

## A SIMPLIFIED $h$ -BOX METHOD FOR EMBEDDED BOUNDARY GRIDS\*

MARSHA BERGER<sup>†</sup> AND CHRISTIANE HELZEL<sup>‡</sup>

**Abstract.** We present a simplified  $h$ -box method for integrating time-dependent conservation laws on embedded boundary grids using an explicit finite volume scheme. By using a method of lines approach with a strong stability preserving Runge–Kutta method in time, the complexity of our previously introduced  $h$ -box method is greatly reduced. A stable, accurate, and conservative approximation is obtained by constructing a finite volume method where the numerical fluxes satisfy a certain cancellation property. For a model problem in one space dimension using appropriate limiting strategies, the resulting method is shown to be total variation diminishing. In two space dimensions, stability is maintained by using rotated  $h$ -boxes as introduced in previous work [M. J. Berger and R. J. LeVeque, *Comput. Systems Engrg.*, 1 (1990), pp. 305–311; C. Helzel, M. J. Berger, and R. J. LeVeque, *SIAM J. Sci. Comput.*, 26 (2005), pp. 785–809], but in the new formulation,  $h$ -box gradients are taken solely from the underlying Cartesian grid, which also reduces the computational cost.

**Key words.** Cartesian grid cut cell method, finite volume, conservation laws

**AMS subject classifications.** 65M08, 65M20, 65M12, 35L63

**DOI.** 10.1137/110829398

**1. Introduction.** When computing inviscid flow using an explicit finite volume method, the time step  $\Delta t$  is typically proportional to the cell volume. However, for embedded boundary grids with cut cells adjacent to the boundary, the cut cell volumes can be orders of magnitude smaller than the regular Cartesian grid cell volume. The use of standard difference procedures would lead to an unacceptably small integration time step. Both accuracy and stability are issues that need to be addressed at these highly irregular cut cells adjacent to solid bodies.

Various methods have been proposed in the literature for dealing with the *small cell* problem. The most intuitive approach is to use cell merging, where a cut cell is combined with one or more neighbors until a cell of sufficient volume is attained. In two space dimensions this approach has been tried in [2, 7, 10, 25, 33]. While conceptually simple, in practice this method is hard to implement. There are many special cases that make it particularly exasperating in three dimensions, and it is difficult even for problems in two dimensions involving moving geometry, where the merging procedure must be invoked at every time step.

A particularly simple approach is flux redistribution, due to Chern and Colella; see [6, 8, 24]. In essence, flux redistribution stabilizes a finite volume scheme by allowing only a fraction of a cell's flux balance at each time step to update the value in the small cell, where the fraction is proportional to the cell's volume. To maintain conservation the rest of the flux balance is distributed to the cut cell's neighbors. This method

---

\*Submitted to the journal's Methods and Algorithms for Scientific Computing section April 4, 2011; accepted for publication (in revised form) January 4, 2012; published electronically March 27, 2012. This work was supported in part by the DOE office of Advanced Scientific Computing under grant DE-FG02-88ER25053, by AFOSR grant FA9550-10-1-0158, and by the DFG through FOR 1048.

<http://www.siam.org/journals/sisc/34-2/82939.html>

<sup>†</sup>Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012 (berger@cims.nyu.edu).

<sup>‡</sup>Department of Mathematics, Ruhr-University Bochum, Universitätsstr. 150, 44780 Bochum, Germany (christiane.helzel@rub.de).

has been implemented in three dimensions for general conservation laws. However, results in [8] indicate that the method is only first order at the cut cells, although this is sufficient to retain second order accuracy globally if using a second order scheme in the interior of the domain. A similar approach has been explored by Klein, Bates, and Nikiforakis [19] in the context of well-balanced atmospheric flow simulations.

Other approaches that have been proposed in the literature include interpolation-based procedures, such as the mirror-cell method by Forrer and Jeltsch [11], and a related ghost-fluid method by Dadone and Grossman [9]. There are also approaches based on finite difference schemes [20, 29, 30] and kinetic schemes [18]. The use of implicit schemes would be another way to stabilize cut cells, but there is a lot of overhead associated with it that may not be warranted unless the method being used to advance the solution in the interior of the domain is also implicit. A partially implicit peer method for atmospheric flow simulations has recently been proposed by Jebens, Knoth, and Weiner [17].

In previous work [3, 16] we introduced an  $h$ -box method in both one and two space dimensions and derived some of its accuracy and stability properties. This was mostly done in the context of the wave propagation algorithm [21, 22] in [3] and a predictor-corrector MUSCL-type scheme [16]. We showed that the method was second order accurate in space and time and was Gustafsson–Kreiss–Sundström (GKS)-stable [13] in one space dimension for arbitrarily small cell sizes.

In this paper we will develop the simplified  $h$ -box method in the context of a method of lines, the source of much of the simplification. We start in section 2 with an overview of the original  $h$ -box method in one space dimension and review some of its properties, since this work builds on it. In section 3 we describe the new  $h$ -box method, also in one space dimension, which uses a method of lines discretization with a strong stability preserving Runge–Kutta integration in time. Such methods guarantee that the nonlinear stability properties satisfied by the spatial discretization when coupled with forward Euler in time will be preserved when a higher order integration in time is used. In section 3 we show using a model problem that by extending the limiting strategy applied to the gradient in a cut cell to take into account more neighbors, a total variation diminishing (TVD) method ensues. In section 4 we review the two-dimensional finite volume method used for the regular part of the grid. We also describe a rotated grid finite volume method in this context. Section 5 presents a two-dimensional rotated grid  $h$ -box method for the cut cells at the embedded boundary, which is again based on the method of lines approach. Computational results of two-dimensional problems are presented in section 6.

**2. One-dimensional  $h$ -box method.** In this and the next section, we consider the scalar one-dimensional conservation law

$$(2.1) \quad q_t(x, t) + f(q(x, t))_x = 0, \quad -\infty < x < \infty,$$

with initial values  $q(x, 0) = q_0(x)$ . Here  $q : \mathbb{R} \times \mathbb{R}^+ \rightarrow \mathbb{R}$  is the conserved quantity, and  $f : \mathbb{R} \rightarrow \mathbb{R}$  is the flux function.

We present and analyze the  $h$ -box method in the context of a model problem in one space dimension where the mesh is uniform of mesh width  $h$  except for one small cell in the middle with mesh width  $\alpha h$ ,  $0 < \alpha \leq 1$  (see Figure 2.1). The small cell  $k$  “represents” the cut cells in an embedded boundary mesh in two space dimensions.

A typical finite volume scheme has the semidiscrete form

$$(2.2) \quad \frac{d}{dt} Q_j(t) = -\frac{1}{h_j} (\mathcal{F}(Q_j(t), Q_{j+1}(t)) - \mathcal{F}(Q_{j-1}(t), Q_j(t)))$$

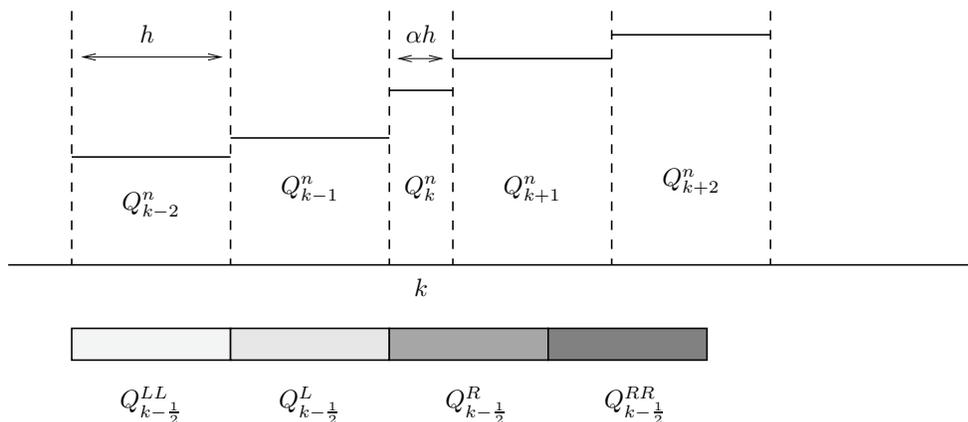


FIG. 2.1. The model problem in one space dimension has one small cell in the middle of the domain with mesh width  $\alpha h$ ,  $0 \leq \alpha \leq 1$ . The boxes below the axis indicate the  $h$ -boxes used to compute the flux at the left interface of the small cell.

with  $h_j = h$  for all cells except the  $k$ th, where  $h_k = \alpha h$ . Typically the flux  $F_{j+1/2} = \mathcal{F}(Q_j, Q_{j+1})$  depends only on the neighboring cells for a first order method, or for a second order method, on the gradient in the cell as well, which can bring in additional neighboring values. For the small cell  $k$  simply replacing  $h_j$  with  $\alpha h$  in (2.2) will be unstable for time steps larger than  $\alpha h/s_k$ , where  $s_k$  is the maximal wave speed of waves at the interfaces of the  $k$ th grid cell. A cut cell approach, however, should be stable for time steps which are restricted by a CFL number for the regular part of the mesh. This requires a special treatment at cut cells.

The main idea of the  $h$ -box method is to extend the domain of dependence in the flux calculations in a particular way that eliminates the dependence on  $\alpha$  in the denominator. The fluxes must satisfy a certain *cancellation* property for this to happen, so that  $F_{k+1/2} - F_{k-1/2} = O(\alpha h)$ . Instead of (2.2) we use a method of the form

$$(2.3) \quad \frac{d}{dt} Q_k(t) = -\frac{1}{\alpha h} (\mathcal{F}(Q_{k+1/2}^L, Q_{k+1/2}^R) - \mathcal{F}(Q_{k-1/2}^L(t), Q_{k-1/2}^R(t))),$$

where the states  $Q^L, Q^R$  are averages over boxes of length  $h$  extending to the left and right from the small cell's interfaces.

These boxes are indicated at the bottom of Figure 2.1 for the cell edge to the left of the small cell. They can be thought of as a *virtual* grid, although solution values are associated only with the actual Cartesian grid, and the  $h$ -box values are calculated on-the-fly.

