Finite-Volume Discretizations and Immersed Boundaries

Yunus Hassen and Barry Koren

Abstract In this chapter, an accurate method, using a novel immersed-boundary approach, is presented for numerically solving linear, scalar convection problems. As is standard in immersed-boundary methods, moving bodies are embedded in a fixed 'Cartesian' grid. The essence of the present method is that specific fluxes in the vicinity of a moving body are computed in such a way that they accurately accommodate the boundary conditions valid on the moving body. To suppress wiggles, tailor-made limiters are introduced for these special fluxes. The first results obtained are very accurate, without requiring much computational overhead. It is anticipated that the method can readily be extended to real fluid-flow equations.

1 Introduction

The immersed-boundary method – or, synonymously, embedded-boundary method – is a method in which boundary conditions are indirectly incorporated into the governing equations. It has undergone numerous modifications, ever since its introduction by Peskin in 1972 [13], and currently many varieties of it exist (see [10] for a review and the references therein for details).

Immersed-boundary methods are very suitable for simulating flows around flexible, moving and/or complex bodies. Basically, the bodies of interest are just em-

Yunus Hassen
Centrum Wiskunde & Informatica, Kruislaan 413, 1098 SJ Amsterdam, the Netherlands,
Faculty of Aerospace Engineering, TU Delft, Kluyverweg 1, 2629 HS Delft, the Netherlands.
e-mail: yunus.hassen@cwi.nl

Barry Koren
Centrum Wiskunde & Informatica, Kruislaan 413, 1098 SJ Amsterdam, the Netherlands,
Faculty of Aerospace Engineering, TU Delft, Kluyverweg 1, 2629 HS Delft, the Netherlands,
Mathematical Institute, Leiden University, Niels Bohrweg 1, 2333 CA Leiden, the Netherlands.
e-mail: barry.koren@cwi.nl

B. Koren and C. Vuik (eds.), *Advanced Computational Methods in Science and Engineering*,
Lecture Notes in Computational Science and Engineering 71,
DOI 10.1007/978-3-642-03344-5_8, © Springer-Verlag Berlin Heidelberg 2010

bedded in non-deforming Cartesian grids that do not conform to the shape of the body. The governing equations are modified to include the effect(s) of the embedded boundaries. Doing so, mesh (re)generation difficulties associated with body-fitted grids are obviated, and the underlying regular fixed grid allows us to use a simple data structure as well as simpler numerical schemes over a majority of the domain.

Peskin, in his original paper [13], introduced the idea of replacing an object in a flow by a field of forces. This gave rise to the notion of an 'immersed body.' Peskin described the fluid variables in an Eulerian manner and the object in the Lagrangian manner and computed, from the boundary configuration, the elastic forces generated within the object. Since the object is in direct contact with the fluid, these elastic forces affect the fluid motion. Peskin then transmitted the forces to the fluid in the immediate vicinity of the boundaries of the object, i.e., to the governing equations he added a forcing function, which is zero everywhere except near the immersed boundaries. The forcing term enforces the no-slip condition on the boundaries of the (immersed) object and thus the flow field indirectly feels the presence of an object immersed in it. Finally, he discretized the extended equation in the entire computational domain, including inside the immersed body. He successfully implemented this immersed-boundary method to simulate blood flow in and around heart valves [14, 15].

In a similar way, but independently, Goldstein et al. [4] employed a forcing term continuously computed from a feedback loop. They borrowed concepts from linear control theory and formulated the forcing term depending solely on the velocity of the boundary-surface points (immersed boundaries). This forcing term is added to the momentum equation, and recomputed/corrected (using the computed velocity) at each time step until the relative velocity on the desired boundary-surface points has been set to zero. This recursive procedure eventually evolves the (desired) virtual surface. The subjective part of this forcing term is that it requires the choice of two negative constants, α and β, and a problem-dependent parameter k, which are not defined properly; they are interrelated by a stability criterion and estimated heuristically. This technique introduces a severe restriction on the time step and it is unstable for (complex) flow computations that require large time steps. Goldstein et al. used a spectral method solver to simulate two-dimensional flow around stationary cylinders and three-dimensional turbulent channel flow. Saiki and Biringen [16] adopted the same (feedback-forcing-function) method using higher-order finite-difference methods, and achieved a relatively stable solution with no time-step restriction. They computed the feedback-forcing function by integrating the relative flow velocity, with the associated negative constants, on the boundary-surface points, and showed that the feedback-forcing method of Goldstein et al. is also capable of handling moving boundary problems. They successfully implemented it for low-Reynolds number (Re ≤ 400) flows around fixed, rotating and oscillating cylinders.

Mohd-Yusof [12] proposed a modified method which is called the direct-forcing method. This method uses a set of points adjacent to the surface and interior to the body and directly imposes the no-slip boundary conditions on the immersed boundary enabling a direct momentum forcing. It is relatively stable, compared to the method of Goldstein et al., does not impose time-step restrictions and does not need the choice of (empirical) negative constants for defining the forcing function. Mohd-Yusof [11] also used the spectral context and simulated three-dimensional flows for complex geometries. Kim et al. [7] used the direct-forcing method and simulated flows over a cylinder and a sphere using the finite-volume approach on a staggered mesh. They introduced external forcing functions to the momentum and continuity equations, to achieve momentum and mass conservation, respectively. In this approach, an unbalanced mass flux results across the body boundary and a source/sink term is added to the continuity equation.

Fadlun et al. [2] extended the direct-forcing method to finite-difference formulation on a staggered grid, and compared the accuracy and efficiency of their method with those of the feedback forcing method, by simulating three-dimensional flows in complex geometries. They found their method to be of the same order of accuracy, but more efficient. They concluded that the direct-forcing approach is more efficient and suitable to simulate unsteady three-dimensional incompressible flows in complex geometries. Most of the feedback and direct-forcing methods are basically similar except for the way of interpolating the fluid velocity to the immersed boundary points and extrapolating the body force back into the computational grid points [2, 17, 19, 20].

More recently, a different class of immersed-boundary methods has started to emerge. Here, no forcing function and spreading is required. Instead, the velocity of grid points around the immersed boundary is interpolated taking the boundary condition (no-slip, for instance) into account. The resulting interpolation equation is then solved along with the (unmodified) Navier-Stokes equations. The main advantage, in this case, is that no extra terms are included in the governing equations and they are solved only in the fluid domain. Ghost-cell [19] and cut-cell [1] methods are typical examples of this class of methods. Some of the major contemporary methods have been described and reviewed in [10].

In this chapter, we follow the forcing-function-free approach and start to build up a new immersed boundary method from scratch, considering for convenience, a simple model equation. Our approach uses a cell-centered finite-volume discretization. The governing partial differential equations are discretized using a standard finite-volume method away from an embedded boundary (EB). Near the EB, a special finite-volume method is derived which takes the prescribed interior boundary conditions into account.

The outline of the chapter is as follows. In § 2, the problem is described, a standard finite-volume method is described and some of the associated results are presented. The special fluxes which take the effects of the embedded boundaries into account are derived and limiters are introduced in § 3. In § 4, the issues associated with temporal discretization, which gives rise to fully discrete equations, are explained. Total-variation diminishing (TVD) regions are defined and tailor-made limiters, for the special fluxes, are also educed from the fully discrete equations. In § 5, some numerical results, based on the present work, are given and a comparison is made with the standard finite-volume results. In § 6, we give a brief account of the possibilities to extend the presented method to more general cases, and finally, concluding remarks are presented in § 7.

2 Model equation and target problems

Many of the partial differential equations that are derived to model physical situations cannot be solved analytically, and are too complex to study their numerics rigorously. It is logical to first develop numerical schemes for appropriate model equations and then to carry these over to the original partial differential equations for which precise analysis is not possible. It is common practice to take model equations that are sufficiently simplified versions of the corresponding physical equations, but still resemble these equations as much as possible.

Here, we consider the one-dimensional, linear advection equation as the model equation for the Euler equations:

$$\frac{\partial c}{\partial t}+\frac{\partial f(c)}{\partial x}=0, \quad f(c)=uc. \tag{1}$$

Equation (1) is a model of scalar quantity $c(x,t)$ that is advected by the velocity u, which is constant, and which we assume to be positive. $f(c)$ is the flux function, which is linear. The independent variables x and t represent space and time, respectively. The generic domain of the solution is a one-dimensional rod, of finite length L, and the time interval, in principle, is infinitely long.

The advection equation (1) is a very simple partial differential equation, but it is an important one. It models fluid-flow equations and it proves challenging to solve it numerically. It is hyperbolic with a single set of characteristic lines. These are straight lines in the (x,t)-plane, which are determined from the solution of the ordinary differential equation:

$$\frac{\mathrm{d}x}{\mathrm{d}t}=u, \tag{2a}$$

whose integration yields the equation of the characteristic lines $x-ut=$ constant. Notice that, along a characteristic line, the dependent variable $c(x,t)$ satisfies:

$$\frac{\mathrm{d}c}{\mathrm{d}t}=\frac{\partial c}{\partial t}+\frac{\partial c}{\partial x}\frac{\mathrm{d}x}{\mathrm{d}t}\equiv\frac{\partial c}{\partial t}+u\frac{\partial c}{\partial x}=0, \tag{2b}$$

and thus, it remains constant along these lines.

Therefore, for a given initial solution $c(x,0)=c_0(x)$, the exact solution of (1), at any location x and time t, can be computed by the method of characteristics, as $c(x,t)=c_0(x-ut)$. That is, as time evolves, the initial data simply propagates unchanged with a velocity u: it propagates to the right if $u>0$ and to the left if $u<0$.

Hence, by using the exact solution as a benchmark, numerous numerical schemes can be developed and tested for the one-dimensional, linear advection equation.

2.1 Standard finite-volume discretization

In the finite-volume method, the spatial domain of the physical problem is subdivided into non-overlapping cells or control volumes. The cells are considered to be of uniform size. The domain is taken to be of unit length, $L = 1$, on the interval $x \in [0,1]$ and is divided into N cells, with the grid size being $h = \frac{1}{N}$.

A single node is located at the geometric centroid of the control volume and the cells are represented with nodal indices: $i-1$, i, $i+1$, etc. The coordinates of the nodes are determined as $x_i = (i - \frac{1}{2})h$, $i = 1,2,...,N$. Analogously, the coordinates of the cell faces are labeled by indices-with-fractions and are computed as $x_{i+\frac{1}{2}} = ih$, $i = 1,2,...,N$. Figure 1 shows the spatial domain with cells and cell faces.

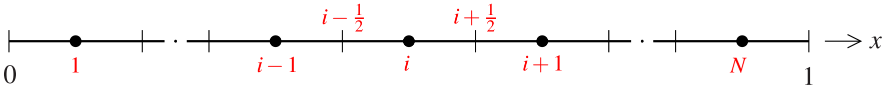


Fig. 1 One-dimensional finite-volume domain

To obtain the finite-volume model, ensuring conservation of c, the model equation is integrated over the control volumes shown in Figure 1. Integrating (1) over the volume Ω of cell i yields:

$$\int_{\Omega_i} \frac{\partial c}{\partial t} \mathrm{d}\Omega_i + \int_{\Omega_i} \frac{\partial f(c)}{\partial x} \mathrm{d}\Omega_i = 0. \tag{3}$$

We denote the discrete solution in cell i and the flux at cell face $i + \frac{1}{2}$, both at time level n, as $c_i^n := c(x_i, t^n)$ and $f_{i+\frac{1}{2}}^n := f(c(x_{i+\frac{1}{2}}, t^n))$, respectively. We assume c_i^n to be constant in space, in that cell. Applying the Gauss integration theorem, (3) can be rewritten as:

$$h \frac{\mathrm{d}c_i}{\mathrm{d}t} + (f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}) = 0. \tag{4}$$

Semi-discrete equation (4) is exact so far in cell Ω_i. It is going to be solved using the method of lines. That is, the fluxes at the cell faces are first approximated and then the temporal part is time-stepped with a suitable time-integration method.

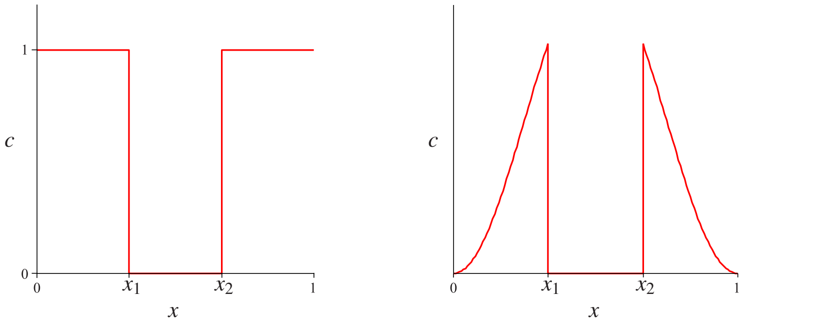
2.2 Initial and boundary conditions

Two initial solutions are considered, each with two interior, moving boundaries. The solution at the left and right of each interior boundary is prescribed. The two interior boundaries represent two infinitely thin bodies that go with the flow. The two moving boundaries have different initial locations (x_1 and x_2, $x_1 \neq x_2$). The solution is discontinuous across both interior boundaries. The two initial solutions are shown in Figure 2, and, in formulae, read:

$$c_0(x) = \begin{cases} 0, & \text{if } x_1 \leq x \leq x_2, \\ 1, & \text{elsewhere;} \end{cases} \tag{5a}$$

$$c_0(x) = \begin{cases} 0, & \text{if } x_1 \leq x \leq x_2, \\ \frac{1}{2}\left(1 - \cos(2\pi x)\right), & \text{elsewhere.} \end{cases} \tag{5b}$$

The cosine function in (5b) exploits the advantage that higher-order accurate numerical schemes have in non-constant, smooth solution regions.



(a) Constant function at peripheries

(b) Smooth (cosine) function at peripheries

Fig. 2 Initial solutions with two discontinuous interior boundaries.

