_{i-1}, x_i\}$

$$u_x(t, x_i) \approx (u_h^i)_x = (u_h^{i-1} - u_h^{i-2})/h_{i-1} + a_2(2h_i + h_{i-1}).$$

If $b_1(t, x_i) < 0$, the same idea is used, where now the choice is between the two polynomials based on the nodes $\{x_{i-1}, x_i, x_{i+1}\}$ and $\{x_i, x_{i+1}, x_{i+2}\}$.

Now, an ENO scheme of order 3 is introduced. Denote by $P_j(x)$ the polynomial that interpolates the function u at the nodes $\{x_{i+1-j}, x_{i+2-j}, x_{i+3-j}, x_{i+4-j}\}$, $j = 1, \cdots, 4$, and set $a_j = (P_j)_x(x_i)$. Then $u_x(t, x_i)$ will be approximated by an appropriate value a_j. Depending on the sign of $b_1(t, x_i)$, one of the polynomials $P_1(x)$ or $P_4(x)$ is not needed. Thus, this scheme possesses a stencil with the six nodes $\{x_{i-3}, \ldots, x_{i+2}\}$ or $\{x_{i-2}, \ldots, x_{i+3}\}$.

Consider the case $b_1(t, x_i) \geq 0$. Following the ENO strategy, the smoothest approximation is applied. The first smoothness indicator checks a quantity which is proportional to the second derivative of the polynomials through the nodes $\{x_{i-2}, x_{i-1}, x_i\}$ and $\{x_{i-1}, x_i, x_{i+1}\}$, respectively. After this, quantities which are proportional to the third derivative of the two polynomials of degree three, which involve the three nodes which were chosen in the first step, are compared. Thus, the algorithm reads as follows

if $|u[x_{i-1}, x_i, x_{i+1}]| < |u[x_{i-2}, x_{i-1}, x_i]|$
  if $|u[x_{i-2}, x_{i-1}, x_i, x_{i+1}]| < |u[x_{i-1}, x_i, x_{i+1}, x_{i+2}]|$
    choose a_3
  else
    choose a_2
  end
else
  if $|u[x_{i-3}, x_{i-2}, x_{i-1}, x_i]| < |u[x_{i-2}, x_{i-1}, x_i, x_{i+1}]|$
    choose a_4
  else
    choose a_3
  end
end

Note that in the case $b_1(t, x_i) \geq 0$ the stencil is biased to the left. An analogous scheme is used in the case $b_1(t, x_i) < 0$ where the stencil is biased to the right.

A further possibility to increase the order of the approximation of $u_x(t, x_i)$ consists in not taking just one of the polynomials but to use a convex combination of them with appropriate weights, which gives a WENO scheme. The WENO interpolation ideas are explained very well in the review [29]. A WENO scheme of fifth order with a stencil of six nodes, that is used in many applications [29], defines the approximation of $u(t, x_i)$ in the case $b_1(t, x) \geq 0$ as follows [12]

$$
\begin{aligned}
(u_h^i)_{x,1} &= \frac{1}{h}\left(-\frac{u_h^{i-1}}{3} - \frac{u_h^i}{2} + u_h^{i+1} - \frac{u_h^{i+2}}{6}\right),\\
(u_h^i)_{x,2} &= \frac{1}{h}\left(\frac{u_h^{i-2}}{6} - u_h^{i-1} + \frac{u_h^i}{2} + \frac{u_h^{i+1}}{3}\right),\\
(u_h^i)_{x,3} &= \frac{1}{h}\left(-\frac{u_h^{i-3}}{3} + \frac{3}{2}u_h^{i-2} - 3u_h^{i-1} + \frac{11}{6}u_h^i\right),\\
(u_h^i)_x &= \omega_1(u_h^i)_{x,1} + \omega_2(u_h^i)_{x,2} + \omega_3(u_h^i)_{x,3},
\end{aligned}
$$

where the weights ω_i are given by

$$
\begin{aligned}
\omega_i &= \frac{\alpha_i}{\alpha_1+\alpha_2+\alpha_3}, \quad i=1,2,3, \quad \text{with}\\
\alpha_i &= \frac{d_i}{c_e+\beta_i}, \quad i=1,2,3, \quad d_1 = 3/10,\ d_2 = 3/5,\ d_3 = 1/10.
\end{aligned}
$$

The parameter c_e is introduced to avoid that the denominator becomes 0. In the numerical studies $c_e = 10^{-6}$ was used. The values β_i are the so-called smooth indicators of the stencil [12]

$$
\begin{aligned}
\beta_1 &= \frac{13}{12}\left(\overline{u}_h^i - 2\overline{u}_h^{i+1} + \overline{u}_h^{i+2}\right)^2 + \frac{1}{4}\left(3\overline{u}_h^i - 4\overline{u}_h^{i+1} + \overline{u}_h^{i+2}\right)^2,\\
\beta_2 &= \frac{13}{12}\left(\overline{u}_h^{i-1} - 2\overline{u}_h^i + \overline{u}_h^{i+1}\right)^2 + \frac{1}{4}\left(\overline{u}_h^{i-1} - \overline{u}_h^{i+1}\right)^2,\\
\beta_3 &= \frac{13}{12}\left(\overline{u}_h^{i-2} - 2\overline{u}_h^{i-1} + \overline{u}_h^i\right)^2 + \frac{1}{4}\left(\overline{u}_h^{i-2} - 4\overline{u}_h^{i-1} + 3\overline{u}_h^i\right)^2,
\end{aligned}
$$

where $\overline{u}_h^i = \left(u_h^i - u_h^{i-1}\right)/h$ are the cell averages of the first spatial derivative. The case $b_1(t,x) < 0$ is treated in a similar way with the stencil biased to the right.

In higher dimensions, the described approximations of the derivatives have to be applied in each direction. In principle, the combination of both introduced TVD Runge–Kutta methods with all given approximations of the first order derivative is possible. Often used are combinations of discretizations which possess the same or at least a similar order. In the numerical studies presented below, the simple upwind scheme (10) will be combined with the method of Heun (8) and the ENO and WENO schemes with the Runge–Kutta method (9). Since the time-stepping schemes are explicit, a CFL condition applies, similarly to the FEM-FCT schemes.

4 Numerical Studies

This section studies the proposed schemes on some examples defined in two- and three-dimensional domains to quantify their differences in accuracy, in the sizes of under- and overshoots, and in efficiency. For shortness of presentation, only a few characteristic examples are presented.

Several iterative methods were studied for solving the linear system of equations. The basic solver was the (flexible) GMRES method [28]. As preconditioner, the Jacobi method, the SSOR method with overrelaxation parameter 1.5, or a multigrid method were used. The multigrid method was performed with the $V(1,1)$-cycle and the SSOR method was used as smoother, also with overrelaxation parameter 1.5. In 2D, in addition the popular sparse direct solver *UMFPACK* [7] was included in the studies.

The nonlinear problems of the nonlinear FEM-FCT method were solved by a fixed point iteration. Besides the standard fixed point iteration, a so-called Anderson acceleration [1,33] was applied. This acceleration leads to a quasi-Newton method. To build the necessary information, certain vectors from the previous m iterations have to be stored. Results for $m = 3$ and $m = 5$ will be presented. The iterations for solving the linear and the nonlinear problems were stopped if the Euclidean norm of the residual vector was less than 10^{-10}.