Examine one of the boxes in particular, say  $Q_{k-1/2}^R$ , the right box at the interface to the left of the small cell illustrated in Figure 2.2. The values  $Q_k$  and  $Q_{k+1}$  represent the cell averages, and the dotted line is a piecewise linear reconstruction in cell  $k+1$ . We can define the solution value associated with the box as the integral average over a box of length  $h$ ,

$$(2.4) \quad \begin{aligned} Q_{k-1/2}^R &= \int_{x_{k-1/2}}^{x_{k-1/2}+h} Q(x) dx \\ &= \alpha Q_k + (1 - \alpha)(Q_{k+1} + (\bar{x} - x_{k+1})\nabla Q_{k+1}). \end{aligned}$$

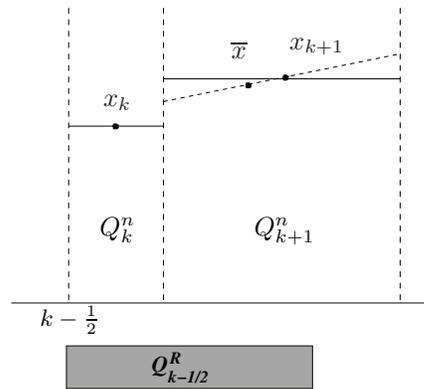


FIG. 2.2. Notation for definition of  $h$ -box  $Q_{k-1/2}^R$ .

The point  $\bar{x}$  is the centroid of the portion of the integral overlapping cell  $k + 1$ , so that evaluating the linear reconstruction in this box at this point gives this part of the integral. For the first order scheme, where the gradient in the Cartesian cell  $k + 1$  is set to  $\nabla Q_{k+1} = 0$ , this reduces to

$$(2.5) \quad \begin{aligned} Q_{k-1/2}^R &= \alpha Q_k + (1 - \alpha) Q_{k+1}, \\ Q_{k+1/2}^L &= \alpha Q_k + (1 - \alpha) Q_{k-1}. \end{aligned}$$

At the other interfaces the boxes overlap exactly with a mesh cell, and the  $h$ -box value is simply the cell average for the regular cells; i.e., this is the standard first order finite volume scheme.

For the higher order scheme, if the gradient is calculated using a backward difference  $\nabla Q_{k+1} = \frac{Q_{k+1} - Q_k}{x_{k+1} - x_k}$ , the resulting formulas are

$$(2.6) \quad \begin{aligned} Q_{k-1/2}^R &= \frac{2\alpha Q_k + (1 - \alpha) Q_{k+1}}{1 + \alpha}, \\ Q_{k+1/2}^L &= \frac{2\alpha Q_k + (1 - \alpha) Q_{k-1}}{1 + \alpha}. \end{aligned}$$

Note that  $\nabla Q_k$  does not play a role since the entire cell average of  $Q_k$  is included in the integral, and no reconstruction is necessary.

In the case of linear advection  $q_t + q_x = 0$  with explicit Euler time discretization and the first order upwind flux (i.e.,  $\mathcal{F}(Q_k, Q_{k+1}) = Q_k$ ), the use of (2.5) gives the small cell update

$$\begin{aligned} Q_k^{n+1} &= Q_k^n - \frac{\Delta t}{\alpha h} (Q_{k+1/2}^L - Q_{k-1/2}^L) \\ &= Q_k^n - \frac{\Delta t}{\alpha h} (\alpha Q_k + (1 - \alpha) Q_{k-1} - Q_{k-1}) \\ &= Q_k^n - \frac{\Delta t}{\alpha h} (\alpha(Q_k - Q_{k-1})) \\ &= Q_k^n - \frac{\Delta t}{h} (Q_k - Q_{k-1}). \end{aligned}$$

The  $\alpha$  in the denominator of the cell update has disappeared. The local truncation error of this formula would appear to be inconsistent (since  $Q_k$  and  $Q_{k-1}$  are not a

distance  $h$  apart), but was shown to be first order in [3] using the Wendroff and White technique for showing supraconvergence [32].

Similarly, using the more accurate linear interpolation formula (2.6) gives

$$\begin{aligned} Q_k^{n+1} &= Q_k^n - \frac{\Delta t}{\alpha h} (Q_{k+1/2}^L - Q_{k-1/2}^L) \\ &= Q_k^n - \frac{\Delta t}{\alpha h} \left( \frac{2\alpha}{(1+\alpha)} Q_k + \frac{(1-\alpha)}{(1+\alpha)} Q_{k-1} - Q_{k-1} \right) \\ &= Q_k^n - \frac{\Delta t}{\frac{(1+\alpha)}{2} h} (Q_k - Q_{k-1}), \end{aligned}$$

so again the  $\alpha$  in the denominator has been mitigated. This is clearly first order, since, using Taylor series,  $(Q_k - Q_{k-1})/(1+\alpha)h/2 = Q_x + O(h)$ . If the  $h$ -box formulas (2.6) are used with the second order Lax–Wendroff flux function, the resulting method has been shown to be second order in [3]. Furthermore, we have also shown that the scheme with the small cell is stable as  $\alpha \rightarrow 0$  according to the theory of Gustafsson, Kreiss, and Sundström [13]. Intuitively, stability is due to the fact that the boxes for the left and right flux calculations overlap each other except for a volume of size  $\alpha h$ , leading to the flux cancellation property mentioned earlier.

Here, we will always compute  $h$ -box values by integrating a piecewise linear function over a box of length  $h$ . We will compute gradients using a least squares approach, and use a slope limiter to guarantee that the total variation of the reconstructed function is limited by the total variation of the cell average values.

**3.  $h$ -box methods using the method of lines.** In this work we separate the discretization in space and time. A method of lines approach is used to obtain second order accuracy in time. We concentrate on using  $h$ -boxes to achieve second order accuracy and stability in space.

Let  $Q(t)$  be the vector with  $i$ th component an approximation to the value  $q(x_i, t)$  at the spatial position  $x_i$ . After introducing a spatial discretization, we obtain a system of ODEs of the form

$$Q_t = \mathcal{L}(Q),$$

where  $\mathcal{L}(Q) \approx -f(q)_x$  denotes the discretization of the flux difference, evaluated at the discrete locations. The spatial discretization will be chosen such that a forward Euler step satisfies

$$(3.1) \quad \|Q^n + \Delta t \mathcal{L}(Q^n)\|_{TV} \leq \|Q^n\|_{TV},$$

where the total variation is defined as

$$\|Q\|_{TV} = \sum_j |Q_{j+1} - Q_j|.$$

Instead of using the forward Euler method for the time integration, we will use a second (or higher) order strong stability preserving (SSP) Runge–Kutta method. For such methods

$$\|Q^{n+1}\|_{TV} \leq \|Q^n\|_{TV}$$

follows from (3.1) under the time step restriction  $0 \leq \Delta t \leq c\Delta t_E$ , where  $\Delta t_E$  is the time step restriction of the explicit Euler method and  $c \leq 1$  is a constant which depends on the Runge–Kutta method; see [12].

Here we use the second order accurate Runge–Kutta method from [28]:

$$(3.2) \quad \begin{aligned} Q^{(1)} &= Q^n + \Delta t \mathcal{L}(Q^n), \\ Q^{n+1} &= \frac{1}{2} \left( Q^n + Q^{(1)} + \Delta t \mathcal{L}(Q^{(1)}) \right). \end{aligned}$$

It is easy to see that the method (3.2) is TVD provided that (3.1) is satisfied. The time integration (3.2) can be replaced by any other SSP Runge–Kutta method. Each stage of such a Runge–Kutta method is a convex combination of explicit Euler methods, and the TVD property can be obtained if the time step is small enough.

**3.1. TVD stability for the uniform grid case.** As a first step in specifying the spatial discretization we review the case of an equispaced mesh with mesh width  $h$  and cell centers  $x_i$ . At each grid cell interface  $x_{i+\frac{1}{2}}$ , we specify numerical fluxes by first reconstructing values of the conserved quantity at the left and right sides of each cell interface,

$$(3.3) \quad \begin{aligned} Q_{i+\frac{1}{2}}^- &= Q_i + \frac{h}{2} \nabla Q_i, \\ Q_{i+\frac{1}{2}}^+ &= Q_{i+1} - \frac{h}{2} \nabla Q_{i+1}, \end{aligned}$$

where  $\nabla Q_i$  is a limited gradient of the conserved quantity in the grid cell  $i$ . A typical choice is the monotized central-difference (MC) limiter or the minmod limiter (see, e.g., [22]):

$$(3.4) \quad \nabla Q_i^{MC} = \text{minmod} \left( \frac{Q_{i+1} - Q_{i-1}}{x_{i+1} - x_{i-1}}, 2 \frac{Q_i - Q_{i-1}}{x_i - x_{i-1}}, 2 \frac{Q_{i+1} - Q_i}{x_{i+1} - x_i} \right),$$

$$(3.5) \quad \nabla Q_i^{\text{minmod}} = \text{minmod} \left( \frac{Q_i - Q_{i-1}}{x_i - x_{i-1}}, \frac{Q_{i+1} - Q_i}{x_{i+1} - x_i} \right).$$

A geometric interpretation of the MC limiter applied to a cell-centered gradient is that the gradient is reduced so that when evaluating a linearly reconstructed solution at the edge of a cell, the value does not exceed the neighboring cell-centered value. For minmod, the gradient is limited so that when the solution is reconstructed to the neighboring cell center, there are no new extrema. We will use these interpretations on the nonuniform grid.

Let  $\mathcal{F}$  be a Lipschitz continuous and consistent numerical flux function. Then we can define the  $i$ th component of the operator  $\mathcal{L}(Q)$  as

$$(3.6) \quad \mathcal{L}(Q)_i = -\frac{1}{h} \left( \mathcal{F}(Q_{i+\frac{1}{2}}^-, Q_{i+\frac{1}{2}}^+) - \mathcal{F}(Q_{i-\frac{1}{2}}^-, Q_{i-\frac{1}{2}}^+) \right).$$

For our stability analysis we consider the advection equation  $f(q) = uq$  with positive advection speed  $u$ . In this case the numerical flux function has the form

$$\mathcal{F} \left( Q_{i+\frac{1}{2}}^-, Q_{i+\frac{1}{2}}^+ \right) = u Q_{i+\frac{1}{2}}^-.$$

PROPOSITION 3.1. *The finite volume method*

$$Q_i^{n+1} = Q_i^n - \nu \left( Q_{i+\frac{1}{2}}^- - Q_{i-\frac{1}{2}}^- \right)$$

with the reconstructed values (3.3) using the MC slope (3.4) is TVD under the time step restriction  $\nu = u\Delta t/h \leq 0.5$ .

This is easily seen as a simple application of Harten’s theorem [14], which we repeat here in preparation for extending it to the case of an  $h$ -box method on a nonuniform mesh. Harten’s theorem states that a method of the form

$$Q_i^{n+1} = Q_i^n - C_{i-1}^n(Q_i^n - Q_{i-1}^n) + D_i^n(Q_{i+1}^n - Q_i^n)$$

satisfies  $\|Q^{n+1}\|_{TV} \leq \|Q^n\|_{TV}$ , provided that

$$(3.7) \quad C_{i-1}^n \geq 0,$$

$$(3.8) \quad D_i^n \geq 0,$$

$$(3.9) \quad C_i^n + D_i^n \leq 1$$

hold for all  $i$ .