The model equation is approximated in a periodic domain. That is, the first and last cell faces are ‘glued together’ and thus the fluxes in the corresponding faces are readily made equal: $f_{\frac{1}{2}} = f_{N+\frac{1}{2}}$. Apparently, periodicity allows us to time-step for as long as we want for a finite spatial domain.

Fixed-grid finite-volume methods for advection problems with interior moving boundaries are underdeveloped. No rigorous studies exist about numerical properties as accuracy and monotonicity. Here, several finite-volume methods for discontinuous moving interior-boundary problems will be derived, analyzed and tested. The moving interior-boundary conditions will be embedded in the fluxes in the direct neighborhood. The precise way in which this embedding is done is the main theme of this chapter.

2.3 Standard finite-volume schemes

Finite-volume methods distinguish themselves in the way the fluxes are computed. To start, three standard finite-volume methods are considered: first-order accurate upwind, second-order accurate central, and second-order accurate fully one-sided upwind. The latter two can be cast into one general form, the κ-scheme [9].

For positive and constant u, and an equidistant grid, the classical fluxes, at time level n, are computed as follows. The fluxes are given for cell face $i+\frac{1}{2}$ (Figure 1); for the other faces, they are computed analogously.

The general flux at cell face $i+\frac{1}{2}$, dropping the time index n, for convenience, reads:

$$f_{i+\frac{1}{2}} = u c_{i+\frac{1}{2}}, \tag{6a}$$

where $c_{i+\frac{1}{2}}$ is the cell-face state at $i+\frac{1}{2}$, which can be approximated in a variety of ways. For example, for $u > 0$, the first-order upwind flux involves only one cell and takes the form:

$$c_{i+\frac{1}{2}} = c_i. \tag{6b}$$

Equation (6b) shows that the first-order upwind flux is solely based on the information from the upstream side of the cell face.

The second-order central and fully one-sided upwind fluxes involve two cells and they take the form:

$$c_{i+\frac{1}{2}} = c_i + \frac{1}{2}(c_{i+1} - c_i), \tag{7a}$$

$$c_{i+\frac{1}{2}} = c_i + \frac{1}{2}(c_i - c_{i-1}), \tag{7b}$$

respectively. Both are written as the first-order upwind cell-face state (6b) plus a correction term. Equation (7a) is obtained by interpolation, assuming a linear variation of c between points x_i and x_{i+1}. And (7b) is obtained by extrapolation, assuming a linear variation of c between points x_{i-1} and x_i.

By blending these basic second-order accurate schemes, we can reconstruct a general higher-order accurate scheme, as:

$$c_{i+\frac{1}{2}} = \theta\left(c_i + \frac{1}{2}(c_{i+1} - c_i)\right) + (1-\theta)\left(c_i + \frac{1}{2}(c_i - c_{i-1})\right), \quad \theta \in [0,1], \tag{8a}$$

with θ the blending parameter. Formula (8a) can be rewritten as:

$$c_{i+\frac{1}{2}} = c_i + \frac{\theta}{2}(c_{i+1} - c_i) + \frac{1-\theta}{2}(c_i - c_{i-1}). \tag{8b}$$

Introducing, instead of θ, the parameter κ:

$$\kappa = 2\theta - 1, \qquad \kappa \in [-1,1], \tag{9}$$

equation (8b) turns out to be the well-known Van Leer κ-scheme [9]:

$$c_{i+\frac{1}{2}} = c_i + \frac{1+\kappa}{4}(c_{i+1} - c_i) + \frac{1-\kappa}{4}(c_i - c_{i-1}). \tag{10}$$

For $\kappa = 1$, we have the second-order accurate central scheme; and for $\kappa = -1$, we have the second-order accurate fully one-sided upwind scheme. A motivation for the blending is that for the unique value $\kappa = \frac{1}{3}$, we have $\mathcal{O}(h^3)$ net flux accuracy in each cell.

The simplicity and monotonicity of the first-order upwind scheme are appealing. However, it has strong numerical diffusion. On the other hand, the solutions of all κ-schemes, hence also those of the $\kappa = \frac{1}{3}$ scheme, may exhibit wiggles. This recalls Godunov's (1959) theorem [3] which states that there is no linear scheme higher than first-order accurate which is monotone.

We verify this here for the κ-scheme (10), considering the monotonicity requirement:

$$\frac{c_{i+\frac{1}{2}} - c_{i-\frac{1}{2}}}{c_i - c_{i-1}} \geq 0. \tag{11}$$

With the local successive solution-gradient ratios:

$$r_{i+\frac{1}{2}} = \frac{c_{i+1} - c_i}{c_i - c_{i-1}}, \tag{12a}$$

$$r_{i-\frac{1}{2}} = \frac{c_i - c_{i-1}}{c_{i-1} - c_{i-2}}, \tag{12b}$$

and κ-scheme (10), requirement (11) yields:

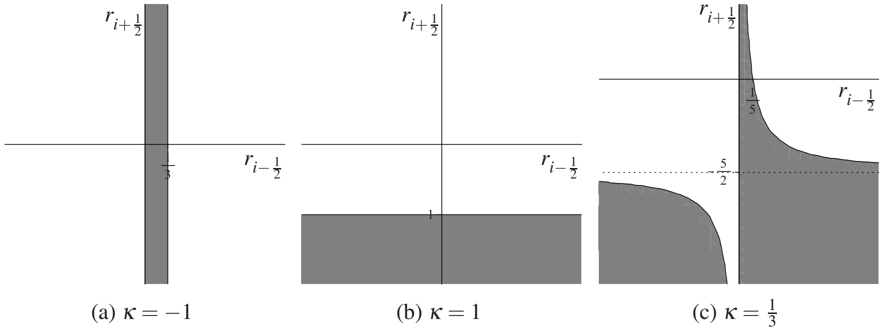
$$(1+\kappa)\, r_{i+\frac{1}{2}} - \frac{1-\kappa}{r_{i-\frac{1}{2}}} \geq 2\,(\kappa - 2)\,. \tag{13}$$

No $\kappa \in [-1,1]$ exists for which (13) is satisfied for all possible combinations of $r_{i-\frac{1}{2}}$ and $r_{i+\frac{1}{2}}$. It can be directly verified that (13) is not satisfied for $r_{i+\frac{1}{2}} < -1$ in case of $\kappa = 1$, and for $\frac{1}{r_{i-\frac{1}{2}}} > 3$ in case of $\kappa = -1$. For $\kappa = \frac{1}{3}$, requirement (13) is not satisfied for $\frac{1}{r_{i-\frac{1}{2}}} - 2\, r_{i+\frac{1}{2}} > 5$. The corresponding regions of non-monotonicity in the $(r_{i-\frac{1}{2}},\, r_{i+\frac{1}{2}})$-plane are depicted in Figure 3.

Notice that monotonicity requirement (11) is always satisfied for the first-order upwind scheme (6b).

Several algorithms have been proposed in the literature that yield higher-order accurate solutions which are free from wiggles. Most of these algorithms exploit the inherent monotonicity of the first-order upwind scheme. The best known representatives of these algorithms are the limited schemes following Sweby's work [18]. Let us consider limiters that resemble κ-schemes to the largest possible extent within Sweby's TVD domain.

With (12a), the limited form of the cell-face state according to (10) can be written as:



(a) $\kappa = -1$ (b) $\kappa = 1$ (c) $\kappa = \frac{1}{3}$

Fig. 3 Non-monotonicity regions for some κ-schemes.

$$c_{i+\frac{1}{2}} = c_i + \frac{1}{2}\phi(r_{i+\frac{1}{2}})(c_i - c_{i-1}), \tag{14a}$$

where $\phi(r)$ is the limiter function, defined as:

$$\phi(r) = \begin{cases} 0, & \text{if} \quad r < 0 \\ 2r, & \text{if} \quad 0 \le r < \frac{1-\kappa}{3-\kappa}, \\ \frac{1-\kappa}{2} + \frac{1+\kappa}{2}r, & \text{if} \quad \frac{1-\kappa}{3-\kappa} \le r < \frac{3+\kappa}{1+\kappa}, \\ 2, & \text{if} \quad \frac{3+\kappa}{1+\kappa} \le r. \end{cases} \tag{14b}$$

Here we specifically adopt the limiter proposed in [8] as the standard limiter, which gives a monotone third-order accurate net flux in a cell, by resembling the $\kappa = \frac{1}{3}$ scheme. This limiter, which is within Sweby's TVD domain, is depicted in Figure 4.

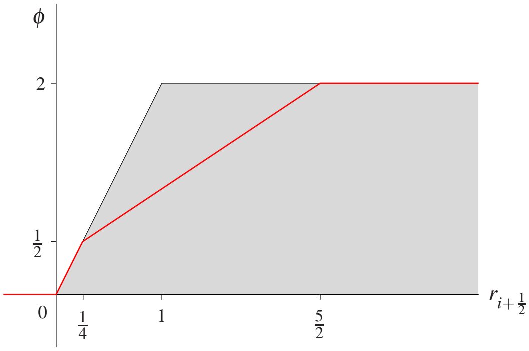
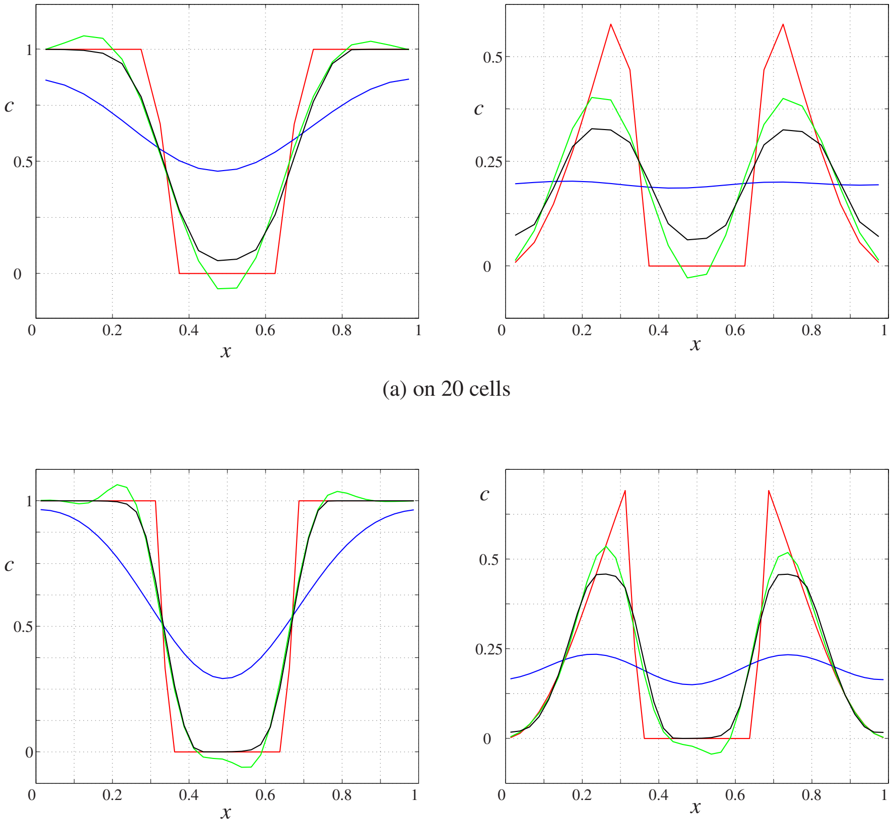


Fig. 4 Standard limiter, which is obtained from (14b) for $\kappa = \frac{1}{3}$.

In the remainder of this chapter, we will derive non-standard finite-volume methods, methods in which the interior boundary conditions are incorporated in the fixed-grid flux formulae. Before doing so, for later comparison purposes, we will show what the solutions are for the standard finite-volume discretizations described above, methods in which no embedded-boundary conditions are imposed, pure capturing methods, in fact. For the time integration, the three-stage Runge-Kutta scheme RK3b from [6] is employed. For both initial solutions given in (5a) and (5b), we consider the initial locations of the EBs to be at $x_1 = \frac{1}{3}$ and $x_2 = \frac{2}{3}$. Furthermore, we take $u = 1$, and we compute the solution at $t_{\max} = 1$, the time at which the solution has made a single full-period. For both the first-order upwind and the $\kappa = \frac{1}{3}$ (unlimited and limited) schemes, the computations are performed on a grid with 20 and 40 cells. The solutions are depicted in Figure 5. The time steps have been taken sufficiently small to ensure that in all cases the time-discretization errors are negligible with respect to the spatial discretization errors.



(a) on 20 cells

(b) on 40 cells

Fig. 5 Standard finite-volume solutions after one full-period, for the initial solutions (5a) and (5b). Red: exact discrete, blue: first-order upwind, green: unlimited higher-order upwind-biased, and black: limited higher-order upwind-biased.

3 Fluxes with embedded moving-boundary conditions

As mentioned, the sharp discontinuities of the initial solutions (5a) and (5b), shown in Figure 2, may be considered as infinitely thin bodies going with the flow and the boundary conditions associated with these may be embedded in some fixed-grid fluxes. Here, the EB (embedded boundary) conditions are user-specified and enforced to remain intact to the EB and unchanged at all times. The solution values on the left and right sides of the EB are designated as c^l_{EB} and c^r_{EB}, respectively (see Figure 6).

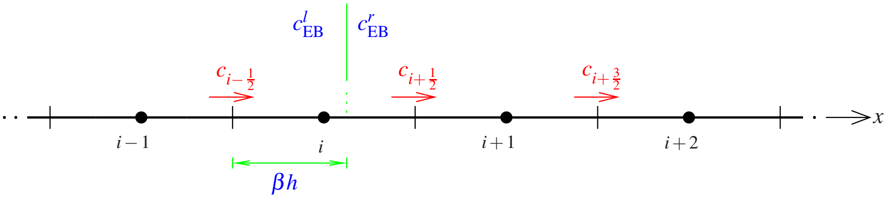


Fig. 6 EB situated in cell i at time t, its associated solution values, and the affected cell-face states.