All simulations in 2D were performed on the unit square $\Omega = (0,1)^2$. A grid consisting of $N \times N$ squares was used. In 3D, the problem was defined on the unit cube $\Omega = (0,1)^3$ and the grid was given by $N \times N \times N$ cubes. The temporal discretizations were applied with the equi-distant length of the time step Δt. Integrals were evaluated by Gaussian quadrature with two nodes in each direction.

The simulations were performed with the code MooNMD [13] on a HP BL2x220c computer with 2933 MHz Xeon processors. The simulations with the finite difference schemes were double checked with a code written in MATLAB.

Example 1. Transport of an impulse. The first example considers the transport of an impulse along a grid line. It is defined in $\Omega = (0,1)^2$ and the coefficients of the equation are given by $\varepsilon = 10^{-8}$, $\mathbf{b} = (1,0)^T$, $c = 0$, and $f = 0$. On the inflow boundary, $x = 0$, the condition

$$u = \begin{cases} 1 \text{ if } y \in [0.5 - 10^{-8}, 0.5 + 10^{-8}], \\ 0 \text{ else} \end{cases}, \quad t \geq 0,$$

is prescribed. Zero boundary conditions are set at the boundaries $y = 0$ and

$y = 1$ and a homogeneous Neumann boundary condition is prescribed at the outflow boundary $x = 1$. At the initial time, u is set to be zero for all internal degrees of freedom. The number of mesh cells in each coordinate direction was $N = 128$ and the length of the time step $\Delta t = 0.001$.

With this setup, the impulse at the inflow boundary should be transported to the outflow boundary along the grid line $y = 0.5$. This transport should take around one time unit such that the value of the solution at the point $(1, 0.5)$ should raise to become one at this time.

Figure 1 shows the temporal development of $u_h(1, 0.5)$ for the considered schemes. It can be observed that the FEM-FCT schemes failed for this example in the sense that the value at the outlet stayed much smaller than one. The strong smearing of the simple upwind scheme is represented by the smooth transition of $u_h(1, 0.5)$ from zero to one. A much sharper transition was obtained with the ENO finite difference scheme. The transition with the WENO scheme is even sharper. These two schemes show certainly the best results in this example among the considered methods.

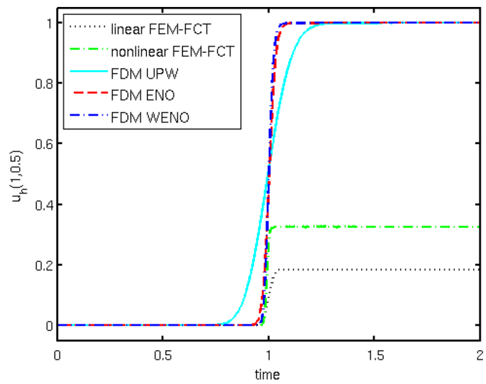

Fig. 1. Transport of an impulse; temporal development of the $u_h(1, 0.5)$.

The different behavior of the methods is due to the fact that a dimensional splitting is used in the finite difference schemes but not in the FEM-FCT schemes. There are no peaks on the horizontal lines but there is a peak on the vertical lines. The finite difference schemes do not see these peaks but the multi-dimensional FEM-FCT schemes do. This aspect leads to a strong clipping of the peaks. One-dimensional versions of the FEM-FCT schemes would cure this situation. The transport of a peak along a grid line will occur seldom in practice such that the results of this example should not be overemphasized. However, it seems us to be important to make the reader aware that schemes can fail in situations where at first sight one would not expect trouble.

Example 2. The rotating body problem. The rotating body problem is a standard example for studying discretizations in the case of very small or even vanishing diffusion [22,20,19]. Let $\Omega = (0, 1)^2$, $\varepsilon = 10^{-20}$, $\mathbf{b} = (0.5-y, x-0.5)^T$, and $c = f = 0$. Initially, three bodies are given, a column with a slit, a cone, and a hump. The position of each body is defined by its center (x_0, y_0) and each of the bodies lies within a circle of radius $r_0 = 0.15$ with the center (x_0, y_0). Outside the three bodies, the initial condition is zero.

Let $r(x, y) = \sqrt{(x - x_0)^2 + (y - y_0)^2}/r_0$. The center of the slotted cylinder is in $(x_0, y_0) = (0.5, 0.75)$ and its geometry is prescribed by

$$u(0; x, y) = \begin{cases} 1 \text{ if } r(x, y) \leq 1, \ |x - x_0| \geq 0.0225 \text{ or } y \geq 0.85, \\ 0 \text{ else.} \end{cases}$$

The conical body at the bottom side is described by $(x_0, y_0) = (0.5, 0.25)$ and

$$u(0; x, y) = 1 - r(x, y).$$

Finally, the hump at the left hand side is given by $(x_0, y_0) = (0.25, 0.5)$ and

$$u(0; x, y) = \frac{1}{4}\Big(1 + \cos(\pi \min\{r(x, y), 1\})\Big).$$

The convection field is prescribed by a rotation around the center $(0.5, 0.5)$ of Ω. If the diffusion is very small, the initial condition should be nearly recovered after one revolution.

First, the case $N = 128$ and $\Delta t = 0.001$ will be studied in more detail since these parameters are often used in the literature, e.g. in [20,19,16]. The solutions obtained with the studied schemes are presented in Figure 2, errors at different times and in different norms are shown in Table 1, and the computing times are given in Table 2. It can be observed in Table 1 that for the FEM-FCT schemes the assembling of matrices in each discrete time and the group finite element approach led in fact to almost identical results. Important criteria for the assessment of the quality of the solutions are the smearing of the layers and the size of under- and overshoots. It can be seen that the simple upwind finite difference method led to a strongly smeared solution, the hump has almost vanished after one rotation. The best solution was clearly obtained with the nonlinear FEM-FCT scheme. Somewhat more smearing was introduced by the WENO scheme. A stronger smearing is clearly visible for the solution obtained with linear FEM-FCT scheme and even a little bit stronger smearing for the result computed with the ENO scheme. The accuracy of the computed solutions from Figure 2, in particular the smearing, is well represented by the errors in $L^2(\Omega)$ in Table 1. With these errors, the superior accuracy of the nonlinear FEM-FCT scheme becomes even more obviously. In addition, it can be seen that the accuracy of the WENO scheme is not much higher than of the linear FEM-FCT scheme.