On a uniform grid with mesh width  $h$ , this can be achieved as follows:

$$(3.10) \quad Q_i^{n+1} = Q_i^n - \nu \left( Q_i^n + \frac{h}{2} \nabla Q_i^n - Q_{i-1}^n - \frac{h}{2} \nabla Q_{i-1}^n \right)$$

$$(3.11) \quad = Q_i^n - \nu \left( Q_i^n - Q_{i-1}^n + \frac{h}{2} \nabla Q_i^n - \frac{h}{2} \nabla Q_{i-1}^n \right)$$

$$(3.12) \quad = Q_i^n - \nu \left( 1 + \frac{\nabla Q_i^n}{\frac{2}{h}(Q_i^n - Q_{i-1}^n)} - \frac{\nabla Q_{i-1}^n}{\frac{2}{h}(Q_i^n - Q_{i-1}^n)} \right) (Q_i^n - Q_{i-1}^n).$$

We can set  $C_{i-1}$  to be  $\nu$  times the term in the large parenthesis, and  $D_i = 0$ . Using  $\nu \leq 0.5$  means we need the sum of the three terms in the parentheses to be  $\leq 2$ , so that (3.9) is satisfied. Looking at the case where the solution is increasing (the other case can be done analogously), the MC slope limiter requires the gradient to satisfy

$$(3.13) \quad Q_i + \frac{h}{2} \nabla Q_i^n \leq Q_{i+1}^n,$$

$$(3.14) \quad Q_i - \frac{h}{2} \nabla Q_j^n \geq Q_{i-1}^n.$$

This leads to  $0 \leq C_{i-1} \leq 1$  in (3.12). If the solution is an extremum, the gradients are set to 0, so again  $0 \leq C_{i-1} \leq 1$ .  $\square$

If we use the second order time integration (3.2), we obtain the TVD result under the time step restriction of the explicit Euler method, i.e.,  $\nu \leq 0.5$ .

*Remark 3.1.* Note that although the finite volume method with the forward Euler scheme in time is TVD with a time step of  $u\Delta t/h \leq 0.5$ , the linear scheme, obtained by omitting the limiting step for the gradients, is linearly unstable as can be seen using a straightforward Fourier analysis. This is true whether we use central, upwind, or downwind differences to compute the gradients.

The finite volume method with the second order Runge–Kutta method (3.2) in time (which is used in our simulations) is linearly stable and has a stability limit of  $u\Delta t/h \leq 1$  for centered gradients and  $u\Delta t/h \leq 0.5$  for one-sided differences in the upwind direction. The latter matches the TVD limit obtained by the SSP theory.

**3.2. TVD stability for the nonuniform grid case.** We now consider the approximation of the advection equation on a grid of the form shown in Figure 2.1. There is one grid cell with width  $\alpha h$ ,  $0 < \alpha \leq 1$ , while all other grid cells have the width  $h$ . The goal is to construct a finite volume method that is second order accurate for smooth solutions and TVD for time steps satisfying the time step condition  $u\Delta t/h \leq 0.5$  independently of the size of the small grid cell.

To achieve this we construct values of the conserved quantity which are assigned to a box of length  $h$  at the left and right of each cell interface. At the interfaces of the small grid cell  $k$  we define these values as

$$(3.15) \quad Q_{k+\frac{1}{2}}^L := \alpha Q_k + (1 - \alpha) \left( Q_{k-1} + \frac{\alpha h}{2} \nabla Q_{k-1} \right),$$

$$(3.16) \quad Q_{k+\frac{1}{2}}^R := Q_{k+1},$$

$$(3.17) \quad Q_{k-\frac{1}{2}}^L := Q_{k-1},$$

$$(3.18) \quad Q_{k-\frac{1}{2}}^R := \alpha Q_k + (1 - \alpha) \left( Q_{k+1} - \frac{\alpha h}{2} \nabla Q_{k+1} \right).$$

Furthermore we assign slopes to each  $h$ -box,

$$(3.19) \quad \nabla Q_{k+\frac{1}{2}}^L := \alpha \nabla Q_k + (1 - \alpha) \nabla Q_{k-1},$$

$$(3.20) \quad \nabla Q_{k+\frac{1}{2}}^R := \nabla Q_{k+1},$$

$$(3.21) \quad \nabla Q_{k-\frac{1}{2}}^L := \nabla Q_{k-1},$$

$$(3.22) \quad \nabla Q_{k-\frac{1}{2}}^R := \alpha \nabla Q_k + (1 - \alpha) \nabla Q_{k+1}$$

by using (limited) slopes from the underlying Cartesian grid cells. This definition of the gradients is an important component of the new method. Note that this choice of  $h$ -box values and gradients is exact for linear solutions.

We can now construct left and right values at each grid cell interface. For the approximation of the advection equation with advection speed  $u > 0$  the only modification (compared to the equidistant grid case) is needed in the construction of  $Q_{k+\frac{1}{2}}^-$ , which has the form

$$(3.23) \quad Q_{k+\frac{1}{2}}^- = Q_{k+\frac{1}{2}}^L + \frac{h}{2} \nabla Q_{k+\frac{1}{2}}^L.$$

We obtain the one-dimensional Runge–Kutta  $h$ -box method for the advection equation by defining

$$(3.24) \quad \begin{aligned} \mathcal{L}(Q)_k &= -\frac{u}{\alpha h} \left( Q_{k+\frac{1}{2}}^- - Q_{k-\frac{1}{2}}^- \right), \\ \mathcal{L}(Q)_i &= -\frac{u}{h} \left( Q_{i+\frac{1}{2}}^- - Q_{i-\frac{1}{2}}^- \right), \quad i \neq k, \end{aligned}$$

and using this operator in the time integration (3.2).

Now we discuss the TVD property of the  $h$ -box method. We will show that the  $h$ -box method defined by (3.15)–(3.24) is TVD if the underlying Cartesian grid cell gradients are limited using the minmod limiter. Since minmod is one of the most diffusive limiters, we also discuss the construction of  $h$ -box methods based on the less diffusive MC limiter. For this case we will show that the  $h$ -box method is TVD if

some additional limiting is applied. This additional limiting either might be applied to the  $h$ -box gradients as discussed in Lemma 3.2 below, or it might be expressed as an additional limiting of the underlying Cartesian gradients near the small cell; see Theorem 3.3.

As outlined above, it is enough to show that the spatial discretization (3.24) is TVD for the explicit Euler method assuming that  $u\Delta t/h \leq 0.5$ . We consider here the case of monotone increasing data. The only equations that are different from the regular grid case are the updates for cells  $k$  and  $k + 1$ , which use the  $h$ -box values  $Q_{k+\frac{1}{2}}^L$  and slope  $\nabla Q_{k+\frac{1}{2}}^L$ , defined above. For these cells the scheme is

$$(3.25) \quad Q_k^{n+1} = Q_k^n - \frac{\nu}{\alpha} \left( Q_{k+1/2}^L + \frac{h}{2} \nabla Q_{k+1/2}^L - Q_{k-1}^n - \frac{h}{2} \nabla Q_{k-1}^n \right),$$

$$(3.26) \quad Q_{k+1}^{n+1} = Q_{k+1}^n - \nu \left( Q_{k+1}^n + \frac{h}{2} \nabla Q_{k+1}^n - Q_{k+1/2}^L - \frac{h}{2} \nabla Q_{k+1/2}^L \right).$$

Looking first at the small cell equation (3.25) and the definitions (3.15) and (3.19), the small cell update simplifies to

$$(3.27) \quad Q_k^{n+1} = Q_k^n - \nu \left( Q_k^n - Q_{k-1}^n + \frac{h}{2} \nabla Q_k^n - \frac{\alpha h}{2} \nabla Q_{k-1}^n \right)$$

$$(3.28) \quad = Q_k^n - \nu \left( 1 + \frac{\nabla Q_k^n}{\frac{2}{h}(Q_k^n - Q_{k-1}^n)} - \frac{\alpha \nabla Q_{k-1}^n}{\frac{2}{h}(Q_k^n - Q_{k-1}^n)} \right) (Q_k^n - Q_{k-1}^n).$$

Note that the small term  $\alpha$  in the denominator has disappeared, again indicating the *cancellation* property of  $h$ -box schemes. As in the uniform grid case, we will choose  $\nu$  times the term in the large parenthesis as  $C_{k-1}$ . This coefficient is less than 1 if we limit the gradient  $\nabla Q_k^n$  using

$$(3.29) \quad Q_k^n - \frac{h}{2} \nabla Q_k^n \geq Q_{k-1}^n$$

instead of (the possibly expected)

$$(3.30) \quad Q_k^n - \frac{\alpha h}{2} \nabla Q_k^n \geq Q_{k-1}^n.$$

Equation (3.30) agrees with the geometric interpretation for MC limiting at the small cell, but for very small cell sizes  $\alpha \approx 0$ , it does not provide enough limiting to achieve the TVD property. Equation (3.29) is a stronger condition. Note that linear solutions automatically satisfy (3.29) and so do not need to be limited (thus preserving second order accuracy). Equation (3.29) is also not quite as stringent as the minmod limiter on a nonuniform mesh, which would require that when the solution in one cell is reconstructed to the neighboring cell center, it should not exceed that value. This would give

$$(3.31) \quad Q_k^n - \frac{(1 + \alpha)h}{2} \nabla Q_k^n \geq Q_{k-1}^n,$$

an even stronger condition.

The MC limiter for  $\nabla Q_{k-1}^n$  gives the term  $\frac{\nabla Q_{k-1}^n}{\frac{2}{h}(Q_k^n - Q_{k-1}^n)} \leq 1$ , and multiplying by  $\alpha$  only further reduces this term.

Looking next at (3.26) for the cell to the right of the small cell, we can rewrite this as

(3.32)

$$\begin{aligned}
 Q_{k+1}^{n+1} &= Q_{k+1}^n - \nu \left( Q_{k+1}^n - Q_{k+1/2}^L + \frac{h}{2} \nabla Q_{k+1}^n - \frac{h}{2} \nabla Q_{k+1/2}^L \right) \\
 (3.33) \quad &= Q_{k+1}^n - \nu \left( \frac{Q_{k+1}^n - Q_{k+1/2}^L}{(Q_{k+1}^n - Q_k^n)} + \frac{\nabla Q_{k+1}^n}{\frac{2(Q_{k+1}^n - Q_k^n)}{h}} - \frac{\nabla Q_{k+1/2}^L}{\frac{2(Q_{k+1}^n - Q_k^n)}{h}} \right) (Q_{k+1}^n - Q_k^n).
 \end{aligned}$$

Again  $C_k$  will be  $\nu$  times the term in the large parentheses. To show that  $C_k \leq 1$  for  $\nu \leq 0.5$ , we prove the following lemma.