As shown in Figure 6, for an EB situated in cell i, with its coordinate $x_{\mathrm{EB}} = x_{\mathrm{EB}}(t)$ given, its relative position with respect to $x_{i-\frac{1}{2}}$ (the left face of cell i) is βh, where $\beta \in [0,1]$ is a (non-dimensional) parameter which is defined as:

$$\beta = \frac{x_{\mathrm{EB}} - x_{i-\frac{1}{2}}}{h}. \tag{15}$$

So, $\beta = 0$ when the EB is situated at cell face $i-\frac{1}{2}$, $\beta = \frac{1}{2}$ when the EB is exactly at the centroid i, and $\beta = 1$ when the EB is at cell face $i+\frac{1}{2}$.

There is no information flow across the EB. Fluxes on one side of the EB, at a specific time t, are all computed based on the information on the same side of the EB, at that time, plus the additional interior-boundary condition on the respective side. In general, when considering three-point upwind-biased interpolation for the fluxes, three cell-face states ($c_{i-\frac{1}{2}}$, $c_{i+\frac{1}{2}}$ and $c_{i+\frac{3}{2}}$) are affected by the presence of a single EB (in cell i) and these are the cell-face states of interest for which tailor-made formulae will be derived.

In general, for an EB in a cell, the three affected cell-face states are computed such that the net fluxes in some neighboring cells are as accurate as possible. This shall be discussed in the next section. So far, it is assumed that two successive EBs are sufficiently far apart that no cell-face state exists that is affected by both EBs. Recall that all but the affected cell-face states are computed based on the standard finite-volume schemes discussed in § 2.3.

3.1 Higher-order accurate embedded-boundary fluxes

If a three-point upwind-biased interpolation is considered for computing fluxes, the cell faces $i-\frac{1}{2}$, $i+\frac{1}{2}$ and $i+\frac{3}{2}$ ‘feel’ the EB situated in cell i (see Figure 6). The higher-order accurate fluxes at these faces are computed from higher-order accurate cell-face states. In principle, all the special cell-face states are written in terms of the blending parameter κ and computed from optimally blended, three-point upwind-biased interpolation formulae. However, for cell-face state $c_{i+\frac{1}{2}}$, no upwind-biased interpolation formula can be derived as we do not draw information across the EB. Hence, there is no blending parameter in the formula for $c_{i+\frac{1}{2}}$, only non-equidistant central interpolation is applied to compute $c_{i+\frac{1}{2}}$. On the other hand, in the formulae for $c_{i-\frac{1}{2}}$ and $c_{i+\frac{3}{2}}$, there will be blending parameters, and $c_{i-\frac{1}{2}}$ and $c_{i+\frac{3}{2}}$ can be taken as optimally weighted averages of two-point central interpolation and two-point fully upwind extrapolation.

Just like away from the EB, also net cell fluxes are optimized for accuracy near the EB. The net fluxes of cells $i-1$, i, $i+1$ and $i+2$ are affected by the EB. Recalling that only $c_{i-\frac{1}{2}}$ and $c_{i+\frac{3}{2}}$ allow for optimization, only two of the four aforementioned net cell fluxes can be optimized for accuracy: either the net flux in cell $i-1$ or cell i, for $c_{i-\frac{1}{2}}$; and either the net flux in cell $i+1$ or cell $i+2$, for $c_{i+\frac{3}{2}}$.

For the accuracy optimizations, Taylor series expansions are used. Doing so, the net flux in cell i cannot be optimized due to the presence of the EB with its discontinuous solution behavior. Hence, the net flux in cell $i-1$ will be optimized for $c_{i-\frac{1}{2}}$. Secondly, for $c_{i+\frac{3}{2}}$, the net flux in cell $i+2$ will be optimized. The reason why the net flux in cell $i+2$ is optimized, instead of that of cell $i+1$, becomes clear at the end of the derivations in § 3.1.2. We start by first deriving the unlimited EB-affected cell-face states, and after that, EB-sensitive limiters will be derived.

3.1.1 Cell-face states

Here, we derive the unlimited forms of the cell-face states in cells $i-1$, $i+1$ and $i+2$. These are the EB-affected cell-face states ($c_{i-\frac{1}{2}}$, $c_{i+\frac{1}{2}}$, $c_{i+\frac{3}{2}}$) and the corresponding regular cell-face states ($c_{i-\frac{3}{2}}$, $c_{i+\frac{5}{2}}$).

a. Cell-face states affected by EB

Cell-face state $c_{i-\frac{1}{2}}$: The second-order accurate, non-equidistant, central interpolation, and the second-order accurate, equidistant, fully upwind extrapolation schemes for $c_{i-\frac{1}{2}}$ can be written as:

$$c_{i-\frac{1}{2}} = c_{i-1} + \frac{1}{1+2\beta}(c^{l}_{\text{EB}} - c_{i-1}), \tag{16a}$$

and

$$c_{i-\frac{1}{2}} = c_{i-1} + \frac{1}{2}(c_{i-1} - c_{i-2}), \tag{16b}$$

respectively. The blend of the above two schemes, is:

$$c_{i-\frac{1}{2}} = c_{i-1} + \frac{1}{1+2\beta}\frac{1+\kappa_{i-\frac{1}{2}}}{2}(c^l_{\mathrm{EB}} - c_{i-1}) + \frac{1-\kappa_{i-\frac{1}{2}}}{4}(c_{i-1} - c_{i-2}), \tag{16c}$$

with $\kappa_{i-\frac{1}{2}}$ the blending parameter. Note that we get the exact result $c_{i-\frac{1}{2}} = c^l_{\mathrm{EB}}$, $\beta = 0$, only for $\kappa_{i-\frac{1}{2}} = 1$. (The accuracy of cell-face states is not our prime interest, the accuracy of net fluxes *is*.)

Cell-face state $c_{i+\frac{1}{2}}$: As mentioned earlier, there are no sufficient number of solution points, on the upstream side of cell face $i+\frac{1}{2}$, up to and including the right face of the EB, to construct a higher-order accurate upwind-biased interpolation scheme. Hence, no κ-scheme is formulated here. Instead, this particular flux is reconstructed with only a, second-order accurate, non-equidistant central interpolation scheme, as:

$$c_{i+\frac{1}{2}} = c^r_{\mathrm{EB}} + \frac{2-2\beta}{3-2\beta}(c_{i+1} - c^r_{\mathrm{EB}}). \tag{17}$$

Note that we get the expected standard second-order accurate central result for $\beta = \frac{1}{2}$, and the exact result for $\beta = 1$.

Cell-face state $c_{i+\frac{3}{2}}$: The second-order accurate central interpolation and the non-equidistant, second-order accurate, fully upwind extrapolation schemes for $c_{i+\frac{3}{2}}$ can be written as:

$$c_{i+\frac{3}{2}} = c_{i+1} + \frac{1}{2}\left(c_{i+2} - c_{i+1}\right), \tag{18a}$$

and

$$c_{i+\frac{3}{2}} = c_{i+1} + \frac{1}{3-2\beta}\left(c_{i+1} - c^r_{\mathrm{EB}}\right), \tag{18b}$$

respectively. Blending the above two schemes, we get:

$$c_{i+\frac{3}{2}} = c_{i+1} + \frac{1+\kappa_{i+\frac{3}{2}}}{4}\left(c_{i+2} - c_{i+1}\right) + \frac{1}{3-2\beta}\frac{1-\kappa_{i+\frac{3}{2}}}{2}\left(c_{i+1} - c^r_{\mathrm{EB}}\right), \tag{18c}$$

with $\kappa_{i+\frac{3}{2}}$ being the blending parameter.

b. Corresponding regular cell-face states

For cell faces $i-\frac{3}{2}$ and $i+\frac{5}{2}$, the standard $\kappa = \frac{1}{3}$ scheme is applied:

$$c_{i-\frac{3}{2}} = c_{i-2} + \frac{1}{3}\left(c_{i-1} - c_{i-2}\right) + \frac{1}{6}\left(c_{i-2} - c_{i-3}\right), \tag{19a}$$

$$c_{i+\frac{5}{2}} = c_{i+2} + \frac{1}{3}\left(c_{i+3} - c_{i+2}\right) + \frac{1}{6}\left(c_{i+2} - c_{i+1}\right). \tag{19b}$$

3.1.2 Net cell fluxes

Here, we compute the net cell fluxes and derive the modified equations for cells $i-1$, $i+1$ and $i+2$, from which the blending parameters $\kappa_{i-\frac{1}{2}}$ and $\kappa_{i+\frac{3}{2}}$ will be optimized. Recall that the net flux in cell i cannot be optimized as the solution is discontinuous in there.

Optimal accuracy in cell $i-1$: With (16c) and (19a), we get as semi-discrete equation for cell $i-1$:

$$\frac{\mathrm{d}c_{i-1}}{\mathrm{d}t} + \frac{u}{h}\left(\frac{1}{1+2\beta}\frac{1+\kappa_{i-\frac{1}{2}}}{2}(c^{l}_{\mathrm{EB}} - c_{i-1}) + \frac{11-3\kappa_{i-\frac{1}{2}}}{12}(c_{i-1} - c_{i-2}) - \frac{1}{6}(c_{i-2} - c_{i-3})\right) = 0. \tag{20a}$$

Substituting Taylor-series expansions of c^{l}_{EB}, c_{i-2} and c_{i-3} around the point $i-1$ into (20a), we get as modified equation for cell $i-1$, ignoring the index $i-1$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{6\beta - 7 + (9+6\beta)\kappa_{i-\frac{1}{2}}}{48}uh\frac{\partial^2 c}{\partial x^2} + \frac{(3+2\beta)(2\beta-1)(1+\kappa_{i-\frac{1}{2}})}{96}uh^2\frac{\partial^3 c}{\partial x^3} = \mathscr{O}(h^3). \tag{20b}$$

Equating the leading term of the truncation error to zero, we get:

$$\kappa_{i-\frac{1}{2}} = \frac{7-6\beta}{9+6\beta}, \qquad \kappa_{i-\frac{1}{2}} \in \left[\frac{1}{15}, \frac{7}{9}\right]. \tag{21}$$

This is the $\kappa_{i-\frac{1}{2}}$ that yields the most accurate net flux in cell $i-1$. It is well within the standard κ-range $[-1,1]$. Its variation for any position of the EB within cell i is depicted in Figure 7(a).

Substituting the optimal value of $\kappa_{i-\frac{1}{2}}$ according to (21) into the modified equation (20b), we get:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{2\beta-1}{18}uh^2\frac{\partial^3 c}{\partial x^3} = \mathscr{O}(h^3). \tag{22}$$

Therefore, in general, we get a second-order (spatial) accuracy in cell $i-1$, with a maximum leading-term truncation-error coefficient of $\pm\frac{1}{18}uh^2$. Evidently, this dispersive term diminishes as the EB is in the immediate vicinity of the center of cell i, $\beta \approx \frac{1}{2}$. For $\beta = \frac{1}{2}$, $\kappa_{i-\frac{1}{2}}$ is restored as $\kappa_{i-\frac{1}{2}} = \frac{1}{3}$ (Figure 7(a)), and then we get third-order spatial accuracy.

Optimal accuracy in cell $i+1$: With (17) and (18c), we get as the semi-discrete equation for cell $i+1$:

$$\frac{\mathrm{d}c_{i+1}}{\mathrm{d}t} + \frac{u}{h}\left(\frac{3-\kappa_{i+\frac{3}{2}}}{6-4\beta}(c_{i+1}-c^r_{\mathrm{EB}}) + \frac{1+\kappa_{i+\frac{3}{2}}}{4}(c_{i+2}-c_{i+1})\right) = 0. \tag{23a}$$

Introducing Taylor-series expansions of c^r_{EB} and c_{i+2} around the point $i+1$, into (23a), we get as modified equation for cell $i+1$, ignoring the index $i+1$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{(6\beta-7)+(5-2\beta)\kappa_{i+\frac{3}{2}}}{16}uh\frac{\partial^2 c}{\partial x^2} + \frac{(12\beta^2-36\beta+31) - (4\beta^2-12\beta+5)\kappa_{i+\frac{3}{2}}}{96}uh^2\frac{\partial^3 c}{\partial x^3} = \mathscr{O}(h^3). \tag{23b}$$

Then equating the leading term of the truncation error to zero, we get:

$$\kappa_{i+\frac{3}{2}} = \frac{7-6\beta}{5-2\beta}, \qquad \kappa_{i+\frac{3}{2}} \in \left[\frac{1}{3}, \frac{7}{5}\right]. \tag{24}$$

This is the $\kappa_{i+\frac{3}{2}}$ that yields the most accurate net flux in cell $i+1$. This $\kappa_{i+\frac{3}{2}}$ is not within the standard κ-range $[-1,1]$.

Substituting (24) into (23b), we get as modified equation for cell $i+1$, ignoring the index $i+1$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{3-2\beta}{12}uh^2\frac{\partial^3 c}{\partial x^3} = \mathscr{O}(h^3). \tag{25}$$

Note that the leading order error-term in cell $i+1$ is second-order for all β; it does not vanish for $\beta = \frac{1}{2}$.

Moreover, with (18c), (19b) and (24), we get as semi-discrete equation for cell $i+2$:

$$\frac{\mathrm{d}c_{i+2}}{\mathrm{d}t} + \frac{u}{h}\left(\frac{1-2\beta}{(3-2\beta)(5-2\beta)}(c_{i+1}-c^r_{\mathrm{EB}}) + \frac{17-2\beta}{30-12\beta}(c_{i+2}-c_{i+1}) + \frac{1}{3}(c_{i+3}-c_{i+2})\right) = 0. \tag{26a}$$

Introducing Taylor-series expansions for c^r_{EB}, c_{i+1} and c_{i+3} around the point $i+2$ into (26a), we get as modified equation for cell $i+2$, ignoring the index $i+2$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{6\beta-7}{24}uh\frac{\partial^2 c}{\partial x^2} = \mathscr{O}(h^2). \tag{26b}$$

Note that the leading order error-term in cell $i+2$ is first-order for all β.