Notable under- and overshoots could be observed only for the ENO and the

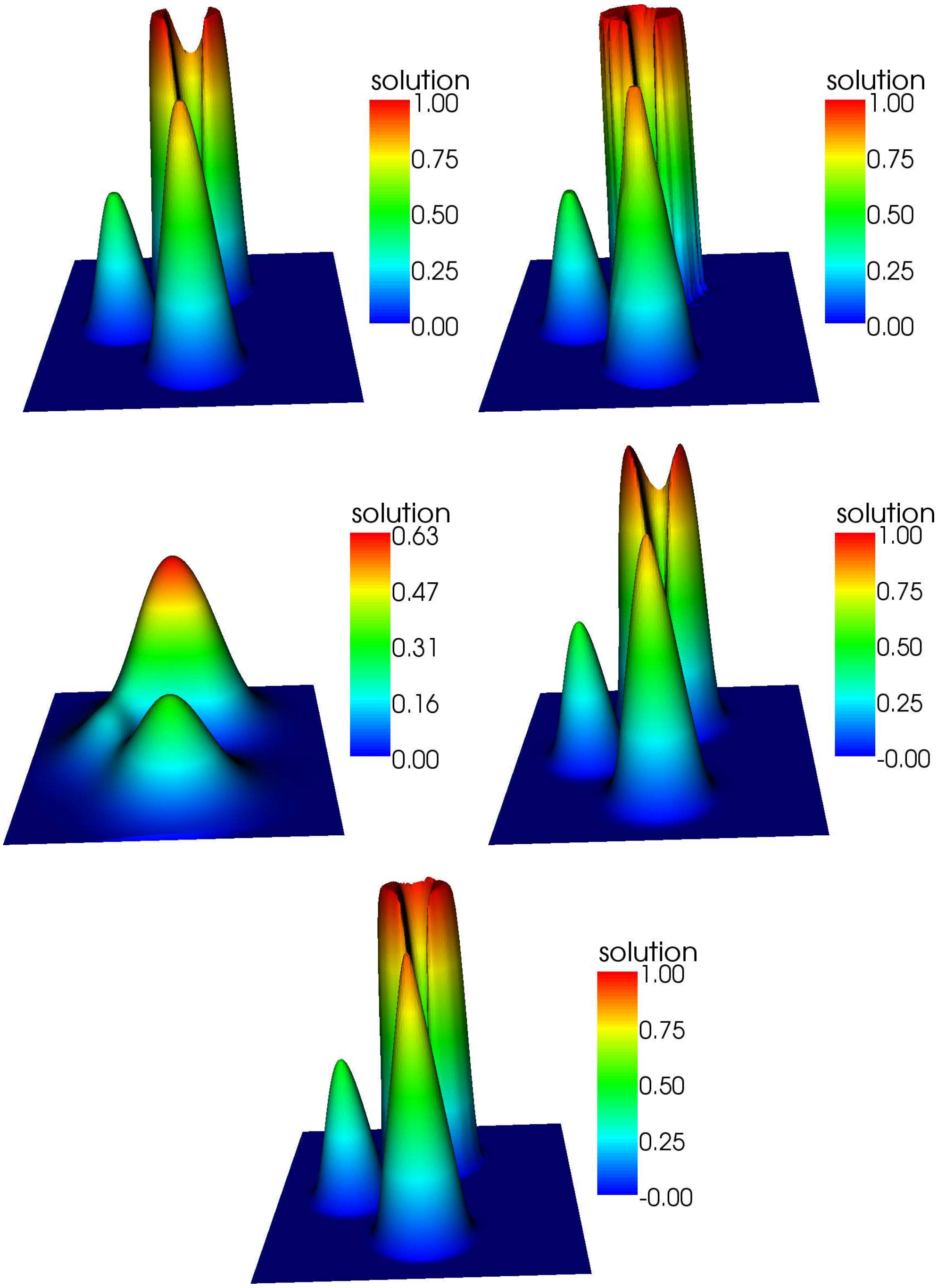

Fig. 2. Rotating body problem, $N = 128$, $\Delta t = 0.001$; solutions after one rotation, linear FEM-FCT, nonlinear FEM-FCT, FDM Upwind, FDM ENO, FDM WENO; from left to right, top to bottom.

WENO scheme, see Figure 3. They were comparably large at the beginning of the simulations, in particular for the WENO scheme. However, this amount of under- and overshoots is still much less than introduced by most of the finite element methods studied in [16] ($20 - 80$ %). The application of prelimiting in the FEM-FCT schemes led to solutions that are free of under- and overshoots

Table 1
Rotating body problem, $N = 128$, $\Delta t = 0.001$; errors e in $L^2(\Omega)$ at different times and error in $L^2(0, 6.28; L^2(\Omega))$.

scheme	$\|e(3.14)\|_{L^2}$	$\|e(6.28)\|_{L^2}$	$\|e\|_{L^2(0,6.28;L^2)}$
FDM upwind + (8)	1.562136e-1	1.782738e-1	3.763692e-1
FDM ENO + (9)	8.993286e-2	9.837932e-2	2.220209e-1
FDM WENO + (9)	7.375359e-2	7.898270e-2	1.835204e-1
lin FEM-FCT + CN	7.927153e-2	8.758556e-2	1.921963e-1
lin GFEM-FCT + CN	7.927153e-2	8.758556e-2	1.921963e-1
nl FEM-FCT + CN	5.844967e-2	6.106624e-2	1.408550e-1
nl GFEM-FCT + CN	5.844967e-2	6.106623e-2	1.408550e-1

if a direct solver was used or at most of the size of the stopping criterion for the iterative solvers. Without prelimiting, small under- and overshoots were observed in [16].

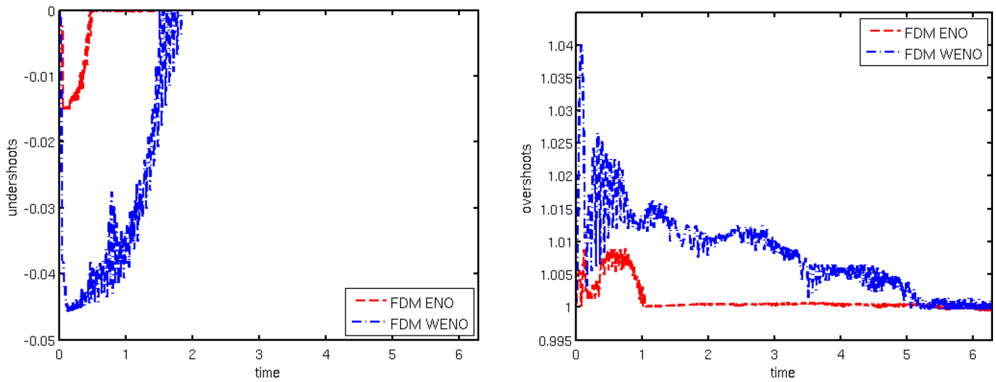

Fig. 3. Rotating body problem, $N = 128$, $\Delta t = 0.001$; undershoots and overshoots with the FDM ENO and the FDM WENO scheme.

Considering the computing times, it can be seen that all explicit finite difference schemes were faster than the implicit finite element methods, as it was expected. With respect to the FEM-FCT schemes, the gains of using the group finite element approach and of applying the Anderson acceleration can be clearly observed. It turned out that the simplest of the considered solvers was most efficient. A main reason for this behavior is the availability of a good initial guess for the iteration, namely the solution of the previous discrete time. Considering methods with comparable results, the ENO simulations were faster by a factor of about two compared with the fastest linear GFEM-FCT scheme. The nonlinear GFEM-FCT scheme took around ten times longer than the WENO scheme. It is probably possible to reduce these differences somewhat for this concrete example by using a different iterative scheme and by optimizing the parameter in the Anderson acceleration.