LEMMA 3.2. Consider a grid with mesh width  $x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} = h$  for all  $i$  with  $i \neq k$  and mesh width  $x_{k+\frac{1}{2}} - x_{k-\frac{1}{2}} = \alpha h$ ,  $0 < \alpha \leq 1$ . Let  $Q$  be a monotone increasing grid function and  $\nabla Q$  a grid function of limited gradients. The  $h$ -box values  $Q_{k+\frac{1}{2}}^L$  and the  $h$ -box gradient  $\nabla Q_{k+\frac{1}{2}}^L$  are computed using (3.15), (3.19). Assume that the  $h$ -box gradient is limited such that

$$(3.34) \quad Q_{k+\frac{1}{2}}^L + \frac{h}{2} \nabla Q_{k+\frac{1}{2}}^L \leq Q_{k+1}.$$

Then the following inequalities hold:

$$(3.35) \quad 0 \leq \frac{\nabla Q_{k+1}}{\frac{2}{h}(Q_{k+1} - Q_k)} \leq 1,$$

$$(3.36) \quad 0 \leq \frac{\nabla Q_{k+\frac{1}{2}}^L}{\frac{2}{h}(Q_{k+1} - Q_k)} \leq \frac{Q_{k+1} - Q_{k+\frac{1}{2}}^L}{Q_{k+1} - Q_k} \leq 1.$$

*Proof.* The inequalities (3.35) hold if a TVD limiter such as the MC limiter is used in order to compute the limited gradient  $\nabla Q_{k+1}$ .

To get

$$(3.37) \quad \frac{Q_{k+1} - Q_{k+1/2}^L}{(Q_{k+1} - Q_k)} \leq 1,$$

we need  $Q_{k+\frac{1}{2}}^L \leq Q_k$ . We have

$$(3.38) \quad Q_{k+\frac{1}{2}}^L = \alpha Q_k + (1 - \alpha) \left( Q_{k-1} + \frac{\alpha h}{2} \nabla Q_{k-1} \right) \leq Q_k,$$

$$(3.39) \quad (1 - \alpha) \left( Q_{k-1} + \frac{\alpha h}{2} \nabla Q_{k-1} \right) \leq (1 - \alpha) Q_k, \quad \text{or}$$

$$(3.40) \quad \left( Q_{k-1} + \frac{\alpha h}{2} \nabla Q_{k-1} \right) \leq Q_k.$$

We already have

$$\left( Q_{k-1} + \frac{h}{2} \nabla Q_{k-1} \right) \leq Q_k$$

for the MC limiter, and the factor  $\alpha$  only helps here.

The inequality

$$(3.41) \quad \frac{\nabla Q_{k+\frac{1}{2}}^L}{\frac{2}{h}(Q_{k+1} - Q_k)} \leq \frac{Q_{k+1} - Q_{k+\frac{1}{2}}^L}{(Q_{k+1} - Q_k)}$$

follows under the assumption (3.34).  $\square$

Our assumption (3.34) is the usual MC limiter but applied here to the  $h$ -box values. Imposing this extra limiting step on the  $h$ -box gradient, in addition to applying it to the Cartesian gradients, results in a TVD scheme.

Alternatively, we can try instead to find conditions for the gradients on the Cartesian grid alone that give a TVD scheme. Substituting in (3.34) using the definition of  $Q^L$  and  $\nabla Q^L$  in (3.15) and (3.19), we need to find conditions for limiting the gradients  $\nabla Q_{k-1}$  and  $\nabla Q_k$  that will give us

$$(3.42) \quad \alpha Q_k + (1 - \alpha)Q_{k-1} + \frac{(1 - \alpha)}{2}\alpha h \nabla Q_{k-1} + \frac{h}{2}(\alpha \nabla Q_k + (1 - \alpha)\nabla Q_{k-1}) \leq Q_{k+1}.$$

This can be rearranged as

$$(3.43) \quad (1 - \alpha) \left( Q_{k-1} + \frac{h}{2}\nabla Q_{k-1} + \frac{\alpha h}{2}\nabla Q_{k-1} \right) + \alpha \left( Q_k + \frac{h}{2}\nabla Q_k \right) \leq Q_{k+1}.$$

Both in the neighboring cell of the small cell as well as in the small cell, we need to assume a slightly stronger limiting of the gradient than what would be obtained by using only the MC limiter. If the minmod limiter is used to limit  $\nabla Q_{k-1}$ , then we obtain

$$(3.44) \quad Q_{k-1} + \frac{h}{2}(1 + \alpha)\nabla Q_{k-1} \leq Q_k.$$

In cell  $k$  we require the additional limiting

$$Q_k + \frac{h}{2}\nabla Q_k \leq Q_{k+1}.$$

This expression for the small cell  $k$  has the same form as (3.29) but applied to the cell on the right. Under these two additional conditions on the gradients (which are automatically satisfied for minmod but not for the MC limiter), the left-hand side of (3.43) is less than or equal to

$$(3.45) \quad (1 - \alpha)Q_k + \alpha Q_{k+1} \leq Q_{k+1}$$

since we are considering the case of monotone increasing data  $Q_k \leq Q_{k+1}$ . Thus (3.34) would be satisfied. A slightly different requirement for  $\nabla Q_{k-1}$  is also possible that would limit that cell against its neighbor two cells away:

$$(3.46) \quad Q_{k-1} + (1 + \alpha)\frac{h}{2}\nabla Q_{k-1} \leq Q_{k+1}.$$

We summarize our results.

**THEOREM 3.3.** *Consider the approximation of the advection equation with positive advection speed on a grid with mesh width  $x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} = h$  for all  $i$  with  $i \neq k$  and mesh width  $x_{k+\frac{1}{2}} - x_{k-\frac{1}{2}} = \alpha h$  with  $0 < \alpha \leq 1$ . The  $h$ -box method consisting of the time integration (3.2) with the operator  $\mathcal{L}(Q)$  defined in (3.24) and the spatial reconstruction (3.23) is TVD under the CFL condition  $\nu \leq 0.5$*

- if either all Cartesian grid cell gradients  $\nabla Q$  (including the small cell gradient) are limited using the minmod limiter; or
- if all Cartesian grid cell gradients are limited using the MC limiter and additional limiting is applied to the gradients in the small cell  $k$  and the neighboring cell  $k - 1$ .

The small cell gradient  $\nabla Q_k$  is limited such that

$$Q_k - \frac{h}{2}\nabla Q_k \geq Q_{k-1} \quad \text{and} \quad Q_k + \frac{h}{2}\nabla Q_k \leq Q_{k+1}$$

hold in the case of increasing data, and

$$Q_{k-1} \geq Q_k - \frac{h}{2}\nabla Q_k \quad \text{and} \quad Q_k + \frac{h}{2}\nabla Q_k \geq Q_{k+1}$$

hold in the case of decreasing data.

The neighboring cell gradient  $\nabla Q_{k-1}$  is limited such that

$$Q_{k-1} + (1 + \alpha)\frac{h}{2}\nabla Q_{k-1} \leq Q_{k+1}$$

holds in the case of monotone increasing data, and

$$Q_{k-1} + (1 + \alpha)\frac{h}{2}\nabla Q_{k-1} \geq Q_{k+1}$$

holds in the case of monotone decreasing data.

**4. Two-dimensional finite volume method.** Now consider a two-dimensional hyperbolic system of the form

$$(4.1) \quad q_t(x, y, t) + f(q(x, y, t))_x + g(q(x, y, t))_y = 0$$

with given initial values  $q(x, y, 0)$ . Here,  $q : \mathbb{R}^2 \times \mathbb{R}^+ \rightarrow \mathbb{R}^m$  is a vector of conserved quantities, and  $f, g : \mathbb{R}^m \rightarrow \mathbb{R}^m$  are vector-valued flux functions.

It is easier to describe the numerical method by looking at the advection equation

$$(4.2) \quad q_t + uq_x + vq_y = 0,$$

with  $u, v \in \mathbb{R}$ .

**4.1. Grid-aligned finite volume method.** We briefly review the basic two-dimensional finite volume method, since we will need to refer to it when we generalize to cut cells. This method will be used on the regular portion of the grid. The two-stage second order SSP method in time is coupled with this standard finite volume scheme in space. At each stage gradients are computed using central differences, and a standard slope limiter is applied. These gradients are used to reconstruct the solution to the midpoint of each cell edge. For example,  $Q_{i+\frac{1}{2},j}^- = Q_{i,j} + \frac{\Delta x}{2}\partial_x Q_{i,j}$ . The cell to the right,  $(i+1, j)$ , will predict the right state at this face  $Q_{i+\frac{1}{2},j}^+ = Q_{i+1,j} - \frac{\Delta x}{2}\partial_x Q_{i+1,j}$ . A Riemann solver, denoted  $RP(Q^-, Q^+)$ , determines the single state to use at each edge to evaluate the operator  $\mathcal{L}$  (i.e., the flux differences of  $f$  and  $g$  in this case).

In two dimensions the stability limit of this scheme for linear advection on an equidistant grid with  $h = \Delta x = \Delta y$  and centered gradients without limiters is  $\Delta t(|u| + |v|)/h \leq 1$ . Note that if the velocities  $u$  and  $v$  are approximately equal, this gives a time step that is half that of a corner-coupled scheme such as the corner transport upwind method. The question of stability in two dimensions will come up again in section 5.2 when we discuss the optimal length for an  $h$ -box.

**4.2. Orthogonal rotated grid method.** We first present the basic idea of rotated grid methods for regular Cartesian grids without embedded boundaries. See also [27, 23] for related regular-grid rotated grid methods.

In a rotated grid approach, the normal flux at a grid cell interface is computed as a superposition of fluxes computed in a direction  $\vec{\xi} = (\alpha, \beta)^T$  and the orthogonal direction  $\vec{\eta} = (-\beta, \alpha)^T$ . The vectors  $\vec{\xi}$  and  $\vec{\eta}$  are unit normals that can be defined in several ways.

We define variables

$$(4.3) \quad \xi = \alpha x + \beta y, \quad \eta = -\beta x + \alpha y,$$

and transform (4.1) into a system in these new variables, which has the form

$$(4.4) \quad \partial_t q(\xi, \eta, t) + \partial_\xi f^\xi(q(\xi, \eta, t)) + \partial_\eta f^\eta(q(\xi, \eta, t)) = 0,$$

with

$$f^\xi(q) = \alpha f(q) + \beta g(q), \quad f^\eta(q) = -\beta f(q) + \alpha g(q).$$

The numerical fluxes  $F^\xi$  and  $F^\eta$  are calculated by solving Riemann problems in the  $\vec{\xi}$  and the  $\vec{\eta}$  direction. For rotationally invariant systems of conservation laws (such as the Euler equations) this is particularly simple, since the form of  $f^\xi$  and  $f^\eta$  agrees with  $f(q)$  by using the rotated velocities instead of the velocities in the  $x$  and the  $y$  direction. Thus a single Riemann solver can be used for all directions. The numerical flux normal to the cell edge is obtained by projecting  $F^\xi$  and  $F^\eta$  onto the direction of the coordinate frame.