Optimal accuracy in cell $i+2$: With (18c) and (19b), we get as semi-discrete equation for cell $i+2$:

$$\frac{\mathrm{d}c_{i+2}}{\mathrm{d}t} + \frac{u}{h}\left(\frac{\kappa_{i+\frac{3}{2}}-1}{6-4\beta}(c_{i+1}-c^r_{\mathrm{EB}}) + \frac{11-3\kappa_{i+\frac{3}{2}}}{12}(c_{i+2}-c_{i+1}) + \frac{1}{3}(c_{i+3}-c_{i+2})\right) = 0. \quad (27a)$$

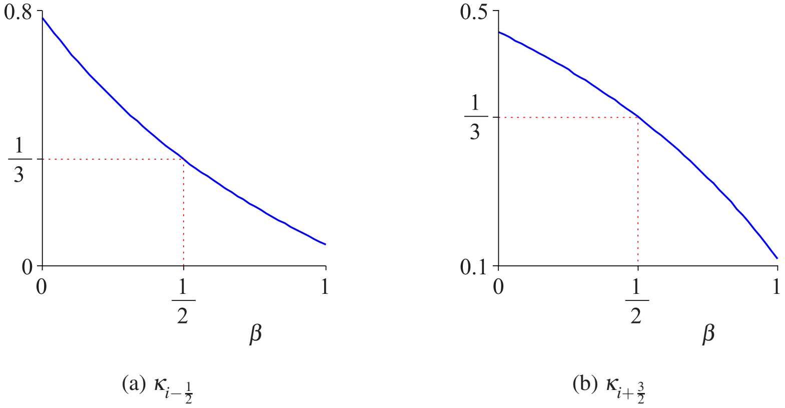
Introducing Taylor-series expansions of c^r_{EB}, c_{i+1} and c_{i+3} around the point $i+2$ into (27a), we get as modified equation for cell $i+2$, ignoring the index $i+2$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{(6\beta-15)\kappa_{i+\frac{3}{2}}+(7-6\beta)}{48}uh\frac{\partial^2 c}{\partial x^2} + \frac{(4\beta^2-24\beta+35)\kappa_{i+\frac{3}{2}} - (4\beta^2-24\beta+19)}{96}uh^2\frac{\partial^3 c}{\partial x^3} = \mathscr{O}(h^3). \quad (27b)$$

Equating the leading term of the truncation error to zero, now we get:

$$\kappa_{i+\frac{3}{2}} = \frac{7-6\beta}{15-6\beta}, \qquad \kappa_{i+\frac{3}{2}} \in \left[\frac{1}{9}, \frac{7}{15}\right]. \quad (28)$$

This is the value of $\kappa_{i+\frac{3}{2}}$ that yields the most accurate net flux in cell $i+2$. As opposed to $\kappa_{i+\frac{3}{2}}$ according to (24), this $\kappa_{i+\frac{3}{2}}$ is well within the standard κ-range $[-1,1]$. Its variation for any position of the EB within cell i is depicted in Figure 7(b).



(a) $\kappa_{i-\frac{1}{2}}$

(b) $\kappa_{i+\frac{3}{2}}$

Fig. 7 Variation of the optimal κ values for any position of the EB within cell i.

Substituting the optimal $\kappa_{i+\frac{3}{2}}$ according to (28) into (27b), we get as modified equation for cell $i+2$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{2\beta - 1}{36}uh^2\frac{\partial^3 c}{\partial x^3} = \mathscr{O}(h^3). \tag{29}$$

In contrast to the dispersive error in (25), the dispersive term in (29) does vanish as the EB gets in the vicinity of the center of cell i, $\beta \approx \frac{1}{2}$. We, thereby, get third-order spatial accuracy, and $\kappa_{i+\frac{3}{2}}$ according to (28) becomes $\kappa = \frac{1}{3}$ (see Figure 7(b), and the Appendix for a more detailed comparison).

With (17), (18c) and (28), we get as semi-discrete equation for cell $i+1$:

$$\frac{\mathrm{d}c_{i+1}}{\mathrm{d}t} + \frac{u}{h}\left(\frac{19-6\beta}{(9-6\beta)(5-2\beta)}(c_{i+1} - c^r_{\mathrm{EB}}) + \frac{11-6\beta}{30-12\beta}(c_{i+2} - c_{i+1})\right) = 0. \tag{30a}$$

And, substituting Taylor-series expansions for c^r_{EB}, and c_{i+2} around the point $i+1$, into (30a), we get as modified equation for cell $i+1$, ignoring the index $i+1$:

$$\frac{\partial c}{\partial t} + u\frac{\partial c}{\partial x} + \frac{6\beta - 7}{24}uh\frac{\partial^2 c}{\partial x^2} = \mathscr{O}(h^2). \tag{30b}$$

Equation (30b) shows that we get a first-order spatial accuracy in cell $i+1$ with a maximum leading-term truncation-error coefficient of $-\frac{7}{24}uh$. Coincidentally, (30b) is the same as (26b); the leading-order error terms in both equations are identical. The accuracy loss in the net flux of a neighboring cell is unavoidable. If the cell-face states were to be first-order accurate, i.e. $c_{i+\frac{1}{2}} = c^r_{\mathrm{EB}}$ and $c_{i+\frac{3}{2}} = c_{i+1}$, the modified equation for cell $i+1$, ignoring the index $i+1$, would become:

$$\frac{\partial c}{\partial t} + \frac{3-2\beta}{2}u\frac{\partial c}{\partial x} - \frac{(3-2\beta)^2}{8}uh\frac{\partial^2 c}{\partial x^2} = \mathscr{O}(h^2), \tag{31}$$

which, for all β, except $\beta = \frac{1}{2}$, is even zeroth-order accurate.

As the optimal $\kappa_{i+\frac{3}{2}}$ we choose (28), the one that gives the highest accuracy in cell $i+2$. In summary, the reasons why we choose this $\kappa_{i+\frac{3}{2}}$, instead of the one yielding the highest accuracy in cell $i+1$ ($\kappa_{i+\frac{3}{2}}$ according to (24)), are the following:

- For $\beta = \frac{1}{2}$, we get a third-order (spatial) accuracy in cell $i+2$ with (28) (see (29)). But with (24) we do not get this in cell $i+1$ for any β (see (25)).
  The truncation error with (28) is much less than that with (24), for any β (see Appendix).
- Noting that the solution is discontinuous across an EB, with (28) we have a dissipative leading-error term in cell $i+1$, which is the cell adjacent to cell i (where the EB is situated), and this makes the solution near the EB less prone to numerical oscillations.

With (24) however, we get the leading-error term in the same cell to be dispersive and this makes the solution near the EB to be more susceptible to numerical oscillations, numerical oscillations which may be hard to suppress because construction of a limiter for cell-face state $c_{i+\frac{1}{2}}$ is hard.

- With (28), the accuracy deterioration due to the presence of an EB in cell i is more confined to the vicinity of the EB. (We get first-order (spatial) accuracy in cell $i+1$, and second-order accuracy in cell $i+2$; whereas with (24), we get second-order accuracy in cell $i+1$, and first-order accuracy in cell $i+2$.)
- $\kappa_{i+\frac{3}{2}}$ according to (28) is well within the standard κ-range $[-1,1]$, but with (24) we get $\kappa_{i+\frac{3}{2}} \in \left[\frac{1}{3}, \frac{7}{5}\right]$.

3.1.3 Formulae for cell-face states affected by EB

Here, the formulae for all the special cell-face states that are affected by the EB, in cell i, viz. $c_{i-\frac{1}{2}}$, $c_{i+\frac{1}{2}}$ and $c_{i+\frac{3}{2}}$, are summarized.

With (16c) and (21), $c_{i-\frac{1}{2}}$ can be rewritten as:

$$c_{i-\frac{1}{2}} = c_{i-1} + \frac{8}{(3+6\beta)(3+2\beta)}(c^{l}_{\text{EB}} - c_{i-1}) + \frac{1+6\beta}{18+12\beta}(c_{i-1} - c_{i-2}). \quad (32a)$$

Further, we have:

$$c_{i+\frac{1}{2}} = c^{r}_{\text{EB}} + \frac{2-2\beta}{3-2\beta}(c_{i+1} - c^{r}_{\text{EB}}). \quad (32b)$$

And, with (18c) and (28), $c_{i+\frac{3}{2}}$ can be rewritten as:

$$c_{i+\frac{3}{2}} = c_{i+1} + \frac{11-6\beta}{30-12\beta}(c_{i+2} - c_{i+1}) + \frac{4}{(9-6\beta)(5-2\beta)}(c_{i+1} - c^{r}_{\text{EB}}). \quad (32c)$$

Verify that, in (32a) and (32c), for $\beta = \frac{1}{2}$, we get exactly the same coefficients as in the the standard $\kappa = \frac{1}{3}$ scheme (see (19a) and (19b)).

3.2 Spatial monotonicity domains and limiters

Recalling Godunov's (1959) theorem, all the linear higher-order accurate fluxes, constructed earlier, may yield wiggles. One negative aspect of wiggles is that they may cause the solution c to be negative. If c is a physical quantity that should not become negative (say, density or temperature), this may be highly undesirable. Wiggles can be avoided by carefully constraining or 'limiting' the advective fluxes calculated by the scheme. By limiting the fluxes, they may become first-order accurate in some solution regions.

A limiter is a nonlinear function that acts like a continuous control between the higher-order and first-order schemes. Obviously, limiters may reduce the overall accuracy of the scheme to some extent, albeit in non-smooth flow regions only. Limited schemes are called 'monotonicity-preserving.'

For the cell-face states that are computed by the standard $\kappa = \frac{1}{3}$ scheme (§ 2.3), the standard $\kappa = \frac{1}{3}$ limiter (14b) will be used. In this section, special limiters will be introduced for the EB-affected cell-face states $c_{i-\frac{1}{2}}$ and $c_{i+\frac{3}{2}}$ according to formulae (32a) and (32c). The cell-face state $c_{i+\frac{1}{2}}$, however, will not be limited. This shall be explained later on.

3.2.1 Spatial monotonicity domain and limiter for cell-face state $c_{i-\frac{1}{2}}$

Referring to Figure 6, for cell face $i-\frac{1}{2}$, we define the non-equidistant local successive solution-gradient ratio $\tilde{r}_{i-\frac{1}{2}}$, as:

$$\tilde{r}_{i-\frac{1}{2}} = \frac{c^l_{\mathrm{EB}} - c_{i-1}}{\frac{1+2\beta}{2}h} \Big/ \frac{c_{i-1} - c_{i-2}}{h} \equiv \frac{2}{1+2\beta}\frac{c^l_{\mathrm{EB}} - c_{i-1}}{c_{i-1} - c_{i-2}}. \tag{33}$$

Notice that for $\beta = \frac{1}{2}$, EB in the center of cell i, $\tilde{r}_{i-\frac{1}{2}}$ reduces to the standard equidistant solution-gradient ratio known from the theory of standard limiters.

We proceed by rewriting (16c) as:

$$c_{i-\frac{1}{2}} = c_{i-1} + \frac{1}{2}\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})(c_{i-1} - c_{i-2}), \tag{34a}$$

with

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) = \frac{1-\kappa_{i-\frac{1}{2}}}{2} + \frac{1+\kappa_{i-\frac{1}{2}}}{2}\tilde{r}_{i-\frac{1}{2}}. \tag{34b}$$

Substituting the optimal $\kappa_{i-\frac{1}{2}}$ according to (21) into (34b), we get:

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) = \frac{1+6\beta}{9+6\beta} + \frac{8}{9+6\beta}\tilde{r}_{i-\frac{1}{2}}. \tag{34c}$$

The family of possible $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ schemes, depending on the position of the EB within cell i ($0 \leq \beta \leq 1$), is represented in Figure 8. The function $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ will be constrained to yield a monotonicity-preserving scheme and to define the appropriate limiter for the special cell-face state $c_{i-\frac{1}{2}}$. The argument $\tilde{r}_{i-\frac{1}{2}}$ measures the local monotonicity of the solution.

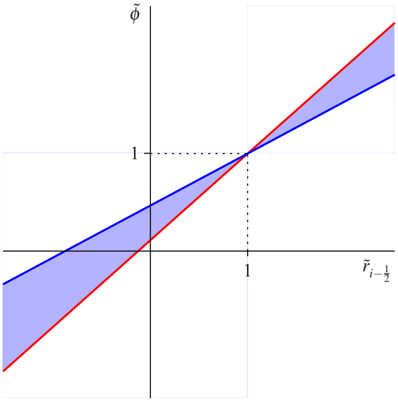


Fig. 8 Family of possible β-schemes according to (34c): the blue line is for $\beta = 1$, the red line is for $\beta = 0$, and the enclosed (colored) region is for all other $\beta \in (0,1)$.

The local solution-gradient ratio for cell face $i - \frac{3}{2}$ is defined as:

$$r_{i-\frac{3}{2}} = \frac{c_{i-1} - c_{i-2}}{c_{i-2} - c_{i-3}}. \tag{35a}$$

And, the limited form of cell-face state $c_{i-\frac{3}{2}}$ is:

$$c_{i-\frac{3}{2}} = c_{i-2} + \frac{1}{2}\phi(r_{i-\frac{3}{2}})(c_{i-2} - c_{i-3}), \tag{35b}$$

where $\phi(r)$ is standard limiter (14b) with $\kappa = \frac{1}{3}$.