However, it can be expected that the order of magnitude for the overhead of the FEM-FCT schemes will stay similar.

Table 2
Rotating body problem; computing times in seconds, $N = 128$, $\Delta t = 0.001$.

scheme	direct or expl.	GMRES +Jacobi	GMRES + SSOR	GMRES +MG
FDM upwind + (8)	8			
FDM ENO + (9)	49			
FDM WENO + (9)	86			
lin FEM-FCT + CN	1890	237	252	271
lin GFEM-FCT + CN	1801	111	127	147
nl FEM-FCT + CN + fp	62012	1992	2322	3887
nl FEM-FCT + CN + acce. $m = 3$	26425	1061	1216	1708
nl FEM-FCT + CN + acce. $m = 5$	24433	1025	1167	1614
nl GFEM-FCT + CN + fp	61800	1843	2168	3401
nl GFEM-FCT + CN + acce. $m = 3$	26577	922	1079	1577
nl GFEM-FCT + CN + acce. $m = 5$	24521	890	1034	1492

Results obtained with different refinement levels in space are presented in Table 3. It can be seen that the ranking of the methods with respect to accuracy and efficiency is the same on all levels. In the case of the finite difference methods, the increase in computing time scales with the number of unknowns. The increase is somewhat larger for the FEM-FCT schemes because the number of iterations for the solvers increases with the refinement.

Table 3
Rotating body problem; errors in $L^2(0, 6.28; L^2(\Omega))$ / computing times in seconds, different refinement in space, $\Delta t = 0.001$, FEM-FCT schemes with GMRES + Jacobi.

scheme / N	64	128	256
FDM upwind + (8)	0.434/2	0.376/8	0.323/34
FDM ENO + (9)	0.290/12	0.222/49	0.164/197
FDM WENO + (9)	0.249/21	0.184/86	0.138/347
lin GFEM-FCT + CN	0.260/25	0.192/111	0.140/525
nl GFEM-FCT + CN + acce. $m = 5$	0.201/229	0.141/890	0.102/3809

Table 4 shows that the error in space dominates the error in time. The very slight increase of the errors for the finite difference schemes is due to larger quadrature errors with respect to the temporal variable because of the longer time intervals. The computing times for the finite difference schemes scale with the number of time steps. For the FEM-FCT schemes, the iterative solver has better initial guesses for small time steps and less iterations per time step are needed in this case such that the computing times scale somewhat better than with the number of time steps.

Table 4
Rotating body problem; errors in $L^2(0, 6.28; L^2(\Omega))$ / computing times in seconds, $N = 128$, different length of the time step, FEM-FCT schemes with GMRES + Jacobi.

scheme / Δt	0.01	0.005	0.001
FDM upwind + (8)	0.376/1	0.376/2	0.376/8
FDM ENO + (9)	0.218/5	0.219/10	0.222/49
FDM WENO + (9)	0.183/8	0.184/17	0.184/86
lin GFEM-FCT + CN	0.198/13	0.195/25	0.192/111
nl GFEM-FCT + CN + acce. $m = 5$	0.155/158	0.145/248	0.141/890

Accuracy per computing time is illustrated in Figure 4. The best simulations in this diagram are those in the lower left corner since they are accurate and they were obtained in a short computing time. Accordingly, the WENO and the ENO scheme with the explicit Runge–Kutta methods are the most efficient schemes with respect to the considered error. But the linear GFEM-FCT method is often only a little less efficient than those schemes.

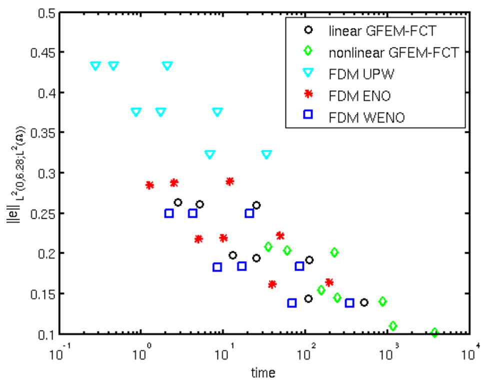

Fig. 4. Rotating body problem; computing time vs. error in $L^2(0, 6.28; L^2(\Omega))$, simulations with $N \in \{64, 128, 256\}$, $\Delta t \in \{0.01, 0.005, 0.001\}$; for the combination of $N = 256$ and $\Delta t = 0.01$ all methods were unstable.

Example 3. Transport of a species through a three-dimensional domain. This example was defined in [17]. It models a typical situation that is encountered in applications. A species enters a domain and it is transported through the domain to an outlet. In the domain, the species is diffused somewhat and in the subregion where the species is transported, also a reaction occurs.

Let $\Omega = (0,1)^3$ with the inlet at $\{0\} \times (5/8, 6/8) \times (5/8, 6/8)$ and the outlet at $\{1\} \times (3/8, 4/8) \times (4/8, 5/8)$. The convection $\mathbf{b} = (1, -1/4, -1/8)^T$ points from the inlet to the outlet. The diffusion is prescribed by $\varepsilon = 10^{-6}$ and the reaction by

$$c(\mathbf{x}) = \begin{cases} 1 \text{ if } \|\mathbf{x} - g\|_2 \leq 0.1, \\ 0 \text{ else,} \end{cases}$$

where g is the line through the center of the inlet and the center of the outlet and $\|\cdot\|_2$ denotes the Euclidean norm. This ratio of diffusion and convection can be found in many applications. There are no sources, i.e. $f = 0$.