In order to calculate fluxes in these rotated directions, values of the dependent variables have to be assigned. At the interface  $(i - \frac{1}{2}, j)$ , for example, we compute values  $(Q_\xi^{L,R})_{i-\frac{1}{2},j}$  as well as  $(Q_\eta^{L,R})_{i-\frac{1}{2},j}$ . Our approach for computing these values is motivated by the multidimensional upwind method of Roe and Sidilkover [27] and can be interpreted as an  $h$ -box method as indicated in Figure 4.1 and explained in [16]. We also assign gradients of these quantities to the rotated boxes. Using these  $h$ -box values and gradients, we can reconstruct values at the grid cell interface which we denote by  $Q_\xi^{-,+}$  and  $Q_\eta^{-,+}$ . Now we compute fluxes  $F^\xi_{i-\frac{1}{2},j}$  and  $F^\eta_{i-\frac{1}{2},j}$  by solving the Riemann problems

$$(4.5) \quad \partial_t q(\xi, \eta, t) + \partial_\xi f^\xi(q(\xi, \eta, t)) = 0,$$

$$(4.6) \quad q(\xi, \eta, 0) = \begin{cases} (Q_\xi^-)_{i-\frac{1}{2},j} & : \quad \xi < x_{i-\frac{1}{2}}\alpha + y_j\beta, \\ (Q_\xi^+)_{i-\frac{1}{2},j} & : \quad \xi > x_{i-\frac{1}{2}}\alpha + y_j\beta \end{cases}$$

and

$$(4.7) \quad \partial_t q(\xi, \eta, t) + \partial_\eta f^\eta(q(\xi, \eta, t)) = 0,$$

$$(4.8) \quad q(\xi, \eta, 0) = \begin{cases} (Q_\eta^-)_{i-\frac{1}{2},j} & : \quad \eta < -x_{i-\frac{1}{2}}\beta + y_j\alpha, \\ (Q_\eta^+)_{i-\frac{1}{2},j} & : \quad \eta > -x_{i-\frac{1}{2}}\beta + y_j\alpha. \end{cases}$$

Analogously, we can compute fluxes at interfaces  $(i, j - \frac{1}{2})$ . The numerical fluxes  $F_{i-\frac{1}{2},j}$  and  $F_{i,j-\frac{1}{2}}$  which are used to update the conserved quantities are finally obtained via

$$(4.9) \quad \begin{aligned} F_{i-\frac{1}{2},j} &= \alpha F^\xi_{i-\frac{1}{2},j} - \beta F^\eta_{i-\frac{1}{2},j}, \\ F_{i,j-\frac{1}{2}} &= \beta F^\xi_{i,j-\frac{1}{2}} + \alpha F^\eta_{i,j-\frac{1}{2}}. \end{aligned}$$

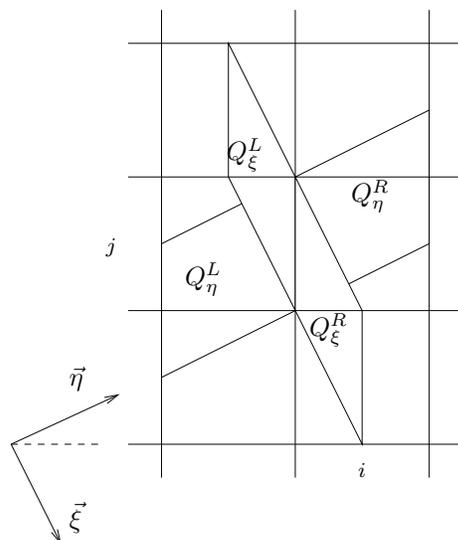


FIG. 4.1. *h*-boxes for systems, used to compute fluxes at a vertical interface.

**4.3. Nonorthogonal rotated grid method.** The directions  $\vec{\xi}$  and  $\vec{\eta}$  do not necessarily have to be orthogonal. Instead we can use general directions  $\vec{\xi} = (\alpha_1, \beta_1)^T$  and  $\vec{\eta} = (\alpha_2, \beta_2)^T$ . The rotated fluxes now have the form

$$(4.10) \quad \begin{aligned} f^\xi(q) &= \alpha_1 f(q) + \beta_1 g(q), \\ f^\eta(q) &= \alpha_2 f(q) + \beta_2 g(q). \end{aligned}$$

Numerical approximations of those fluxes at the interface  $(i - \frac{1}{2}, j)$  are again denoted by  $F_{i-\frac{1}{2},j}^\xi$  and  $F_{i-\frac{1}{2},j}^\eta$ . We obtain the numerical flux  $F_{i-\frac{1}{2},j}$  used in the finite volume approach by combining those fluxes:

$$(4.11) \quad F_{i-\frac{1}{2},j} = AF_{i-\frac{1}{2},j}^\xi + BF_{i-\frac{1}{2},j}^\eta.$$

The coefficients have to be chosen in such a way that

$$(4.12) \quad A(\alpha_1 f + \beta_1 g) + B(\alpha_2 f + \beta_2 g) = f;$$

i.e., we obtain the linear system

$$(4.13) \quad \begin{pmatrix} \beta_1 & \beta_2 \\ \alpha_1 & \alpha_2 \end{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

with the solution

$$(4.14) \quad A = \frac{\beta_2}{\alpha_1\beta_2 - \alpha_2\beta_1}, \quad B = \frac{-\beta_1}{\alpha_1\beta_2 - \alpha_2\beta_1}.$$

Clearly the directions  $\vec{\xi}$  and  $\vec{\eta}$  must not be parallel; i.e.,  $\alpha_1\beta_2 \neq \alpha_2\beta_1$ .

Analogously, at the interface  $(i, j - \frac{1}{2})$  we require

$$(4.15) \quad A(\alpha_1 f + \beta_1 g) + B(\alpha_2 f + \beta_2 g) = g$$

and obtain the coefficients

$$(4.16) \quad A = \frac{\alpha_2}{\beta_1 \alpha_2 - \beta_2 \alpha_1}, \quad B = \frac{\alpha_1}{\beta_2 \alpha_1 - \beta_1 \alpha_2}.$$

In our application of rotated grid methods for the construction of cut cell methods, the directions  $\vec{\xi}$  and  $\vec{\eta}$  will be far from parallel but not necessarily orthogonal.

**5. The embedded boundary  $h$ -box method.** We now use the rotated grid  $h$ -box method for the approximation of two-dimensional systems of conservation laws (4.1) at embedded boundaries. We have implemented and tested our embedded boundary method for both the two-dimensional advection equation as well as the Euler equations of gas dynamics. We always assume that the embedded boundary is a solid wall reflecting boundary. For the advection equation, this means that the boundary flux is equal to zero. For the Euler equations the velocity normal to the boundary segment is equal to zero.

We assume that the embedded boundary is described by a piecewise linear segment that cuts through each boundary cell once. Each irregular grid cell is then a polygon with at most five sides, and the finite volume method can be written in the semidiscrete form

$$(5.1) \quad \frac{d}{dt} Q_{ij}(t) = -\frac{1}{\kappa_{ij} \Delta x \Delta y} (h_{i+\frac{1}{2},j} F_{i+\frac{1}{2},j}(t) - h_{i-\frac{1}{2},j} F_{i-\frac{1}{2},j}(t) + h_{i,j+\frac{1}{2}} F_{i,j+\frac{1}{2}}(t) - h_{i,j-\frac{1}{2}} F_{i,j-\frac{1}{2}}(t) + h_b F_b(t)),$$

where we make use of the Cartesian grid structure. For cut cells, some of the edge lengths might vanish. The length of the boundary segment is denoted by  $h_b$ , and the corresponding flux in Cartesian coordinates is denoted by  $F_b$ . The area of the grid cell  $(i, j)$  is  $|C_{ij}| = \kappa_{ij} \Delta x \Delta y$ , where  $0 \leq \kappa_{ij} \leq 1$  is the area fraction. For the discretization in time we again use the SSP Runge–Kutta method (3.2).

**5.1.  $h$ -box values at smoothly varying embedded boundaries.** To explain the flux calculation at embedded boundaries, we first describe the computation of  $h$ -box values, which we denote as in the previous section by  $Q_{\xi}^{L,R}$  and  $Q_{\eta}^{L,R}$ . Let us consider a triangular-shaped cut cell as depicted in Figure 5.1 and a planar boundary segment. Let  $\vec{n}$  be the direction normal to the boundary segment, pointing into the flow domain. We use a rotated grid method with  $\vec{\eta} = (-\beta, \alpha)^T = \vec{n}$  and  $\vec{\xi} = (\alpha, \beta)^T$ .

At the boundary segment, we construct  $h$ -boxes as indicated in Figure 5.1(a). The boundary  $h$ -box which points into the flow domain overlaps with at most four grid cells. The  $h$ -box value  $Q_{\eta}^R$  which is assigned to this  $h$ -box is obtained by an area-weighted average over piecewise linear reconstructed values of the underlying Cartesian cut cell grid. To model a reflecting boundary, the value  $Q_{\eta}^L$  is obtained by reflecting  $Q_{\eta}^R$ , i.e., by negating the normal velocity component.

We also construct  $h$ -boxes in the  $\vec{\eta}$  direction at the horizontal and vertical grid cell interface. These boxes are indicated in Figure 5.1(b). Note that those  $h$ -boxes overlap with the boundary boxes of Figure 5.1(a) except for an area the size of the small grid cell. This will lead to the cancellation property for fluxes in the  $\vec{\eta}$  direction.