The following monotonicity requirement is enforced, to constrain the function $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$:

$$\frac{c_{i-\frac{1}{2}} - c_{i-\frac{3}{2}}}{c_{i-1} - c_{i-2}} \geq 0. \tag{36a}$$

Substituting (34a) and (35b) into (36a), using (35a), we get as constraint relation:

$$1 + \frac{1}{2}\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) - \frac{1}{2}\frac{\phi(r_{i-\frac{3}{2}})}{r_{i-\frac{3}{2}}} \geq 0. \tag{36b}$$

The standard limiter already satisfies $1 - \frac{1}{2}\frac{\phi(r_{i-\frac{3}{2}})}{r_{i-\frac{3}{2}}} \geq 0, \forall r_{i-\frac{3}{2}}$; therefore, the (in)equality (36b) holds good if:

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) \geq 0, \quad \forall \tilde{r}_{i-\frac{1}{2}}. \tag{36c}$$

Moreover, enforcing the additional monotonicity requirement:

$$\frac{c^l_{\text{EB}} - c_{i-\frac{1}{2}}}{c^l_{\text{EB}} - c_{i-1}} \geq 0, \tag{37a}$$

and substituting (34a) into (37a), using (34c) and (33), we get:

$$\frac{\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})}{\tilde{r}_{i-\frac{1}{2}}} \leq 1 + 2\beta, \quad \forall \tilde{r}_{i-\frac{1}{2}}. \tag{37b}$$

The (in)equalities (36c) and (37b) define the spatial monotonicity domain for the special limiter function $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$. They delineate the left and lower bounds of the domain. The upper bound is still to be derived in § 4 from the fully discrete equation. Once we also have defined this upper bound, the respective limiter will be introduced, in § 4.1.

3.2.2 Limiter for cell-face state $c_{i+\frac{1}{2}}$

Regarding cell-face state $c_{i+\frac{1}{2}}$, a regular monotonicity argument $r_{i+\frac{1}{2}}$ can not be defined here. A regular monitor uses two solution values upstream of cell faces. In this case, since we do not want to use solution values from the other side of the EB, and therefore not c_i, we have only one upstream solution, c^r_{EB}, too little to introduce the regular smoothness monitor. Therefore, $c_{i+\frac{1}{2}}$ is not limited.

3.2.3 Spatial monotonicity domain and limiter for cell-face state $c_{i+\frac{3}{2}}$

Referring to Figure 6, the monotonicity argument $\tilde{r}_{i+\frac{3}{2}}$ is defined, as:

$$\tilde{r}_{i+\frac{3}{2}} = \frac{c_{i+2} - c_{i+1}}{h} / \frac{c_{i+1} - c^r_{\text{EB}}}{\frac{3-2\beta}{2}h} \equiv \frac{3-2\beta}{2} \frac{c_{i+2} - c_{i+1}}{c_{i+1} - c^r_{\text{EB}}}. \tag{38}$$

As expected, for $\beta = \frac{1}{2}$, $\tilde{r}_{i+\frac{3}{2}}$ according to (38) reduces to the known equidistant-grid ratio. Similar to the rewriting of expression (16c) for $c_{i-\frac{1}{2}}$, here we rewrite (18c) for $c_{i+\frac{3}{2}}$, as:

$$c_{i+\frac{3}{2}} = c_{i+1} + \frac{1}{3-2\beta} \tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})(c_{i+1} - c^r_{\text{EB}}), \tag{39a}$$

with

$$\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}}) = \frac{1-\kappa_{i+\frac{3}{2}}}{2} + \frac{1+\kappa_{i+\frac{3}{2}}}{2}\tilde{r}_{i+\frac{3}{2}}. \tag{39b}$$

Substituting the optimal $\kappa_{i+\frac{3}{2}}$ according to (28) into (39b), we get:

$$\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}}) \;=\; \frac{4}{15-6\beta} \;+\; \frac{11-6\beta}{15-6\beta}\tilde{r}_{i+\frac{3}{2}}. \tag{39c}$$

The family of possible $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$ schemes is given in Figure 9. Just as $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$, in § 3.2.1, the function $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$ will be constrained to yield a monotonicity-preserving scheme and to define the appropriate limiter for the special cell-face state $c_{i+\frac{3}{2}}$.

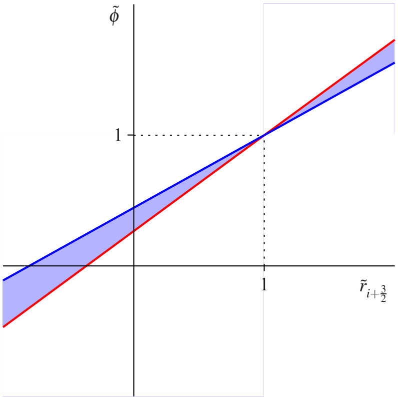


Fig. 9 Family of possible β-schemes according to (39c): the blue line is for $\beta = 1$, the red line is for $\beta = 0$, and the enclosed (colored) region is for all other $\beta \in (0,1)$.

The monotonicity argument $r_{i+\frac{5}{2}}$ is defined as:

$$r_{i+\frac{5}{2}} = \frac{c_{i+3}-c_{i+2}}{c_{i+2}-c_{i+1}}. \tag{40a}$$

And, the limited form of $c_{i+\frac{5}{2}}$ is:

$$c_{i+\frac{5}{2}} = c_{i+2} + \frac{1}{2}\phi(r_{i+\frac{5}{2}})(c_{i+2}-c_{i+1}), \tag{40b}$$

where $\phi(r_{i+\frac{5}{2}})$ is limiter (14b) with $\kappa = \frac{1}{3}$.

To constrain $\tilde{\phi}(r_{i+\frac{3}{2}})$, the following monotonicity requirements are enforced:

$$\frac{c_{i+\frac{3}{2}}-c_{i+\frac{1}{2}}}{c_{i+1}-c^{r}_{\mathrm{EB}}} \geq 0, \tag{41a}$$

$$\frac{c_{i+\frac{5}{2}}-c_{i+\frac{3}{2}}}{c_{i+2}-c_{i+1}} \geq 0. \tag{41b}$$

Substituting (17) and (39a) into (41a), we get as restriction for $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$:

$$\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}}) \geq -1, \quad \forall \tilde{r}_{i+\frac{3}{2}}. \tag{42a}$$

And, substituting (39a) and (40b) into (41b), we get:

$$\frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}} - \phi(r_{i+\frac{5}{2}}) \leq 2. \tag{42b}$$

Since the standard limiter satisfies $\phi(r_{i+\frac{5}{2}}) \geq 0, \forall r_{i+\frac{5}{2}}$, the (in)equality (42b) holds good if:

$$\frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}} \leq 2, \quad \forall \tilde{r}_{i+\frac{3}{2}}. \tag{42c}$$

The (in)equalities (42a) and (42c) define the spatial monotonicity domain for the special limiter function $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$. They delineate the left and lower bounds of the domain. Once again, after defining the upper bound of the domain in § 4.1, the respective limiter for the special cell-face state $c_{i+\frac{3}{2}}$ will be introduced. Note that as opposed to the spatial monotonicity domain for $c_{i-\frac{1}{2}}$ (see (36c) and (37b)), the monotonicity domain for $c_{i+\frac{3}{2}}$ is independent of β.

4 Temporal discretization

The semi-discrete equation (4), after substituting the appropriate discretizations for the spatial operator, is discrete in space but still continuous in time. It can be compactly written as:

$$\frac{\mathrm{d}c_i}{\mathrm{d}t} = -\frac{u}{h}(c_{i+\frac{1}{2}} - c_{i-\frac{1}{2}}) \equiv F(c), \tag{43}$$

which is an ordinary differential equation that can be discretized in time using a variety of explicit and implicit time integration methods, to get a fully discrete system of equations. Here, only two explicit schemes are considered: the Forward Euler method and the three-stage Runge-Kutta, RK3b, scheme [6]. The latter gives third-order accuracy in time.

For the Forward Euler method, (43) becomes:

$$c_i^{n+1} = c_i^n + \tau F(c^n) \equiv c_i^n - \nu(c_{i+\frac{1}{2}}^n - c_{i-\frac{1}{2}}^n), \tag{44}$$

where $\nu = u\tau/h$ is the CFL number, and τ the time step. Similarly, for the RK3b scheme, we have:

$$c_i^{n+1} = c_i^n + \frac{1}{6}(R_1 + R_2 + 4R_3), \tag{45a}$$

where the R_j's $(j = 1, 2, 3)$ are internal vectors that are computed as:

$$\begin{aligned} R_1 &= \tau F(c^n), \\ R_2 &= \tau F(c^n + R_1), \\ R_3 &= \tau F(c^n + \frac{1}{4}R_1 + \frac{1}{4}R_2). \end{aligned} \tag{45b}$$

4.1 TVD conditions and time step

The limited numerical flux conditions, as derived in § 3.2, are still insufficient to guarantee monotonicity during time integration. Harten's theorem [5] provides additional conditions that are necessary for the convergence of the fully discrete solutions to the exact, monotone solutions. These conditions define the upper bounds for the limiter functions $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ and $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$, and consequently result into more stringent restrictions on the CFL number, than the condition for stability.

The theorem in [5] states that any consistent scheme for a conservation law, written in the conservative form:

$$c_i^{n+1} = c_i^n - D^-_{i-\frac{1}{2}}(c_i^n - c_{i-1}^n) + D^+_{i+\frac{1}{2}}(c_{i+1}^n - c_i^n), \tag{46a}$$

where the D's are solution-dependent coefficients, is total-variation diminishing (TVD) if, for all i:

$$D^{\pm}_{i+\frac{1}{2}} \geq 0, \tag{46b}$$

$$D^-_{i+\frac{1}{2}} + D^+_{i+\frac{1}{2}} \leq 1. \tag{46c}$$

The total variation (TV) at time level n is defined, in discrete form, by:

$$\mathrm{TV}(c^n) = \sum_i |c_{i+1}^n - c_i^n|, \tag{47}$$

and, any scheme is said to be TVD if $\mathrm{TV}(c^{n+1}) \leq \mathrm{TV}(c^n)$.

Both conditions, (46b) and (46c), can be interpreted as positive coefficient requirements. To do so, we rewrite (46a) as

$$c_i^{n+1} = D^-_{i-\frac{1}{2}} c_{i-1}^n + (1 - D^-_{i-\frac{1}{2}} - D^+_{i+\frac{1}{2}}) c_i^n + D^+_{i+\frac{1}{2}} c_{i+1}^n. \tag{48}$$

The positive coefficient requirements for (48) are:

$$D^-_{i-\frac{1}{2}} \geq 0, \tag{49a}$$

$$1 - D^-_{i-\frac{1}{2}} - D^+_{i+\frac{1}{2}} \geq 0, \tag{49b}$$

$$D^+_{i+\frac{1}{2}} \geq 0. \tag{49c}$$

Equation (48) holds for any i, so also for $i+1$:

$$c^{n+1}_{i+1} = D^-_{i+\frac{1}{2}} c^n_i + (1 - D^-_{i+\frac{1}{2}} - D^+_{i+\frac{3}{2}}) c^n_{i+1} + D^+_{i+\frac{3}{2}} c^n_{i+2}. \tag{50}$$

The positive coefficient requirements applied to (50) yield, among others,

$$D^-_{i+\frac{1}{2}} \geq 0. \tag{51}$$

So, with (49c) and (51) we have already interpreted (46b) as a positive coefficient requirement. From (49b) it follows

$$D^-_{i-\frac{1}{2}} + D^+_{i+\frac{1}{2}} \leq 1. \tag{52}$$

Combining (52), (49a) and (49c), it may be written:

$$0 \leq D^-_{i-\frac{1}{2}} \leq 1 - \gamma, \tag{53a}$$

$$0 \leq D^+_{i+\frac{1}{2}} \leq \gamma, \tag{53b}$$

with γ some constant in the range [0,1]. We assume that the upper bound $1-\gamma$ holds for all i, hence also for $D^-_{i+\frac{1}{2}}$:

$$0 \leq D^-_{i+\frac{1}{2}} \leq 1 - \gamma. \tag{54}$$

Summation of (53b) and (54) gives

$$0 \leq D^-_{i+\frac{1}{2}} + D^+_{i+\frac{1}{2}} \leq 1. \tag{55}$$

Combined with (49c) and (51) this may be reduced to

$$D^-_{i+\frac{1}{2}} + D^+_{i+\frac{1}{2}} \leq 1, \tag{56}$$

which is TVD requirement (46c).

With this we have shown that TVD requirements (46b) and (46c) are positive coefficient requirements.

It can also be verified that condition (46b) is identical to the monotonicity requirements that we have already considered in § 3.2 (the conditions of which (36a), (37a), (41a) and (41b) are examples). Condition (46c) though has not been considered yet. It will yield the sought upper bounds for our specific limiters under construction. These bounds will be ν-dependent. Here, we will impose TVD requirement (46c) to $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ and $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$.

We consider the Forward Euler scheme and write the fully discrete equations, in the form (46a). Referring to Figure 6, cell $i-1$ is considered for $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$, and cells $i+1$ and $i+2$ are analyzed for $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$. Recall that the EB is situated in cell i and, as in all preceding sections, we will not consider this particular cell for any analysis.

4.1.1 TVD conditions for limiter function $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$

Using (34a) and (35b) and writing the fully discrete equation for cell $i-1$ in the conservative form, we get:

$$c_{i-1}^{n+1} = c_{i-1}^{n} - \frac{\nu}{2}\left(2 + \tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) - \frac{\phi(r_{i-\frac{3}{2}})}{r_{i-\frac{3}{2}}}\right)(c_{i-1}^{n} - c_{i-2}^{n}). \tag{57a}$$

Thus from (57a), we have as the corresponding coefficients:

$$D_{i-\frac{3}{2}}^{-} = \frac{\nu}{2}\left(2 + \tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) - \frac{\phi(r_{i-\frac{3}{2}})}{r_{i-\frac{3}{2}}}\right), \tag{57b}$$

$$D_{i-\frac{1}{2}}^{+} = 0. \tag{57c}$$

Enforcing condition (46c), we get:

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) - \frac{\phi(r_{i-\frac{3}{2}})}{r_{i-\frac{3}{2}}} \leq \frac{2}{\nu} - 2. \tag{58a}$$

Because the standard limiter satisfies $0 \leq \frac{\phi(r_{i-\frac{3}{2}})}{r_{i-\frac{3}{2}}} \leq 2$, $\forall r_{i-\frac{3}{2}}$, the above inequality reduces to:

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) \leq \frac{2}{\nu} - 2, \quad \forall \tilde{r}_{i-\frac{1}{2}}. \tag{58b}$$

Taking the (in)equalities (36c), (37b) and (58b) into account, the TVD domain of the special limiter for cell-face state $c_{i-\frac{1}{2}}$ is graphically illustrated in Figure 10.