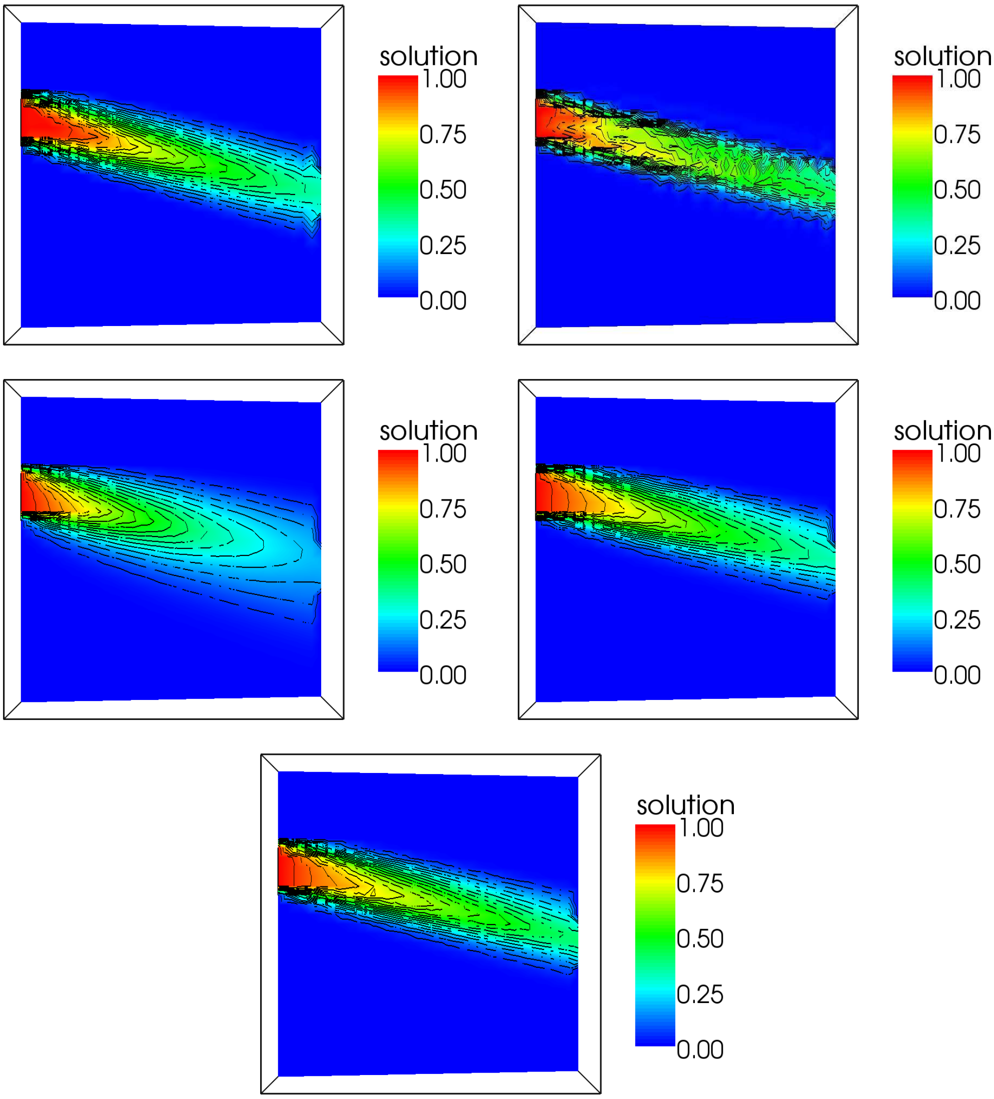

Fig. 5. Transport of a species through a three-dimensional domain, $N = 32$; solutions at $t = 2$, linear FEM-FCT, nonlinear FEM-FCT, FDM Upwind, FDM ENO, FDM WENO; from left to right, top to bottom.

In contrast to the first example, the transport of the species does not occur along grid lines. At the initial time, there is no species inside the domain,

$u(0, \mathbf{x}) = 0$. Then, the injection of the species starts, it increases, stays constant for a while, and finally it decreases. A boundary condition at the inlet describing this process is given by

$$u_{\rm in}(t) = \begin{cases} \sin(\pi t/2) & \text{if } t \in [0,1], \\ 1 & \text{if } t \in (1,2], \\ \sin(\pi(t-1)/2) & \text{if } t \in (2,3]. \end{cases}$$

Homogeneous Neumann boundary conditions are prescribed at the outlet and homogeneous Dirichlet conditions on the rest of the boundary.

Result are presented for $N = 16$, $N = 32$, and $\Delta t = 0.001$. Solutions for $N = 32$ at $t = 2$ are given in Figure 5. The cut planes contain the line between the center of the inlet and the center of the outlet. In addition, evolutions of the concentration at the center of the outlet are given in Figure 6. These curves provide information of the amount of loss of concentration due to the smearing of the solutions by numerical diffusion. It can be observed that the solutions obtained with the finite difference schemes and the linear FEM-FCT scheme were much smoother than the solution computed with the nonlinear FEM-FCT scheme. The latter solution can be considered to be the most accurate solution obtained in this study, note also the largest concentration at the center of the outlet, Figure 6. The finite difference simple upwind scheme was again much more diffusive than the other schemes. Only a comparable small amount of the species reached the outlet. Based on Figures 5 and 6, the second most accurate solutions were obtained with the WENO scheme. Then the solutions computed with the linear FEM-FCT scheme follow, which in turn are slightly better than the solutions obtained with the ENO scheme. Thus, on the one hand, the nonlinear FEM-FCT scheme and the WENO scheme and on the other hand, the linear FEM-FCT scheme and the ENO scheme led to similar results, with the FEM-FCT schemes in both cases somewhat more accurate. Due to the coarser grids, we could observe that the differences between the standard and the group finite element version of the FEM-FCT schemes were more visible than in Example 2. However, the overall quality of the respective solutions was the same.

Undershoots, which are larger than those caused by the stopping criteria of the iterations, could be observed in this example only for the solution computed with the WENO scheme. They were of order 10^{-5}.

For the FEM-FCT scheme it is not clear if the incorporation of the reactive term in the matrix A in (2) and consequently in the FCT algorithm is an appropriate way. For this reason, simulations were performed with an explicit treatment of the reactive term such that only diffusion and convection were treated with the FCT algorithm. It can be seen in Figure 7 that almost the

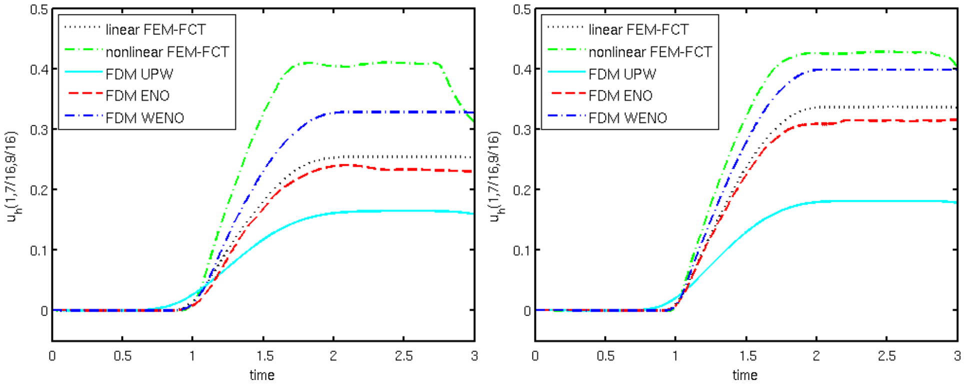

Fig. 6. Transport of a species through a three-dimensional domain; temporal development of the numerical solutions at the center of the outlet, left $N = 16$, right $N = 32$, standard version of FEM-FCT schemes.

same results were obtained. Thus, in the case that reaction does not dominate, the inclusion of the reactive term in the matrix A seems to be a reasonable approach. This might change in the reaction-dominated regime, whose study is, however, not within the scope of this paper.

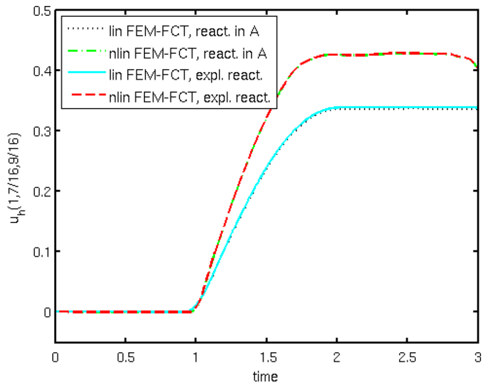

Fig. 7. Transport of a species through a three-dimensional domain, $N = 32$; comparing the approach of including the reactive term into the matrix A for the FCT algorithm and for treating the reactive term explicitly, standard version of FEM-FCT schemes.