At the horizontal and vertical edges of the cut cell, we furthermore construct  $h$ -boxes in the  $\xi$  direction, shown in Figures 5.1(c) and 5.1(d), and we assign  $h$ -box

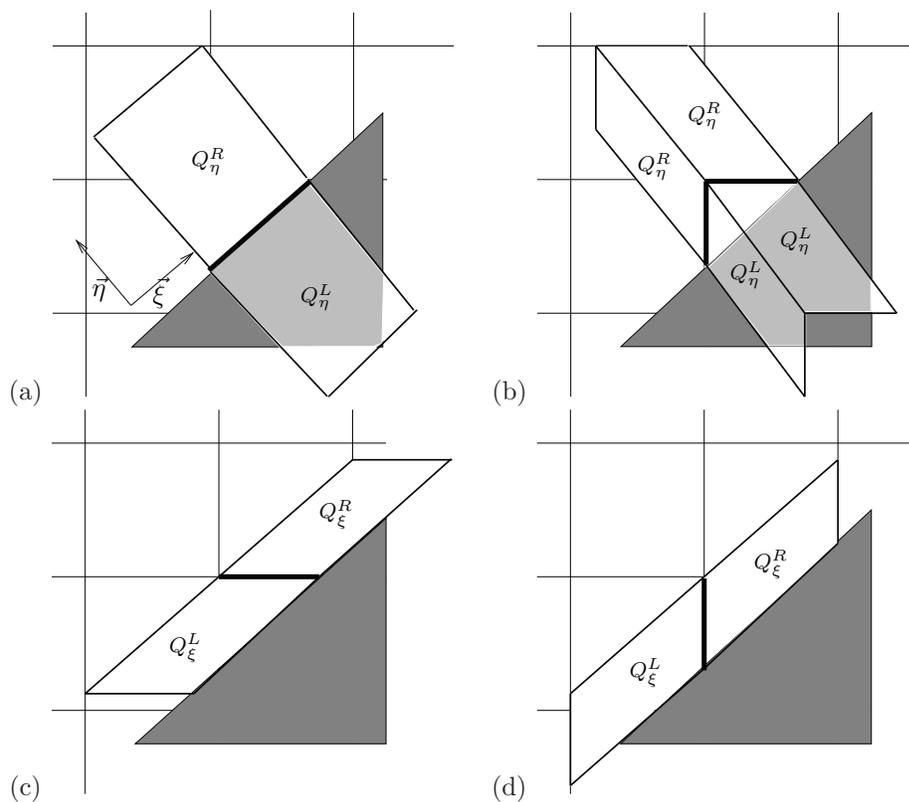


FIG. 5.1. *h*-boxes of the rotated grid cut cell method: (a) boundary *h*-box, (b) *h*-boxes in direction normal to the boundary segment at the two cut cell interfaces, (c) and (d) *h*-boxes tangential to the boundary segment.

values to these boxes, also from the underlying Cartesian grid. Note that these boxes overlap except for an area the size of the small grid cell.

Let us now consider in more detail the computation of *h*-box values in the  $\vec{\eta}$  direction at one grid interface; compare with Figure 5.2. Consider first the *h*-box which lies completely inside the flow domain. This *h*-box intersects with two grid cells of the underlying Cartesian cut cell mesh. We compute the area fractions for those polygonal fractions of the *h*-box. Furthermore, we compute the center of mass for those fractions. Now the *h*-box value is obtained from the value of the dependent variables at those centers of mass and then using an area-weighted average over the reconstructed values. This reconstruction uses gradients in the grid cells of the underlying Cartesian grid which are computed using a standard least square procedure that includes all neighbors sharing an edge with a cut cell [1]. If that does not yield at least three equations, the diagonal cell between the two neighbors is included as well.

An *h*-box might also lie partly beyond the boundary. In this case the part of the *h*-box which lies inside the solid body is reflected to the flow domain, as indicated in Figure 5.2(b). Again using an area-weighted average, we assign a value of the dependent variables to the reflected fraction of the *h*-box. The part of the *h*-box which lies inside the solid body contributes the reflected value to the computation of the *h*-box value.

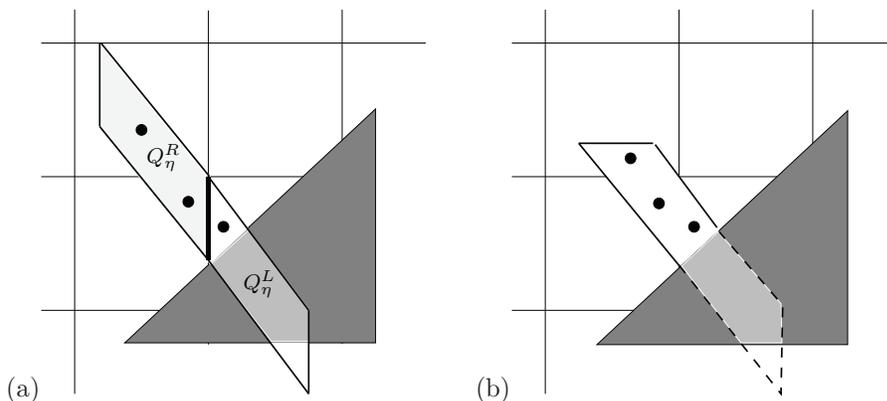


FIG. 5.2. Computation of  $h$ -box values. The dots indicate the centers of mass of the polygonal fractions. (a) shows the location of the  $h$ -boxes. The shaded  $h$ -box is used to compute  $Q_\eta^R$ . (b) shows the reflection of the  $h$ -box fraction (dashed lines) lying inside the solid body. The reflected fraction contributes to the computation of  $Q_\eta^L$ .

If a grid cell interface belongs to only one cut cell, then the rotated grid is chosen in a direction normal and tangential to the boundary segment of that cut cell. However, a grid cell interface might also belong to two cut cells. The directions of the rotated grid for this interface can then be chosen based on an average of the directions imposed by the two neighboring boundary segments. Alternatively, the interface flux can be computed using a rotated grid with an orientation that is imposed by the smaller cut cell. For a smoothly varying boundary this will still guarantee the cancellation property. We have used both and have seen no reason to prefer one or the other.

In a preprocessing step, all these area fractions and centers of mass are computed once. During each time step, this geometric information is used in the form of coefficients in our formulas for the computation of  $h$ -box values.

Since the rotated grid method is more expensive than the regular method, involving twice as many Riemann problems at cut cell edges, we usually apply this to compute only the cut cell fluxes. At regular grid cell interfaces we use the nonrotated method described in section 4.1. This nonsmooth change is a potential source of additional truncation error. We will examine this in the computational experiments in section 6.

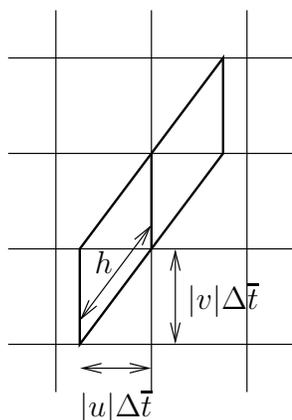
**5.2. Determination of  $h$ -box length.** We next discuss the issue of what the length of  $h$ -boxes should be. Along a vertical cell interface midpoint  $(x_{i+\frac{1}{2}}, y_j)$ , the domain of dependence for the advection equation (4.2) at time  $T$  is the point  $(x_{i+\frac{1}{2}} - uT, y_j - vT)$ .

First consider a rotated grid  $h$ -box method with  $h$ -box constructed in the upwind direction and  $h$ -box values which are obtained by area-weighted averaging over piecewise constant cell averages. The domain of dependence of the exact solution is inside the numerical domain of dependence if and only if

$$(5.2) \quad h \geq \sqrt{u^2 + v^2} \Delta t;$$

compare with Figure 5.3.

Since the TVD-Runge–Kutta method of section 4.1, used for the regular part of

FIG. 5.3. Determination of  $h$ -box length

the grid, is linearly stable, provided that

$$\Delta t(|u| + |v|)/\Delta_{x,y} \leq 1,$$

the  $h$ -box length

$$(5.3) \quad h = \frac{\sqrt{u^2 + v^2}}{|u| + |v|} \Delta_{x,y}$$

satisfies the geometric CFL condition for all appropriate time steps. Here we assume that the regular grid part is equidistant with  $\Delta_{x,y} = \Delta x = \Delta y$ .

Alternatively, we can use the larger, fixed  $h$ -box length

$$h = \Delta_{x,y},$$

which has the (slight) computational advantage of being constant independent of direction.

For the advection equation (4.2), Roe and Sidilkover suggested a rotated grid upwind method which is stable, provided that

$$(5.4) \quad \max\left(\frac{|u|\Delta t}{\Delta x}, \frac{|v|\Delta t}{\Delta y}\right) \leq 1.$$

In [16], we showed that this method can be interpreted as a rotated grid  $h$ -box method with  $h$ -boxes constructed in the upwind direction and with the  $h$ -box length

$$(5.5) \quad h = \sqrt{1 + \left(\frac{\min(|u|, |v|)}{\max(|u|, |v|)}\right)^2} \Delta_{x,y}.$$

If the overall scheme used in the regular parts of the grid is stable for time steps which satisfy the less restrictive time step (5.4), then we use this longer  $h$ -box length.

We can extend the idea to the case of advection in a spatially varying velocity field, i.e.,  $u = u(x, y)$ ,  $v = v(x, y)$ . Now the  $h$ -box length depends on the position. This is used in the first test case of section 6. In the more general case of an embedded boundary method for which the boundary segment is not aligned with the

flow direction, we typically construct  $h$ -boxes in directions tangential and normal to the boundary. Then the formulas for the  $h$ -box length of an  $h$ -box in the  $\vec{\xi} = (\alpha, \beta)$  direction are obtained by replacing  $(u, v)$  in (5.3) or (5.5) by  $(\alpha, \beta)$ .

Although we restrict our considerations in this paper to equidistant Cartesian grids with  $\Delta_{x,y} = \Delta x = \Delta y$ ,  $h$ -box methods can easily be extended to the more general case  $\Delta x \neq \Delta y$ . In this case the  $h$ -box length can again be determined by starting with inequality (5.2) and replacing the  $\Delta t$  term by the CFL condition for these grids.

In the graphical illustrations of our  $h$ -box method, we used the  $h$ -box length (5.5).

**5.3. Spatial reconstruction of interface values.** In order to get second order accurate approximations in space with the method of lines approach, we again need to reconstruct values to the grid cell interfaces and compute fluxes by solving Riemann problems for these reconstructed values.

To compute these values, we need to assign a gradient to each  $h$ -box. These gradients are denoted by  $\nabla Q_{\xi,\eta}^{L,R}$ . These are gradients in Cartesian coordinates assigned to the  $h$ -boxes in the  $\vec{\xi}$  and  $\vec{\eta}$  directions. Recall that we have already computed limited least square gradients in every Cartesian cut cell. We use a scalar minmod limiter which generalizes the one-dimensional TVD results of section 3.2. A simple approach for the  $h$ -box *virtual* grid cells is to compute a gradient using an area-weighted average of gradients from the Cartesian cut cell mesh. Here we use the same area-weighted average that was used for the computation of  $h$ -box values. Note that this choice of  $h$ -box gradients is linearity preserving. Left and right values at the grid cell interface can now be computed:

$$(5.6) \quad \begin{aligned} Q_{\xi}^{-} &= Q_{\xi}^L + \frac{h}{2} \vec{\xi} \cdot \nabla Q_{\xi}^L, & Q_{\xi}^{+} &= Q_{\xi}^R - \frac{h}{2} \vec{\xi} \cdot \nabla Q_{\xi}^R, \\ Q_{\eta}^{-} &= Q_{\eta}^L + \frac{h}{2} \vec{\eta} \cdot \nabla Q_{\eta}^L, & Q_{\eta}^{+} &= Q_{\eta}^R - \frac{h}{2} \vec{\eta} \cdot \nabla Q_{\eta}^R. \end{aligned}$$

Another possibility is to compute a (limited) gradient in the  $\xi$  and the  $\eta$  direction by using two additional  $h$ -box values as indicated in the one-dimensional case in Figure 2.1. This was done in our original  $h$ -box method but is more complicated than the first approach. In the computational experiments we will compare the accuracy.