(In)equality (58b) confirms that the upper bound of the special limiter function $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ depends on the CFL number ν. The upper bound increases when lowering ν. Note that the choice $\nu = 1$, the stability bound for Forward Euler,

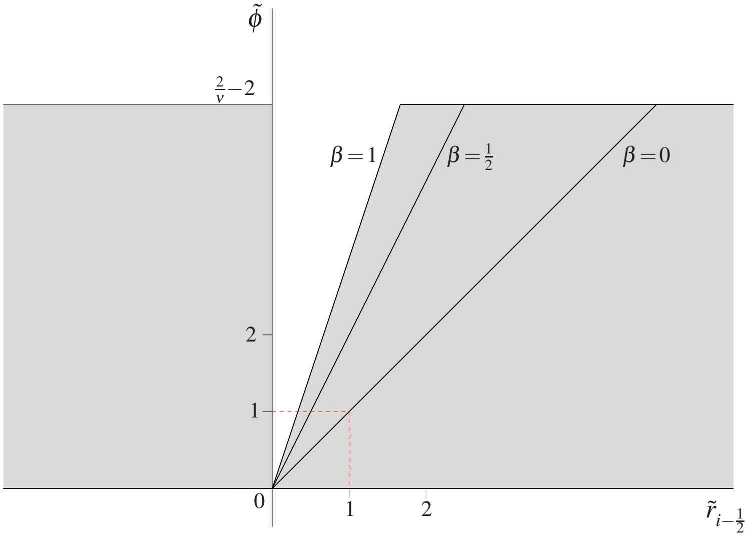


Fig. 10 TVD domain for cell-face state $c_{i-\frac{1}{2}}$, for some characteristic values of β. Note that $\nu \leq \frac{1}{2}$.

yields $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) = 0$. Hence, with $\nu = 1$, the second-order accuracy requirement $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}} = 1) = 1$ cannot be satisfied for this limiter. $\nu \leq \frac{2}{3}$ allows to meet this accuracy requirement. $\nu < \frac{2}{3}$ even allows for third-order accuracy in space. We take $\nu = \frac{1}{2}$ as the upper bound; it gives sufficient room for good spatial accuracy and does not bound the time step too much. Moreover, it will appear to be the hard upper bound for ν in using limiter $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$.

When we impose this bound for ν on $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$, the ν-dependence of the upper bound in (58b) is avoided; the (in)equality (58b) can then be simplified, $\forall \tilde{r}_{i-\frac{1}{2}}$, to the more practical inequality:

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) \leq 2, \qquad \text{for } \nu \leq \frac{1}{2}. \tag{58c}$$

With this, the TVD domain in Figure 10 simplifies to that given in Figure 11.

We now strive for a practical limiter $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ which coincides with the target scheme (34) to the maximal possible extent. An algorithm for computing this limiter $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ reads:

1. Compute β according to (15).
2. Compute the actual value of the monotonicity argument $\tilde{r}_{i-\frac{1}{2}}$ according to (33).
3. Compute the values $\tilde{r}^{*}_{i-\frac{1}{2}}$ and $\tilde{r}^{**}_{i-\frac{1}{2}}$ for which the target function $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ according to (34c) intersects the simplified TVD domain's bounds, $(1+2\beta)\tilde{r}_{i-\frac{1}{2}}$ and 2, respectively.

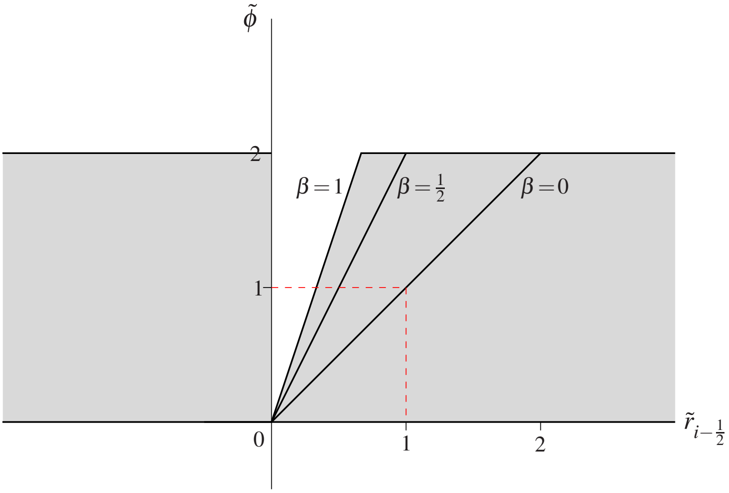


Fig. 11 Simplified TVD domain for cell-face state $c_{i-\frac{1}{2}}$, for some characteristic values of β.

$$\tilde{r}^{*}_{i-\frac{1}{2}} = \frac{1+6\beta}{1+24\beta+12\beta^2}, \tag{59a}$$

$$\tilde{r}^{**}_{i-\frac{1}{2}} = \frac{17+6\beta}{8}. \tag{59b}$$

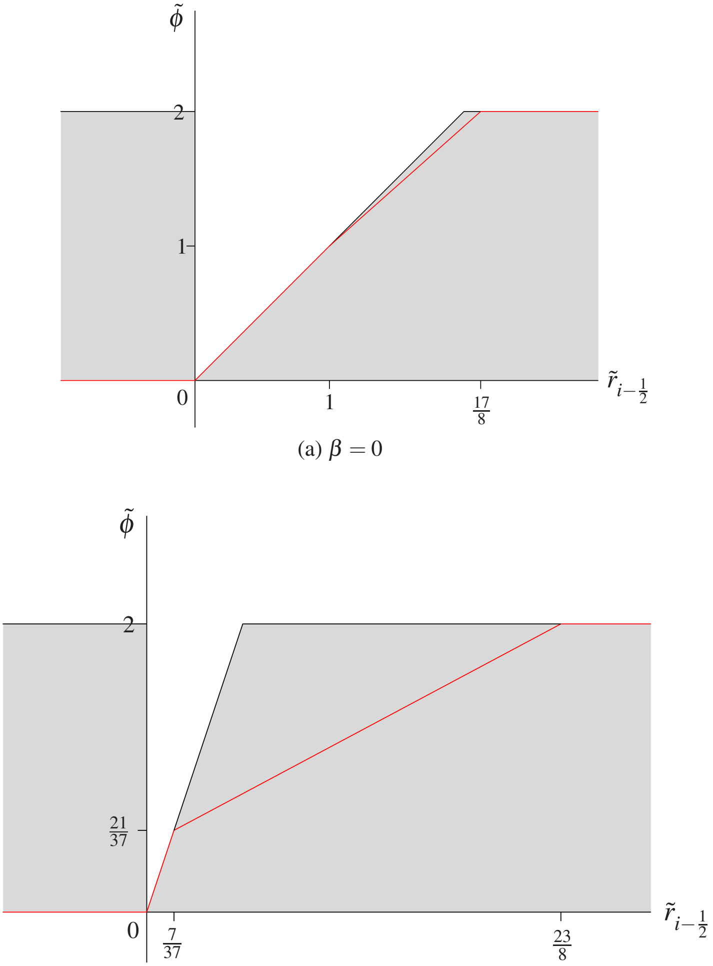
4. Then, the special limiter for the cell-face state $c_{i-\frac{1}{2}}$ reads:

$$\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}}) = \begin{cases} 0, & \text{if } \tilde{r}_{i-\frac{1}{2}} < 0, \\ (1+2\beta)\tilde{r}_{i-\frac{1}{2}}, & \text{if } 0 \le \tilde{r}_{i-\frac{1}{2}} < \tilde{r}^{*}_{i-\frac{1}{2}}, \\ \frac{1+6\beta}{9+6\beta} + \frac{8}{9+6\beta}\tilde{r}_{i-\frac{1}{2}}, & \text{if } \tilde{r}^{*}_{i-\frac{1}{2}} \le \tilde{r}_{i-\frac{1}{2}} < \tilde{r}^{**}_{i-\frac{1}{2}}, \\ 2, & \text{if } \tilde{r}_{i-\frac{1}{2}} \ge \tilde{r}^{**}_{i-\frac{1}{2}}. \end{cases} \tag{60}$$

The EB-sensitive limiter (60) is depicted in Figure 12 for the two extreme values of β, $\beta = 0$ and $\beta = 1$, together with the corresponding TVD domains.

4.1.2 TVD conditions for limiter function $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$

Similarly, to fully constrain the limiter function $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$, the fully discrete equations for cells $i+1$ and $i+2$ are analyzed. Substituting (32b), (39a) and (40b) into (46a), rewritten for c^{n+1}_{i+1} and c^{n+1}_{i+2}, the fully discrete equations can be written as:



(a) $\beta = 0$

(b) $\beta = 1$

Fig. 12 Special EB-sensitive limiters (in red) for the cell-face state $c_{i-\frac{1}{2}}$ and the corresponding TVD domains for the two extreme values of β.

$$c_{i+1}^{n+1} = c_{i+1}^{n} - \frac{\nu}{3-2\beta}\left(1+\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})\right)(c_{i+1}^{n} - c_{\mathrm{EB}}^{r}), \tag{61a}$$

$$c_{i+2}^{n+1} = c_{i+2}^{n} - \frac{\nu}{2}\left(2+\phi(r_{i+\frac{5}{2}}) - \frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}}\right)(c_{i+2}^{n} - c_{i+1}^{n}). \tag{61b}$$

Thus from (61a) and (61b), we have as the corresponding Harten coefficients, for cells $i+1$ and $i+2$:

$$D^-_{i+\frac{1}{2}} = \frac{\nu}{3-2\beta}\left(1 + \tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})\right), \tag{62a}$$

$$D^+_{i+\frac{3}{2}} = 0, \tag{62b}$$

and

$$D^-_{i+\frac{3}{2}} = \frac{\nu}{2}\left(2 + \phi(r_{i+\frac{5}{2}}) - \frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}}\right), \tag{62c}$$

$$D^+_{i+\frac{5}{2}} = 0, \tag{62d}$$

respectively. Enforcing condition (46c), we get:

$$\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}}) \leq \frac{3-2\beta}{\nu} - 1, \quad \forall \tilde{r}_{i+\frac{3}{2}}, \tag{63a}$$

and

$$\phi(r_{i+\frac{5}{2}}) - \frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}} \leq \frac{2}{\nu} - 2. \tag{63b}$$

The standard limiter satisfies $0 \leq \phi(r_{i+\frac{5}{2}}) \leq 2$, $\forall r_{i+\frac{5}{2}}$, and therefore (in)equality (63b) reduces to:

$$\frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}} \geq 4 - \frac{2}{\nu}, \quad \forall \tilde{r}_{i+\frac{3}{2}}. \tag{63c}$$

Taking the (in)equalities (42a), (42c), (63a) and (63c) into account, the TVD domain of the special limiter for cell-face state $c_{i+\frac{3}{2}}$ is depicted in Figure 13.

Concerning the just derived ν-dependent bounds, $\frac{3-2\beta}{\nu} - 1$ and $4 - \frac{2}{\nu}$, we notice that here $\nu = \frac{1}{2}$ is the maximum value that still allows to meet the second-order accuracy requirement $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}} = 1) = 1$, for $\beta = 1$. Hence, $\nu \leq \frac{1}{2}$ is the CFL restriction for the fully discrete systems (61a) and (61b) to be TVD and second-order accurate.

For $\nu \leq \frac{1}{2}$, the (in)equalities (63a) and (63c) can be simplified to:

$$\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}}) \leq 5 - 4\beta, \quad \forall \tilde{r}_{i+\frac{3}{2}}, \tag{63d}$$

and

$$\frac{\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})}{\tilde{r}_{i+\frac{3}{2}}} \geq 0, \quad \forall \tilde{r}_{i+\frac{3}{2}}. \tag{63e}$$

Doing so, the TVD domain in Figure 13 simplifies to the one shown in Figure 14.

In analogy to the algorithm for $\tilde{\phi}(\tilde{r}_{i-\frac{1}{2}})$ given in § 4.1.1, here an algorithm is also given for limiter $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$:

1. Compute β according to (15).
2. Compute the actual value of the monotonicity argument $\tilde{r}_{i+\frac{3}{2}}$ according to (38).

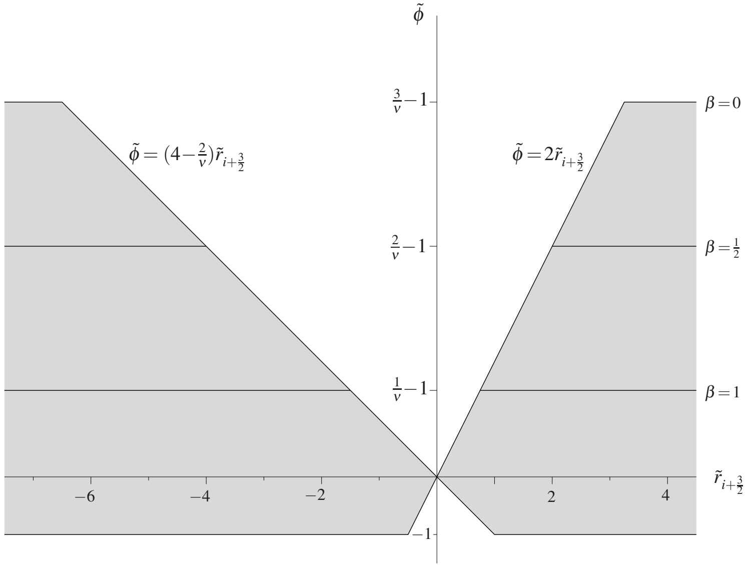


Fig. 13 TVD domain (drawn into scale for $\nu = 0.4$) for cell-face state $c_{i+\frac{3}{2}}$, for some characteristic values of β.