Since the scalings of the computing times with respect to the length of the time step and to the number of degrees of freedom were similar as in Example 2, detailed results are presented only for one grid in time and space, see Table 5. Again, the higher efficiency of the group version of the FEM-FCT schemes and the positive impact of using the Anderson acceleration can be clearly observed. Comparing methods with similar accuracy, the simulation with the ENO scheme was faster of nearly four times than the fastest simulation with the linear group FEM-FCT scheme, which is twice as much as in Example 2. Obviously, the overhead of solving the linear systems of equations is larger in three dimensions. The solution of the nonlinear problems in the FEM-FCT scheme needed less iterations than in Example 2. It turned out that the WENO scheme was about thirteen times faster compared with the fastest simulation of the nonlinear group FEM-FCT scheme. Altogether, the differences in efficiency are somewhat larger than in the two-dimensional problem. But they are not so large that it should be dissuaded from the use of the FEM-FCT schemes, depending of course on the requirements of the concrete application. Since there is no actual error which can be measured in this example, the accuracy per computing time could not be studied.

Table 5
Transport of a species through a three-dimensional domain $\Delta t = 0.001$, $N = 32$; computing times in seconds.

scheme	GMRES +Jacobi	GMRES + SSOR	GMRES +MG
FDM upwind + (8)	15		
FDM ENO + (9)	70		
FDM WENO + (9)	119		
lin FEM-FCT + CN	926	942	987
lin GFEM-FCT + CN	265	275	333
nl FEM-FCT + CN + fp	3858	4136	6544
nl FEM-FCT + CN + acce. $m = 3$	2517	2827	3084
nl FEM-FCT + CN + acce. $m = 5$	2265	2413	3013
nl GFEM-FCT + CN + fp	3434	3759	6174
nl GFEM-FCT + CN + acce. $m = 3$	1997	2047	2512
nl GFEM-FCT + CN + acce. $m = 5$	1635	1855	2559

5 Further Aspects of the Methods

This section discusses some further aspects of the methods, like adaptivity and more complicated domains. Both issues deserve comprehensive studies in future, which are, however, beyond the scope of this paper.

Depending on the concrete example, adaptivity in time and space might be very useful to increase the efficiency of the methods.

Consider first the FEM-FCT schemes which are combined with simple implicit time-stepping schemes. Adaptive time step control is often based on the comparison of the results of two time-stepping schemes. Besides the Crank–Nicolson scheme, one has to perform the current time step with another scheme, like backward Euler or the fractional-step θ-scheme, as it was proposed for the time-dependent Navier–Stokes equations in [31]. This procedure roughly doubles the costs per time step. Somewhat less expensive is the use of an explicit scheme for comparison, like the Adams–Bashforth scheme as proposed in [9]. A heuristic approach, which does not increase the computational costs, consists in measuring the rate of change of the solutions of two subsequent time steps [32,26]. Concerning adaptivity in space, standard error indicators can be used for adaptively refining or coarsening the grids. Such indicators include the gradient indicator or the Zienkiewicz–Zhu indicator [35]. Since there is no numerical analysis of FEM-FCT schemes for time-dependent equations available, there are, in particular, no a posteriori error estimators, i.e., there are no computable quantities which bound the error of the FEM-FCT schemes in a rigorous way. Adaptive grid refinement in combination with FEM-FCT methods can be found, e.g., in [24,26].

Adaptive time step control for the upwind, ENO, and WENO schemes combined with explicit TVD Runge–Kutta methods can be done easily via embedded schemes. In such schemes, only a different linear combination of the stages k_i is applied to obtain a scheme with one order less. Appropriate linear combinations for the schemes (8) and (9) are

$$u_n = u_{n-1} + \Delta t k_1, \quad u_n = u_{n-1} + \frac{\Delta t}{2}(k_1 + k_2),$$

respectively. The embedded schemes are just the forward Euler scheme and the method of Heun. In this way, results of two schemes with different orders are obtained with negligible additional costs. Based on these results, the time step can be controlled, e.g., with the PI or PID controller [10]. The extension of the considered schemes to include adaptive time step control in this way is straightforward.

Finite difference methods can be used most easily on regular grids, consisting of lines which are parallel to the coordinate axes. This requirement makes the application of adaptive grid refinement or coarsening in multiple dimensions less flexible than for finite element methods, because this property of the grid can be kept only by removing or adding whole grid lines. For 1D problems, ENO and WENO schemes with spatial adaption were presented in [5,18]. This adaption is based on a sparse point representation using wavelets. In multiple dimensions, a substantial gain in efficiency can only be expected, if the features of the solution, which require a fine grid, are concentrated in a small part of the domain. Optimally, these features should be aligned to one of the coordinate axes. For convection-diffusion equations, layers may have this property, since these are local features. As indicator for adding and removing grid lines, a criterion can serve that is similar to the gradient indicator, i.e., grid lines are included where the gradient of the current approximation of the solution is large and they are removed where the gradient is very small. Thus, a simple adaptive algorithm for the ENO scheme in one dimension looks as follows:

(1) Choose an initial subdivision of the interval $[0, 1]$.
(2) Compute the ENO approximation at the first discrete time.
(3) For every subinterval $I_i = [x_{i-1}, x_i]$ compute $e_i = |u_h^i - u_h^{i-1}|$, $i = 1, \ldots, N$.
(4) If e_i is greater than a given tolerance tol_{upp}, then halve the interval I_i, whenever the new intervals do not become smaller than a prescribed minimal interval size $I_{\min}$.
(5) If e_i and e_{i+1} are both less than a given tolerance tol_{low}, suppress x_i, whenever the new interval does not exceed a maximum prescribed size $I_{\max}$.
(6) Insert nodes such that the ratio of the lengths of two subsequent intervals of the new grid is not smaller than 0.5 and not larger than 2.
(7) Interpolate the ENO approximation to the new mesh and use it as initial condition for the next time step.
(8) Continue with the procedure until the final time is reached.

In multiple dimensions, a new grid can be chosen first with respect to the x-coordinate and the solution is interpolated to this new grid, then the same procedure is applied for the y-coordinate and finally for the z-coordinate. The insertion of nodes in Step 6 prevents the situation that neighboring intervals are of very different sizes. Since the stencil of the ENO scheme covers several intervals, a rather smooth transition of the lengths of the intervals seems to be advisable. In fact, we could observe in numerical tests that the application of Step 6 increased the accuracy of the obtained results considerably compared with the adaptive algorithm without Step 6. An appropriate example for adaptive grid refinement is certainly the transport of an impulse from Example 1. A typical result and the evolution of $u_h(1, 0.5)$ are presented in Figure 8. Using the adaptive grid, the speed-up of the simulations was around 25 – 30. However, the solution on the adaptive grid is more smeared than on the uniform grid. We have observed an increase of the smearing on adaptive grids in other examples in one and two dimensions, too. In most of the examples, the speed-up obtained with the adaptive simulations led to an improvement of the efficiency since an adaptive approximation of comparable accuracy to a uniform approximation could be computed in less CPU time. Further research is needed to improve the proposed adaptive algorithm such that less smeared solutions are computed.