The flux components are then obtained by rotating these interface values in the direction  $\vec{\xi}$  and  $\vec{\eta}$ , respectively, and evaluating the fluxes by solving Riemann problems, i.e.,

$$(5.7) \quad F^{\xi} = \mathcal{F}(Q_{\xi}^{-}, Q_{\xi}^{+}), \quad F^{\eta} = \mathcal{F}(Q_{\eta}^{-}, Q_{\eta}^{+}).$$

These fluxes are then rotated back to Cartesian coordinates, and the interface fluxes are computed via (4.9).

**5.4. The rotated grid cut cell method for nonsmooth embedded boundaries.** Now assume that the boundary varies nonsmoothly. We distinguish between a convex and a concave boundary segment as illustrated in Figure 5.4. A convex boundary segment is easier to handle. In Figure 5.4(a), there are two triangular-shaped cut cells. The nonsmooth variation of the rotated grid method does not lead to any problems in this case. For both triangular-shaped cut cells, we can choose a rotated grid using the normal and tangential directions of the boundary segment from that cell. In Figure 5.4(b), we depict a case where a fraction of the  $x$ -face at a grid cell interface (which is marked in bold) belongs to two cut cells which have very different

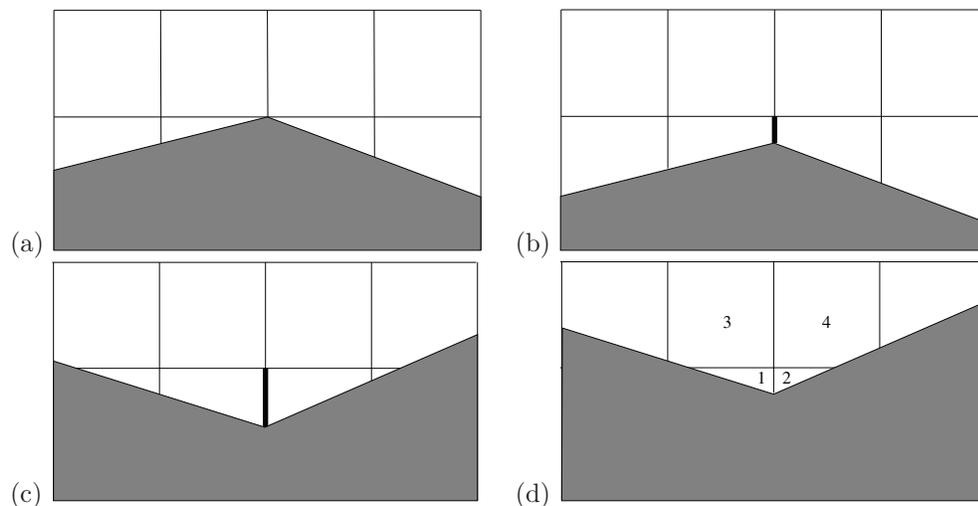


FIG. 5.4. Cut cells at a convex (a)–(b) and a concave (c)–(d) nonsmoothly varying boundary segment.

orientations of the boundary segment. With an orthogonal rotated grid method, we can either compute the fluxes based on the orientation of the cut cell to the left or the cut cell to the right. Since these cut cells are not very small, this situation is not critical in terms of stability. However, it can be seen that only the boxes in the direction normal to the boundary segment contribute to the cancellation property for the left and right grid cells. Therefore, one can compute the flux at the bold-marked interface using a nonorthogonal rotated grid method in which the two directions are the normal directions of the two boundary segments.

Next consider the case of a concave boundary segment. The situation in Figure 5.4(c) is again less critical, since the cut cells are relatively large. In this situation we can use the orthogonal rotated grid method based on the local boundary orientation. At the bold-marked interface we can compute fluxes using the nonorthogonal rotated grid method in which the two normal directions determine the directions for the flux computation. Alternatively, the two tangential directions can be used for this flux computation.

Figure 5.4(d) shows the most challenging case: Two small triangular-shaped grid cells are located at the corner of a nonsmoothly varying concave boundary segment. There are different possibilities for constructing a stable rotated grid method in this situation.

(1) One can compute the cell average of the conserved quantities based on an orthogonal rotated grid method for each of these two small triangular-shaped cut cells. Such a method would satisfy the cancellation property, but it would not be conservative, since the flux at the interface which is shared by both of the triangular-shaped cells (i.e., the flux between cell 1 and cell 2) would have a different form for each of the two grid cells. However, a loss of conservation is what we typically want to avoid when approximating conservation laws. Therefore, we do not use such an approach here.

(2) One can change the geometry in such a way that tiny cells with this particular configuration are ignored; i.e., we would assume the two tiny grid cells are completely inside the solid body. This approach would make the cut cell grid generation more difficult.

(3) We compute fluxes at the boundary of the two small cells in the direction normal to the boundary segment using our rotated grid  $h$ -box method. The flux between cell 1 and cell 2 is computed using the nonorthogonal rotated grid  $h$ -box method by choosing the two normal directions. Now we cannot compute the flux between cell 1 and cell 3 as well as the flux between cell 2 and cell 4 in such a way that the cancellation property is satisfied. We ignore this flux computation and instead compute all the remaining fluxes of cell 3 and cell 4. Using all these fluxes we compute average values of the conserved quantities for cell 1 and cell 3 and use this value in both grid cells. Analogously we compute the average of the conserved quantities in cell 2 and cell 4. This can be thought of as cell merging but on a much smaller scale. Still, an important question is whether this approach will carry over to three-dimensional complicated geometries in a straightforward way. This approach will be used for a test problem in section 6.2.

## 6. Numerical results.

**6.1. Convergence study.** First we present an accuracy study for solid body rotation inside an annulus defined by two concentric circles. The radius of the inner circle is  $R_1 = 0.75$ , and the radius of the outer circle is  $R_2 = 1.25$ . The annulus is embedded in the rectangular domain  $[-1.5, 1.5] \times [-1.5, 1.5]$  which is discretized with a Cartesian mesh. We use initial values of the form

$$q(x, y, 0) = 0.5 (\operatorname{erf}(5(\theta - \pi/3)) + \operatorname{erf}(5(2\pi/3 - \theta)))$$

with  $\theta = \arctan(y/x)$ . The velocity field describing solid body rotation was obtained using the stream function  $\psi(x, y) = \pi(R_2^2 - r^2)/5$  with  $r = \sqrt{x^2 + y^2}$ . One rotation is completed at time  $t = 5$ .

In all our calculations, we choose the time steps in such a way that

$$\max_{x,y} (|u(x, y)| + |v(x, y)|) \frac{\Delta t}{\Delta x} \leq 1$$

is satisfied. For this smooth test problem we do not use limiters.

In order to measure the accuracy of our embedded boundary method at the boundary as well as inside the domain, we use two different norms. Inside the domain (regular cells + cut cells) we calculate the error using the  $L_1$ -norm of the relative error

$$(6.1) \quad E_d = \frac{\sum_{i,j} |Q_{i,j} - q(x_i, y_j)| \kappa_{i,j}}{\sum_{i,j} |q(x_i, y_j)| \kappa_{i,j}},$$

where  $Q_{i,j}$  is the numerical solution in grid cell  $(i, j)$ ,  $q(x_i, y_j)$  is the exact solution at the center of mass of the grid cell, and  $\kappa_{i,j}$  is the area fraction of the grid cell. To separately measure the boundary error, we sum over the cut cells and calculate the norm

$$(6.2) \quad E_b = \frac{\sum_{(i,j) \in K} |Q_{i,j} - q(x_i, y_j)| b_{i,j}}{\sum_{(i,j) \in K} |q(x_i, y_j)| b_{i,j}},$$

where  $b_{i,j}$  is the length of the boundary segment in grid cell  $(i, j)$  and  $K$  is the set of all cut cells. We compute the error on grids with different mesh widths and use these error values to estimate the experimental order of convergence (EOC).

For the method of lines approach used in this paper, we obtain the most accurate results using the  $h$ -box length (5.3). Therefore, we present numerical results for that  $h$ -box length. However, there is not much variation in accuracy between the methods.

Tables 6.1–6.3 show results of a numerical convergence study.

For the computations of Table 6.1 we use the rotated grid approach for all cut cell fluxes and the grid-aligned method at the regular grid cell interfaces. The gradient  $\nabla Q_\xi$  assigned to an  $h$ -box is computed by an area-weighted average of gradients from the grid cells covered by the  $h$ -box. This is our simplest version of the  $h$ -box method.