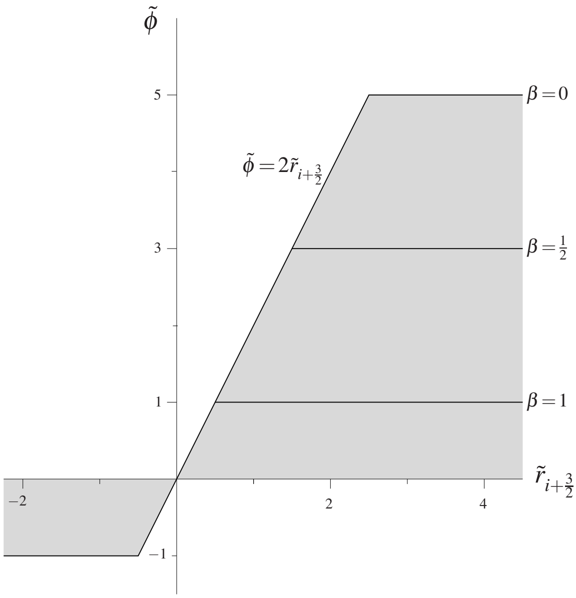


Fig. 14 Simplified TVD domain for cell-face state $c_{i+\frac{3}{2}}$, for some characteristic values of β.

3. Compute the values $\tilde{r}^{*}_{i+\frac{3}{2}}$, $\tilde{r}^{**}_{i+\frac{3}{2}}$, $\tilde{r}^{***}_{i+\frac{3}{2}}$ and $\tilde{r}^{****}_{i+\frac{3}{2}}$ for which the target function $\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}})$ according to (39c) equals -1 and 0, and intersects the bounds $2\tilde{r}_{i+\frac{3}{2}}$ and $5-4\beta$ of the simplified TVD domain (Figure 14), respectively:

$$\tilde{r}^{*}_{i+\frac{3}{2}} = -\frac{19-6\beta}{11-6\beta}, \tag{64a}$$

$$\tilde{r}^{**}_{i+\frac{3}{2}} = -\frac{4}{11-6\beta}, \tag{64b}$$

$$\tilde{r}^{***}_{i+\frac{3}{2}} = \frac{4}{19-6\beta}, \tag{64c}$$

$$\tilde{r}^{****}_{i+\frac{3}{2}} = \frac{71-90\beta+24\beta^2}{11-6\beta}. \tag{64d}$$

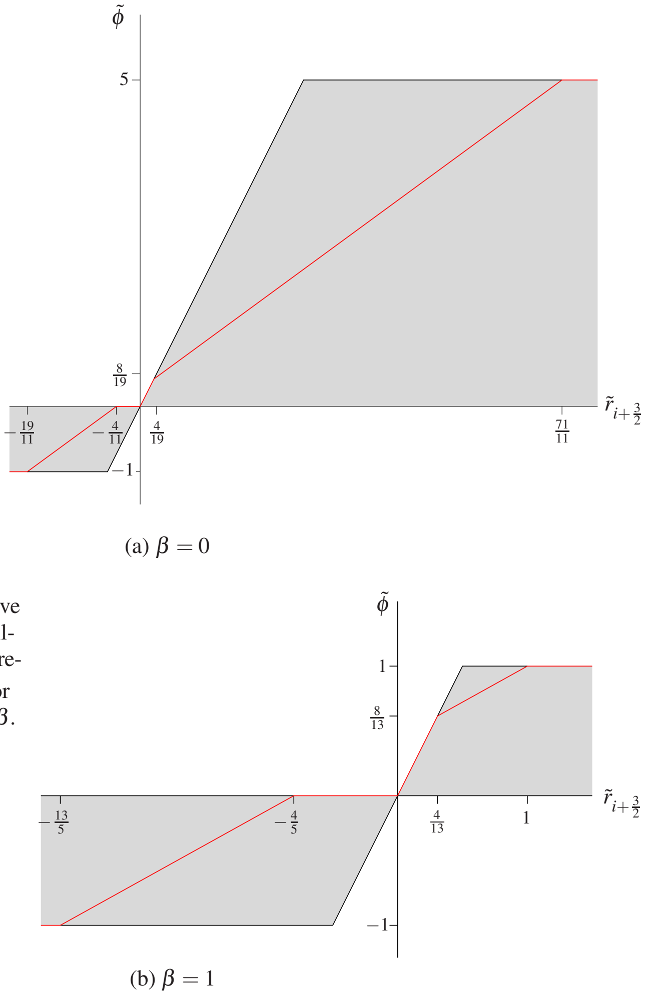
4. Then, the special limiter for the cell-face state $c_{i+\frac{3}{2}}$ reads:

$$\tilde{\phi}(\tilde{r}_{i+\frac{3}{2}}) = \begin{cases} -1, & \text{if } \tilde{r}_{i+\frac{3}{2}} < \tilde{r}^{*}_{i+\frac{3}{2}}, \\ \frac{4}{15-6\beta} + \frac{11-6\beta}{15-6\beta}\tilde{r}_{i+\frac{3}{2}}, & \text{if } \tilde{r}^{*}_{i+\frac{3}{2}} \le \tilde{r}_{i+\frac{3}{2}} < \tilde{r}^{**}_{i+\frac{3}{2}}, \\ 0, & \text{if } \tilde{r}^{**}_{i+\frac{3}{2}} \le \tilde{r}_{i+\frac{3}{2}} < 0, \\ 2\tilde{r}_{i+\frac{3}{2}}, & \text{if } 0 \le \tilde{r}_{i+\frac{3}{2}} < \tilde{r}^{***}_{i+\frac{3}{2}}, \\ \frac{4}{15-6\beta} + \frac{11-6\beta}{15-6\beta}\tilde{r}_{i+\frac{3}{2}}, & \text{if } \tilde{r}^{***}_{i+\frac{3}{2}} \le \tilde{r}_{i+\frac{3}{2}} < \tilde{r}^{****}_{i+\frac{3}{2}}, \\ 5-4\beta, & \text{if } \tilde{r}_{i+\frac{3}{2}} \ge \tilde{r}^{****}_{i+\frac{3}{2}}. \end{cases} \tag{65}$$

In Figure 15, we give the limiter (65) for the two extreme values of β, $\beta = 0$ and $\beta = 1$, with the corresponding TVD domains.

4.2 Local adaptivity in time

Consider the stencil in the (x,t)-plane in Figure 16. The EB is situated in cell i at t^n in such a way that it migrates to the next cell $i+1$ at some time in between t^n and t^{n+1}. Apparently, the solutions c^n_i and c^n_{i+1} are updated, in Forward Euler, using the modified cell-face states $c^n_{i-\frac{1}{2}}$, $c^n_{i+\frac{1}{2}}$ and $c^n_{i+\frac{3}{2}}$. However, as the EB crosses the cell face at $x_{i+\frac{1}{2}}$, there is an abrupt change in the state at this face. Before the crossing, the state at this cell face must be computed based on the data to the right of the EB; whereas, after the crossing, it must be computed based on the (different) data to the left of the EB. The two solutions c^{n+1}_i and c^{n+1}_{i+1}, which are affected by the flux across this particular cell face, need to 'feel' this reversal, i.e., the abrupt change in $c_{i+\frac{1}{2}}$.



(a) $\beta = 0$

(b) $\beta = 1$

Fig. 15 Special EB-sensitive limiters (in red) for the cell-face state $c_{i+\frac{3}{2}}$ and the corresponding TVD domains for the two extreme values of β.

Referring to Figure 16, the time adaptivity or splitting is carried out in the following manner:

1. Compute β at t^n according to (15) and next the cell-face state $c^n_{i+\frac{1}{2}}$ according to (32b).
2. Calculate the time fraction α at which the EB crosses the cell face at $x_{i+\frac{1}{2}}$, i.e:

$$\alpha = \frac{x_{i+\frac{1}{2}} + \varepsilon - x^n_{EB}}{u\tau}, \qquad \alpha \in (0,1), \tag{66}$$

where x^n_{EB} is the location of the EB at time level n. Note that the EB is placed at infinitesimal distance ε off $x_{i+\frac{1}{2}}$, in the direction of the flow.

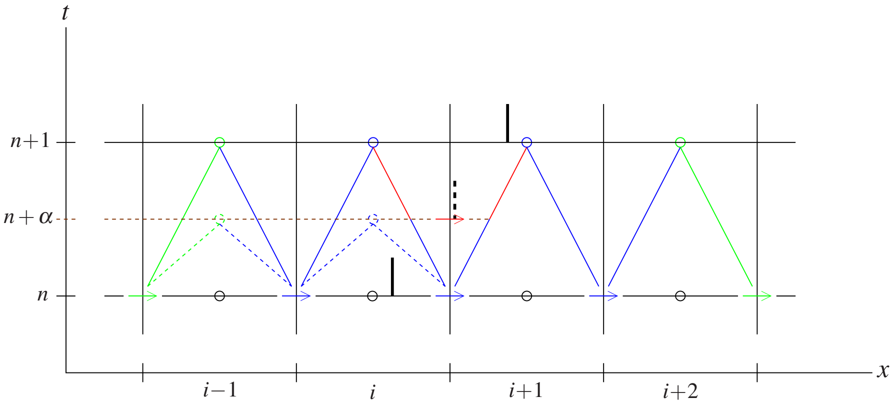


Fig. 16 Stencil for local adaptivity in time. The standard, modified and the intermediate cell-face states are designated in green, blue, and red, respectively.

3. Update the solution values c_{i-1}^{n} and c_{i}^{n} to time level $n+\alpha$, i.e., compute $c_{i-1}^{n+\alpha}$ and $c_{i}^{n+\alpha}$.
4. Compute the intermediate cell-face state $c_{i+\frac{1}{2}}^{n+\alpha}$ according to (32a), with all indices in (32a) increased with 1, using $\beta = 0$ (formally $\beta = \varepsilon/h$), $c_{i-1}^{n+\alpha}$ and $c_{i}^{n+\alpha}$.
5. Take the weighted average of $c_{i+\frac{1}{2}}^{n}$ and $c_{i+\frac{1}{2}}^{n+\alpha}$, and recompute the time-adapted cell-face state at $x_{i+\frac{1}{2}}$, as:

$$c_{i+\frac{1}{2}}^{n} := \alpha c_{i+\frac{1}{2}}^{n} + (1-\alpha) c_{i+\frac{1}{2}}^{n+\alpha}. \tag{67}$$

6. Use the time-adapted cell-face state $c_{i+\frac{1}{2}}^{n}$ and continue updating the solution everywhere with the regular time step τ.

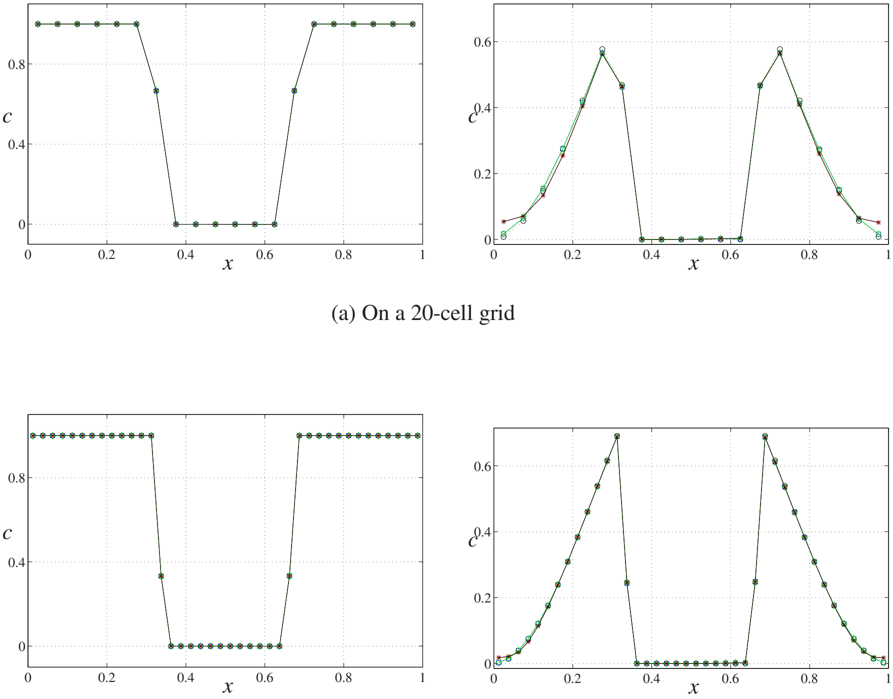
Besides the above approach, in which only the jumping cell-face state $c_{i+\frac{1}{2}}$ is recomputed at $t^{n+\alpha}$, spatially more elaborate ways of doing the time adaptivity might be investigated. For instance, all cell-face states that stop or start to be affected by the EB, viz. $c_{i-\frac{1}{2}}$, $c_{i+\frac{1}{2}}$, $c_{i+\frac{3}{2}}$ and $c_{i+\frac{5}{2}}$, might be recomputed at $t^{n+\alpha}$. Or even, the cell-face states of all cells that start or stop to feel the EB might be recomputed, i.e., $c_{i-\frac{3}{2}}$, $c_{i-\frac{1}{2}}$, $c_{i+\frac{1}{2}}$, $c_{i+\frac{3}{2}}$, $c_{i+\frac{5}{2}}$ and $c_{i+\frac{7}{2}}$. However, the gain we achieve in accuracy, as we consider more intermediate cell-face states than only $c_{i+\frac{1}{2}}$, is marginal for the given cost increase. As expected, recomputation of only the jumping cell-face state $c_{i+\frac{1}{2}}$ is necessary and sufficient for significantly improving the solution accuracy.