Another possibility to increase the efficiency of ENO and WENO schemes consists in combining them with other finite difference methods. For instance, in a vicinity of the layers, the WENO scheme is used and in smooth regions a less expensive scheme is applied. This approach belongs to the class of hybrid adaptive ENO schemes as proposed in [2].

An important aspect are also more complicated domains than $\Omega = (0, 1)^d$. A detailed discussion of this issue, in particular for the finite difference schemes, is beyond the scope of this paper. It will be only indicated that there are ways

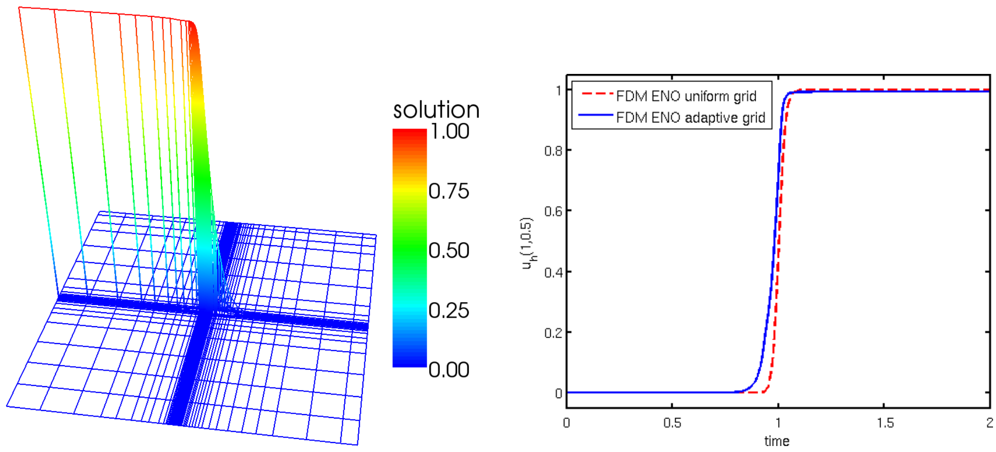

Fig. 8. Transport of an impulse; adaptive grid at $t = 0.5$ with $I_{\max} = 0.1$, $I_{\min} = 0.001$, $tol_{\rm upp} = 0.1$, and $tol_{\rm low} = 0.001$, and temporal development of $u_h(1, 0.5)$.

to deal with this issue also for the finite difference ENO and WENO schemes.

If the domain is more complicated than a tensor product of intervals, FEM-FCT schemes can be applied without modification, even on triangular or tetrahedral grids. For the finite difference methods, an approach similar to the spectral smoothed boundary method can be used, which was proposed in [4]. The basic idea consists in including the domain Ω in a tensor product domain $\tilde{\Omega}$ and to triangulate $\tilde{\Omega}$ by a tensor product mesh. Considering first homogeneous Dirichlet boundary conditions, then (1) is solved in Ω and the solution at the nodes in $\tilde{\Omega} \setminus \Omega$ is just set to be zero. An example of this strategy is presented in Figure 9. For the simulations of this example $\Omega = \{(x_1, x_2) : (x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.4^2\} \setminus \{(x_1, x_2) : (x_1 - 0.5)^2 + (x_2 - 0.5)^2 < 0.2^2\}$, $\varepsilon = 10^{-10}$, $\mathbf{b} = (2, 3)^T$, $c = 0$, $f = 1$, $u_0(x_1, x_2) = 0$, and $T = 0.6$ were used. The third order ENO method was applied and a solution with sharp boundary layers without visible oscillations was obtained.

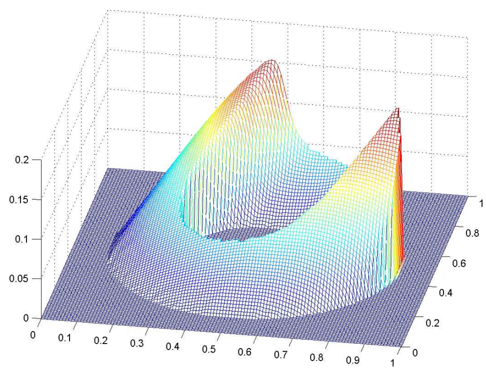

Fig. 9. A problem in a more complicated domain, result obtained with the FDM ENO method.

In the case of non-homogeneous Dirichlet boundary conditions, one can try to apply a trick which is used in the numerical analysis of finite element methods. One has to find a sufficiently smooth function $\tilde{u}$ which fulfills the boundary condition and considers an equation for $u - \tilde{u}$. Of course, depending on the actual boundary condition, it might be difficult to find an appropriate function. Homogeneous Neumann boundary conditions might be treated in the same fashion as in [4] by employing a mollification of the characteristic function of Ω.

6 Conclusions

This paper studied the accuracy and efficiency of finite element and finite difference discretizations for evolutionary convection-diffusion-reaction equations which give solutions without under- and overshoots or with small under- and overshoots. The simplest situation was considered in detail, namely that the domain is a tensor product of intervals and that the discretizations are applied uniformly in time and space.

Concerning the accuracy, the following observations were made:

- The most accurate results were generally obtained with the nonlinear FEM-FCT scheme.
- The differences in accuracy of the results obtained with linear FEM-FCT scheme and the finite difference ENO and WENO schemes were often not large. Among these schemes, the best results were generally computed with the WENO scheme and the less accurate results with the ENO scheme.
- The simple upwind scheme led to very inaccurate solutions due to a very large smearing of the layers.
- In all considered examples, no under- and overshoots (or at most of magnitude of the stopping criteria for the iterative solvers) could be observed for the FEM-FCT schemes and the simple upwind finite difference method.
- For the ENO scheme and especially for the WENO scheme, small under- or overshoots were often present.

A very special problem was constructed, Example 1, for which the two-dimensional version of the FEM-FCT schemes led to unsatisfactory results.

With respect to efficiency, the following results were obtained:

- The group version GFEM-FCT of the FEM-FCT schemes was more efficient than the standard version with repeated matrix assembling.
- A simple iterative solver was sufficient for solving the linear systems of equations in the FEM-FCT schemes efficiently.
- An acceleration technique, like the Anderson acceleration, increases the efficiency of solving the nonlinear equations of the nonlinear FEM-FCT scheme

considerably.

- On a fixed mesh in time and space:
  - Simple upwinding was by far the fastest approach. It was faster by a half to one order of magnitude compared with the ENO and WENO scheme.
  - The nonlinear FEM-FCT scheme took the most computing time.
  - The ENO scheme needed roughly half of the computing time of the WENO scheme.
  - The ENO scheme was faster than the linear GFEM-FCT scheme by a factor of two to four.
  - The WENO scheme was faster by about one order of magnitude than the nonlinear GFEM-FCT scheme.
  - The differences in computing time between the GFEM-FCT schemes and the finite difference schemes were larger in three dimensions than in two dimensions.
- Considering accuracy per computing time, the differences between the schemes became smaller. The linear GFEM-FCT scheme might be even competitive with the finite difference schemes. However, this aspect depends on the example.