For RK3b, we do not yet resort to the temporal local-adaptivity procedure devised, for Forward Euler, above. We instead split the regular time step τ into smaller time steps, depending on the number of EBs crossing cell faces, and update the intermediate solutions everywhere. For instance, for a single EB crossing a cell face, we divide τ into two smaller time steps $\alpha\tau$ and $(1-\alpha)\tau$.

5 Numerical examples

Numerical results are given to validate the immersed-boundary approach presented in this work. We take the same data as in § 2, i.e., the initial solutions (5); initial EB locations $x_1 = \frac{1}{3}$ and $x_2 = \frac{2}{3}$; flow speed $u = 1$; and final time $t_{\max} = 1$. Further, we consider again a grid of 20 and 40 cells.

The results obtained, shown in Figure 17, are remarkably accurate. The results show a significant improvement in resolution, without much computational overhead, over those computed using the standard methods, Figure 5. For the more discriminating initial solution (5b) (the cosine with cavity), the numerical results of the limited higher-order upwind-biased schemes are slightly deficient at the peripheries. This is due to the property of standard limiters that they clip physically correct extrema. Apparently, the deficiency becomes smaller with decreasing mesh width.

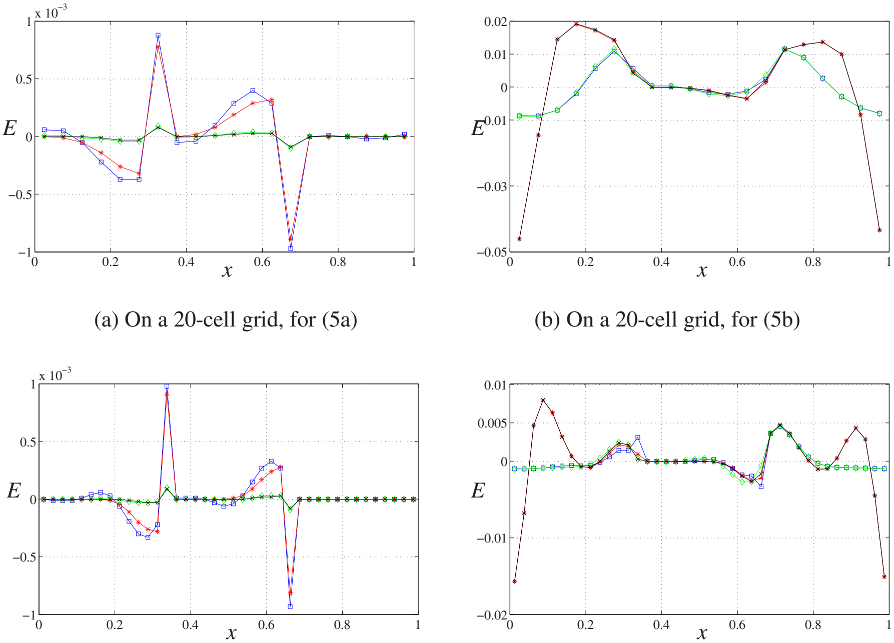


(a) On a 20-cell grid

(b) On a 40-cell grid

Fig. 17 Immersed-boundary solutions after one full-period, for the initial solutions (5). ○: exact discrete, □: unlimited higher-order upwind-biased with Forward Euler, $*$: limited higher-order upwind-biased with Forward Euler, ◇: unlimited higher-order upwind-biased with RK3b, ×: limited higher-order upwind-biased with RK3b.

In Figure 18, we show the error E, which is computed as the difference between the exact and numerical solutions, for the solutions given in Figure 17. As can be seen, there is relatively more discrepancy near the EBs. This near-EB discrepancy is of the same order for both test cases. However, as mentioned earlier, the discrepancy is significantly larger, about five times more, in the cosine-with-cavity case at and near the extrema, due to the limiters. In the former case, the results obtained with the RK3b scheme are superior to those obtained with the Forward Euler scheme, for the obvious reason. In the latter case, both schemes, limited and/or unlimited, yield almost the same accuracy.



(a) On a 20-cell grid, for (5a)

(b) On a 20-cell grid, for (5b)

(c) On a 40-cell grid, for (5a)

(d) On a 40-cell grid, for (5b)

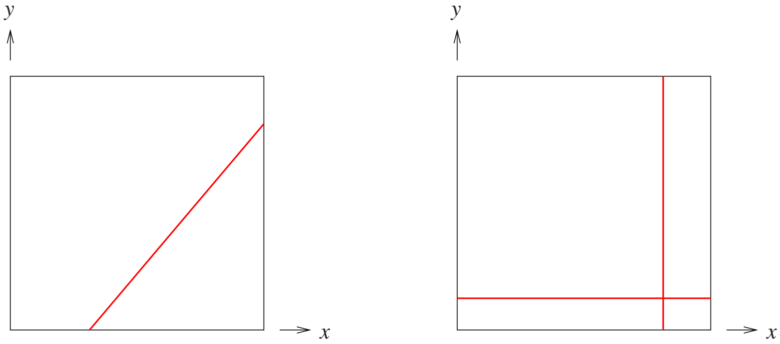
Fig. 18 Errors after one full-period, for the initial solutions (5). □-blue: unlimited higher-order upwind-biased with Forward Euler, ∗-red: limited higher-order upwind-biased with Forward Euler, ◇-green: unlimited higher-order upwind-biased with RK3b, ×-black: limited higher-order upwind-biased with RK3b.

6 Extension to more general cases

First extensions to the method presented so far, which must and will be made, are: (i) to higher dimensions and (ii) to higher-order accuracy in time. We already performed some work into this direction. Here follows a brief account of the ideas we are currently pursuing.

6.1 Extension to higher dimensions

Extension to 2D and 3D of the current 1D space discretization is done by dimensional splitting. For this purpose, a multi-D embedded boundary is to be projected first on each of its separate coordinate directions (Figure 19). The details of the projection step are crucial. Dimensional splitting has been applied with success to the standard κ-scheme and, as such, is widely spread in CFD. We presume that it will be successful here as well.



(a) control volume with EB (in red)

(b) 1D projections of the EB

Fig. 19 Example of 1D projections of a 2D embedded boundary.

6.2 Higher-order accuracy in time

Forward Euler can readily be made second-order accurate by following the Modified Euler approach. Given (43), Modified Euler reads:

$$\hat{c}_i^{n+1} = c_i^n + \tau F(c^n), \tag{68a}$$

$$c_i^{n+1} = c_i^n + \tfrac{1}{2}\tau\left(F(\hat{c}^{n+1}) + F(c^n)\right). \tag{68b}$$

Forward Euler step (68a) is the predictor and (68b) the corrector step. Modified Euler is still explicit. Extension of the local adaptivity in time, introduced in § 4.2, from Forward Euler to Modified Euler is rather straightforward, given the close similarity of the two schemes. Details about our local time adaptivity method and Modified Euler will be given in future work.

7 Conclusion

A novel immersed-boundary approach, for solving advection problems, has been introduced. The essence of the approach is that moving bodies are embedded in a regular fixed grid and specific fluxes in the vicinity of the embedded boundary (EB) are computed in such a way that they accurately and monotonously accommodate the boundary conditions valid on the moving body. To suppress the wiggles that exhibit near discontinuities, tailor-made limiters are introduced for the fluxes that are especially modified. Then, over the majority of the domain, where we do not have influence of the embedded boundaries, we can readily use standard methods on the underlying regular fixed grid. Excellent results are achieved, without much computational overhead.

In summary:

- A generalized κ-scheme that uses EB information (EB location and EB solution values), and which is an optimally accurate upwind-biased finite-volume discretization, has been proposed. This near-EB spatial discretization is a generalization of a well-proven finite-volume discretization, which allows us to accommodate EBs.
- Generalized limiters that use EB information have been proposed. These limiters satisfy the spatial monotonicity requirement and Harten's TVD requirement. To be consistent with standard limiters, the generalized limiters are made independent of the CFL number.
- Locally adaptive splitting of the time step, near EBs, has been proposed; a two-stage approach which requires the least computational time and memory for a given gain in accuracy.

We foresee that the numerical methods, developed so far and still to be developed, can readily be extended to realistic flow problems.

Acknowledgements The first author's research is funded by the Delft Centre for Computational Science and Engineering (DCSE).

Appendix

Referring to (25) and (29), the local truncation error terms e in cells $i+1$ and $i+2$, when the net fluxes in cells $i+1$ and $i+2$ are considered to optimize $\kappa_{i+\frac{3}{2}}$, respectively, are:

$$e_{i+1} = \frac{3-2\beta}{12} uh^2 \frac{\partial^3 c}{\partial x^3}, \tag{69a}$$

and

$$e_{i+2} = \frac{2\beta-1}{36} uh^2 \frac{\partial^3 c}{\partial x^3}. \tag{69b}$$

For a given grid size h, (69) can be rewritten, as a function of $\beta \in [0,1]$, as:

$$\frac{e_{i+1}}{b} = 9 - 6\beta, \tag{70a}$$

and

$$\frac{e_{i+2}}{b} = 2\beta - 1, \tag{70b}$$

where

$$b = \frac{uh^2}{36}\frac{\partial^3 c}{\partial x^3}. \tag{70c}$$

The scaled error terms (70a) and (70b) are plotted in the $(e/b,\, \beta)$-diagram given in Figure 20. We see that $|\, e_{i+1} \,| \;\; \geq \; 3 \,|\, e_{i+2} \,|, \forall\beta$. Also notice that $e_{i+2} = 0$ for $\beta = \frac{1}{2}$; i.e., we get a third-order accurate net flux in cell $i+2$ when the EB is situated at the center of cell i.

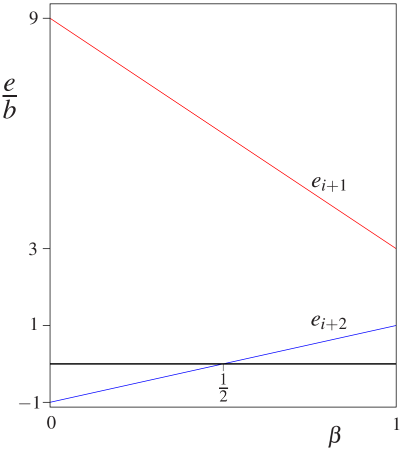


Fig. 20 Variation of the scaled, leading local truncation error terms in cells $i+1$ and $i+2$ with β.

References

1. Calhoun, D.: A Cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions. Journal of Computational Physics **176**(2), 231–275 (2002)
2. Fadlun, E.A., Verzicco, R., Orlandi, P., Mohd-Yusof, J.: Combined immersed-boundary methods for three-dimensional complex flow simulations. Journal of Computational Physics **161**, 35–60 (2000)
3. Godunov, S.K.: Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. Matematicheskii Sbornik **44**, 271–306 (1959). Translated from Russian at the Cornell Aeron. Lab.
4. Goldstein, D., Handler, R., Sirovich, L.: Modeling a no-slip flow boundary with an external force field. Journal of Computational Physics **105**, 354–366 (1993)

5. Harten, A.: On a class of high resolution total-variation-stable finite-difference schemes. SIAM Journal on Numerical Analysis **21**, 1–23 (1984)
6. Hundsdorfer, W., Koren, B., van Loon, M., Verwer, J.G.: A positive finite-difference advection scheme. Journal of Computational Physics **117**, 35–46 (1995)
7. Kim, J., Kim, D., Choi, H.: An immersed-boundary finite-volume method for simulations of flow in complex geometries. Journal of Computational Physics **171**, 132–150 (2001)
8. Koren, B.: A robust upwind finite-volume method for advection, diffusion and source terms. In: Vreugdenhil, C.B., Koren, B. (eds.) Notes on Numerical Fluid Mechanics, **45**, pp. 117-138. Vieweg, Braunschweig (1993)
9. Leer, B. van: Upwind-difference methods for aerodynamic problems governed by the Euler equations. In: Lectures in Applied Mathematics, **22 - 2**, pp. 327-336. American Mathematical Society, Providence, RI (1985)
10. Mittal, R., Iaccarino, G.: Immersed boundary methods. Annual Review of Fluid Mechanics **37**, 239–261 (2005)
11. Mohd-Yusof, J.: Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries. In: CTR Annual Research Briefs, pp. 317-327. Center for Turbulence Research, NASA Ames/Stanford University (1997)
12. Mohd-Yusof, J.: Development of immersed boundary methods for complex geometries. In: CTR Annual Research Briefs, pp. 325-336. Center for Turbulence Research, NASA Ames/Stanford University (1998)
13. Peskin, C.S.: Flow patterns around heart valves: a numerical method. Journal of Computational Physics **10**, 252–271 (1972)
14. Peskin, C.S.: Numerical analysis of blood flow in the heart. Journal of Computational Physics **25**, 220–252 (1977)
15. Peskin, C.S.: The fluid dynamics of heart valves: experimental, theoretical and computational methods. Annual Review of Fluid Mechanics **14**, 235–259 (1982)
16. Saiki, E.M., Biringen, S.: Numerical simulation of a cylinder in uniform flow: application of virtual boundary method. Journal of Computational Physics **123**, 450–465 (1996)
17. Su, S.-W., Lai, M.-C., Lin, C.-A.: An immersed boundary technique for simulating complex flows with rigid boundary. Computers & Fluids **36**, 313–324 (2007)
18. Sweby, P.: High resolution schemes using flux limiters for hyperbolic conservation laws. SIAM Journal on Numerical Analysis **21**, 995–1011 (1984)
19. Tseng, Y.H., Ferziger, J.H.: A ghost-cell immersed boundary method for flow in complex geometry. Journal of Computational Physics **192**, 593–623 (2003)
20. Zhang, N, Zheng, Z.C.: An improved direct-forcing immersed-boundary method for finite difference applications. Journal of Computational Physics **221**, 250–268 (2007)