The actual recommendations from these results depend on the requirements of the considered application:

- under- or overshoots cannot be tolerated:
  - computing time is a strong issue: use the simple upwind scheme,
  - computing time and accuracy are both important issues: use the linear GFEM-FCT scheme,
  - accuracy is a strong issue: use the nonlinear GFEM-FCT scheme,
- small under- or overshoots can be tolerated or the cut-off of small under- and overshoots can be tolerated:
  - computing time is a strong issue: use the ENO scheme,
  - otherwise: use the WENO scheme.

Some remarks on more general situations were also provided in Section 5. Comprehensive studies of such situations are future research.

Acknowledgements

We would like to acknowledge three unknown referees whose suggestions greatly helped to improve this paper. In particular, the explanation of the behavior of the FEM-FCT schemes in Example 1 was provided by one of the referees.

References

[1] D.G. Anderson. Iterative procedures for nonlinear integral equations. *J. Assoc. Comput. Machinery*, 12:547 – 560, 1965.

[2] R.B. Bauer. A hybrid adaptive ENO scheme. *J. Comput. Phys.*, 136:180 – 196, 1997.

[3] R. Bordás, V. John, E. Schmeyer, and D. Thévenin. Measurement and simulation of a droplet population in a turbulent flow field. Preprint 1590, WIAS, 2011.

[4] A. Bueno-Orovio, V.M. Pérez-García, and F.H. Fenton. Spectral methods for partial differential equations in irregular domains: the spectral smoothed boundary method. *SIAM J. Sci. Comput.*, 28:886 – 900, 2006.

[5] R. Bürger and A. Kozakevicius. Adaptive multiresolution WENO schemes for multi-species kinematic flow models. *J. Comput. Phys.*, 224:1190 – 1222, 2007.

[6] E. Burman and M.A. Fernández. Finite element methods with symmetric stabilization for the transient convection-diffusion-reaction equation. *Comput. Methods Appl. Mech. Engrg.*, 198:2508 – 2519, 2009.

[7] T.A. Davis. Algorithm 832: UMFPACK V4.3 – an unsymmetric-pattern multifrontal method. *ACM Trans. Math. Software*, 30:196 – 199, 2004.

[8] C.A.J. Fletcher. The group finite element formulation. *Int. J. Numer. Methods Fluids*, 37:225 – 243, 1983.

[9] P.M. Gresho, D.F. Griffiths, and D.J. Silvester. Adaptive time-stepping for incompressible flow part I: Scalar advection-diffusion. *SIAM J. Sci. Comp.*, 30:2018 – 2054, 2008.

[10] K. Gustafsson, M. Lundh, and G. Söderlind. A PI stepsize control for the numerical solution of ordinary differential equations. *BIT*, 28:270 – 287, 1988.

[11] A. Harten, B. Enquist, S. Osher, and S. Chakravarthy. Uniformly high order essentially non-oscillatory schemes III. *J. Comput. Phys.*, 71:231 – 303, 1987.

[12] G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *J. Comput. Phys.*, 126:202 – 228, 1996.

[13] V. John and G. Matthies. MooNMD - a program package based on mapped finite element methods. *Comput. Visual. Sci.*, 6:163 – 170, 2004.

[14] V. John and J. Novo. Error analysis of the supg finite element discretization of evolutionary convection-diffusion-reaction equations. *SIAM J. Numer. Anal.*, 49:1149 – 1176, 2011.

[15] V. John, M. Roland, T. Mitkova, K. Sundmacher, L. Tobiska, and A. Voigt. Simulations of population balance systems with one internal coordinate using finite element methods. *Chem. Engrg. Sci.*, 64:733 – 741, 2009.

[16] V. John and E. Schmeyer. Stabilized finite element methods for time–dependent convection–diffusion–reaction equations. *Comput. Methods Appl. Mech. Engrg.*, 198:475 – 494, 2008.

[17] V. John and E. Schmeyer. On finite element methods for 3d time–dependent convection–diffusion–reaction equations with small diffusion. In A. Hegarty et al., editor, *BAIL 2008 – Boundary and Interior Layers*, number 69 in Lecture Notes in Computational Science and Engineering, pages 173 – 181. Springer, 2009.

[18] A.J. Kozakevicius and L.C.C. Santos. ENO adaptive method for solving one-dimensional conservation laws. *Appl. Numer. Math.*, 59:2337 – 2355, 2009.

[19] D. Kuzmin. Explicit and implicit FEM-FCT algorithms with flux linearization. *J. Comput. Phys.*, 228:2517 – 2534, 2009.

[20] D. Kuzmin and M. Möller. Algebraic flux correction I. Scalar conservation laws. In R. Löhner D. Kuzmin and S. Turek, editors, *Flux-Corrected Transport: Principles, Algorithms and Applications*, pages 155 – 206. Springer, 2005.

[21] D. Kuzmin and S. Turek. Flux correction tools for finite elements. *J. Comp. Phys.*, 175:525 – 558, 2002.

[22] R.J. LeVeque. High–resolution conservative algorithms for advection in incompressible flow. *SIAM J. Numer. Anal.*, 33:627 – 665, 1996.

[23] X.-D. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 115:200 – 212, 1994.

[24] R. Löhner, K. Morgan, J. Peraire, and M. Vahdati. Finite element flux-correct transport (FEM-FCT) for the Euler and the Navier–Stokes equations. *Int. J. Numer. Meth. Fluids*, 7:1093 – 1109, 1987.

[25] H. Luo, J.D. Baum, and R. Löhner. A Hermite WENO-based limiter for discontinuous galerkin method on unstructured grids. *J. Comput. Phys.*, 225:686 – 713, 2007.

[26] M. Möller. *Adaptive High-Resolution Finite Element Schemes.* PhD thesis, Technical University Dortmund, 2008.

[27] H.-G. Roos, M. Stynes, and L. Tobiska. *Robust Numerical Methods for Singularly Perturbed Differential Equations*, volume 24 of *Springer Series in Computational Mathematics.* Springer, 2nd edition, 2008.

[28] Y. Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.*, 14(2):461–469, 1993.

[29] C.-W. Shu. High order weighted essentially nonoscillatoy schemes for convection dominated problems. *SIAM Review*, 51:82 – 126, 2009.

[30] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes. *J. Comput. Phys.*, 77:439 – 471, 1988.

[31] S. Turek. *Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach*, volume 6 of *Lecture Notes in Computational Science and Engineering*. Springer, 1999.

[32] A.M.P. Valli, G.F. Carey, and A.L.G.A. Coutinho. Control strategies for timestep selection in finite element simulation of incompressible flows and coupled reaction-convection-diffusion processes. *Int. J. Numer. Meth. Fluids*, 47:201 – 231, 2005.

[33] H.F. Walker and P. Ni. Anderson acceleration for fixed-point iterations. *SIAM J. Numer. Anal.*, 49:1715 – 1735, 2011.

[34] S.T. Zalesak. Fully multi-dimensional flux corrected transport algorithms for fluid flow. *J. Comput. Phys.*, 31:335 – 362, 1979.

[35] O.C. Zienkiewicz and J.Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *Internat. J. Numer. Methods Engrg.*, 24(2):337 – 357, 1987.