TABLE 6.1

*Convergence study for annulus test problem. The  $h$ -box gradient  $\nabla Q_\xi$  is computed using area-weighted averaging. The rotated grid method is used only for cut cell fluxes. The  $h$ -box length is computed using (5.3). The time step is 0.005, 0.0025, 0.00125, and 0.000625, respectively.*

Mesh	Domain error	Outer boundary	Inner boundary
100 × 100	$3.6258 \times 10^{-2}$	$2.8652 \times 10^{-2}$	$6.2931 \times 10^{-2}$
200 × 200	$9.4289 \times 10^{-2}$	$7.1730 \times 10^{-3}$	$2.0467 \times 10^{-2}$
EOC	1.94	2.00	1.62
400 × 400	$2.3614 \times 10^{-3}$	$1.9339 \times 10^{-3}$	$6.1384 \times 10^{-3}$
EOC	2.00	1.89	1.74
800 × 800	$5.9263 \times 10^{-4}$	$7.3541 \times 10^{-4}$	$1.9252 \times 10^{-3}$
EOC	1.99	1.39	1.67

TABLE 6.2

*Convergence study for annulus test problem. The gradient  $\nabla Q_\xi^L$  is computed using additional  $h$ -box values as indicated in Figure 2.1. The rotated grid is used only for cut cell fluxes. The  $h$ -box length is computed using (5.3). Same constant time steps as above.*

Mesh	Domain error	Outer boundary	Inner boundary
100 × 100	$3.3963 \times 10^{-2}$	$2.0970 \times 10^{-2}$	$4.8592 \times 10^{-2}$
200 × 200	$9.0872 \times 10^{-3}$	$5.2268 \times 10^{-3}$	$1.4906 \times 10^{-2}$
EOC	1.90	2.00	1.46
400 × 400	$2.3160 \times 10^{-3}$	$1.4705 \times 10^{-3}$	$4.4369 \times 10^{-3}$
EOC	1.97	1.83	1.75
800 × 800	$5.8679 \times 10^{-4}$	$5.9915 \times 10^{-4}$	$1.4643 \times 10^{-3}$
EOC	1.98	1.29	1.60

TABLE 6.3

*Convergence study for annulus test problem. The gradient  $\nabla Q_\xi^L$  is computed using additional  $h$ -box values as indicated in Figure 2.1. The rotated grid method is used for all grid cell interfaces. The  $h$ -box length is computed using (5.3). Same constant time steps as above.*

Mesh	Domain error	Outer boundary	Inner boundary
100 × 100	$2.6955 \times 10^{-2}$	$1.8720 \times 10^{-2}$	$4.0417 \times 10^{-2}$
200 × 200	$7.0471 \times 10^{-3}$	$4.6140 \times 10^{-3}$	$1.1433 \times 10^{-2}$
EOC	1.93	2.02	1.82
400 × 400	$1.7720 \times 10^{-3}$	$1.1459 \times 10^{-3}$	$3.0071 \times 10^{-3}$
EOC	1.99	2.01	1.93
800 × 800	$4.4314 \times 10^{-4}$	$2.8817 \times 10^{-4}$	$7.9922 \times 10^{-4}$
EOC	2.00	1.99	1.91

Note that there are two issues which could potentially lead to a loss of accuracy and convergence rate. There is a nonsmooth change of the numerical method from a grid-aligned to a rotated grid approach. Furthermore, the computation of gradients by the averaging technique is in general only first order accurate, while we use second order accurate central difference gradients for the regular grid cells. In Table 6.2 we instead compute the  $h$ -box gradients using centered differencing. To do this, we construct additional  $h$ -boxes (one additional  $h$ -box in the upwind direction and one  $h$ -box in the downwind direction), which allows us to use centered differencing to assign a gradient to the  $h$ -box. Note that this modification affects only the cut cell fluxes. The error is only slightly smaller, so we conclude that it is not worth this extra work.

Finally, in Table 6.3 we show the results of a computation using a rotated grid for all fluxes. The gradients  $\nabla Q_\epsilon$  are again computed using centered differencing of  $h$ -box values. This gives the smallest error and full second order convergence rates in the cut cells as well as in the whole domain. However, the discretization errors which we obtain for our simplest  $h$ -box method are again only slightly larger. We conclude that the simplest version of our embedded boundary method is a good choice also for more complex applications.

In Figure 6.1 we plot the solution in the cut cells as a function of the angle  $\theta$ . These plots show that there are no accuracy or stability problems in any of the cut cells.

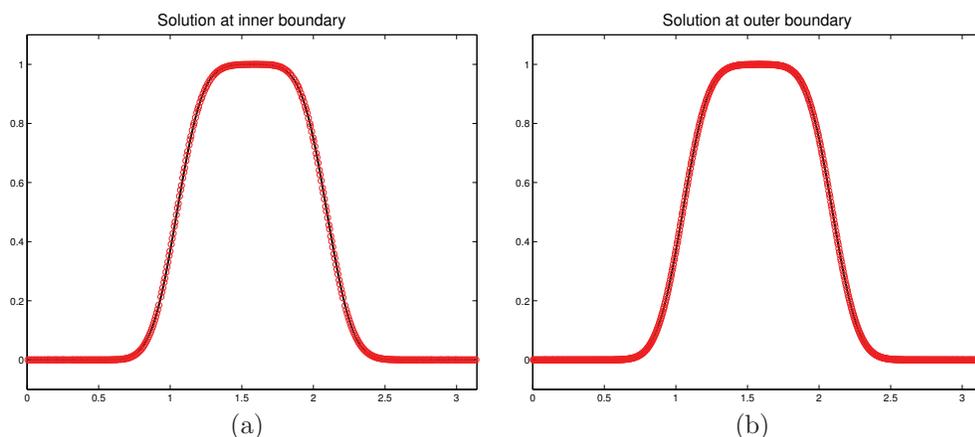


FIG. 6.1. Plot of the solution in the cut cells as a function of  $\theta$  after one rotation (i.e., at time  $t = 5$ ) computed on a  $400 \times 400$  grid. (a) Along the inner boundary segment, which contains 780 cut cells. (b) Along the outer boundary segment, which contains 1332 cut cells. The solid line is the exact solution.

**6.2. Shock reflection.** As a test case for the two-dimensional Euler equations we consider shock reflection off a double wedge. The rectangular computational domain is  $[0, 4] \times [0, 2]$ . The embedded geometry is described by the function

$$y(x) = \begin{cases} 2(x - 2.00025) & : y \leq 1, \\ 2 - 2(x - 2.00025) & : y > 1. \end{cases}$$

We assume that this is a reflecting boundary.

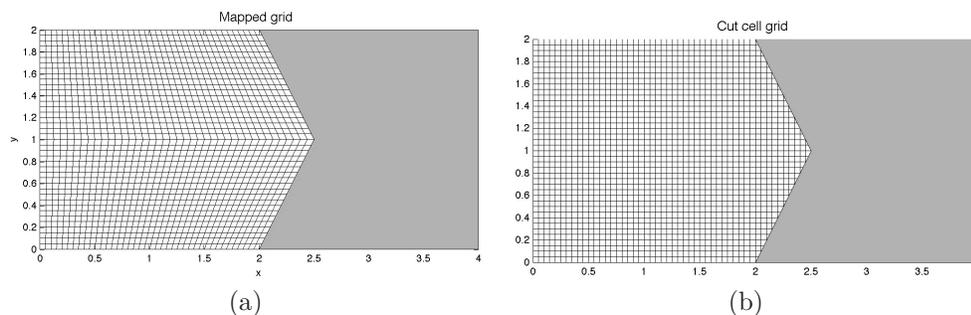


FIG. 6.2. Coarse versions of the (a) mapped grid and (b) cut cell mesh.

The regular grid cells have the lengths  $\Delta x = \Delta y = 1/100$  and  $\Delta x = \Delta y = 1/200$ , respectively. There are two very small triangular-shaped grid cells located in the corner. This configuration is analogous to the most challenging situation described in section 5.4. The area fraction (compared to a regular cell) is  $6.25 \times 10^{-4}$  on the coarser mesh and  $2.5 \times 10^{-3}$  on the finer mesh.

The flow domain is the region to the left of this boundary curve. Initially a Mach two shock wave is located at  $x = 1.5$ . The state in front of the shock is  $\rho = 1.4$ ,  $u = v = 0$ ,  $p = 1$ . The shock is moving to the right and is reflected from the embedded boundary. At a later time the two reflected shocks interact with each other; compare with Figure 6.3.

For this experiment gradients are computed using primitive variables, and the minmod limiter is used on the entire domain. The fluxes are computed using the HLL solver introduced by Harten, Lax, and van Leer [15]. For this computation we use the grid-aligned finite volume method in all regular mesh cells. For the flux calculation at the interface shared by the two tiny grid cells we use the nonorthogonal rotated grid cut cell method with the directions  $\xi = (2, -1)^T$  and  $\eta = (2, 1)^T$ , i.e., the two normal directions. For all the other cut cell fluxes, we use the orthogonal rotated grid cut cell method. We compute the average of the tiny grid cell with the neighboring cut cell in the horizontal direction, as explained in section 5.4.

To verify the accuracy of our computation, we compare the result with a computation on a mapped grid. The mapping is illustrated in Figure 6.2(a). We extend our method of lines finite volume method to mapped grids in an analogous manner as in [22, 5]. This problem (in particular on the mapped grid) is sensitive to the carbuncle problem; compare with, e.g., [26]. Therefore, we used the HLL Riemann solver, which does not resolve contact discontinuities.

As a reference solution, we use a highly refined mapped grid with  $1000 \times 1000$  grid cells. Figure 6.3 shows contour lines of density at four different times. In Figure 6.4 we plot the density at time  $t = 0.6$  along the reflecting double wedge as a function of the distance from the corner point at  $(2.50025, 1)$ . Negative values of the  $x$ -axis correspond to the lower part of the boundary segment (i.e.,  $y \leq 1$ ), and positive values correspond to the upper part. The solid line is obtained from the refined reference solution. On top of this reference solution we plot the results from coarser mapped grid computations. These can be compared with the results which we obtain for our cut cell computation; see Figure 6.6.

We now compute the problem on a cut cell mesh. Figure 6.5 shows contour plots of density at two different times obtained on cut cell grids with  $\Delta x = \Delta y = 1/100$  and

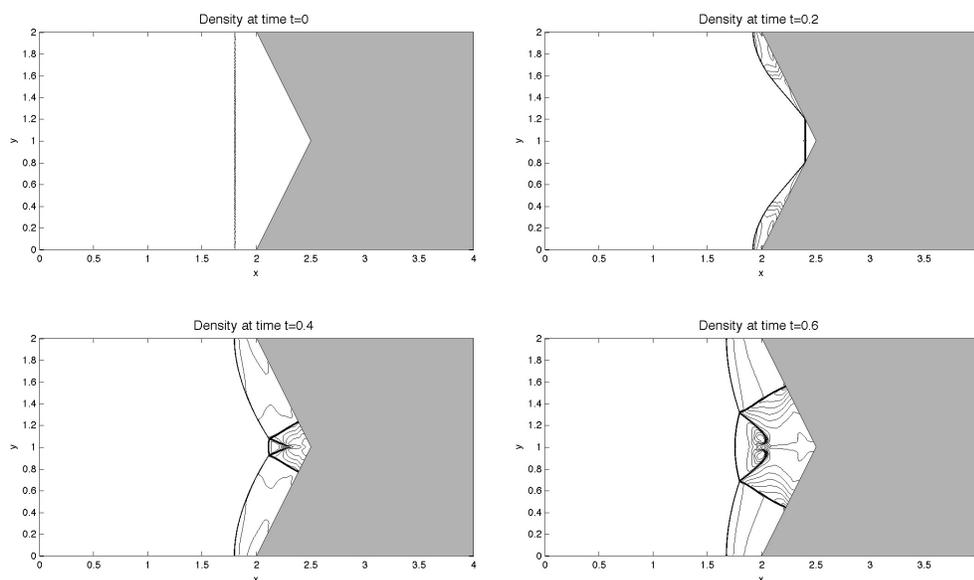


FIG. 6.3. Contour plots of density at different times computed on a mapped grid with  $1000 \times 1000$  grid cells.

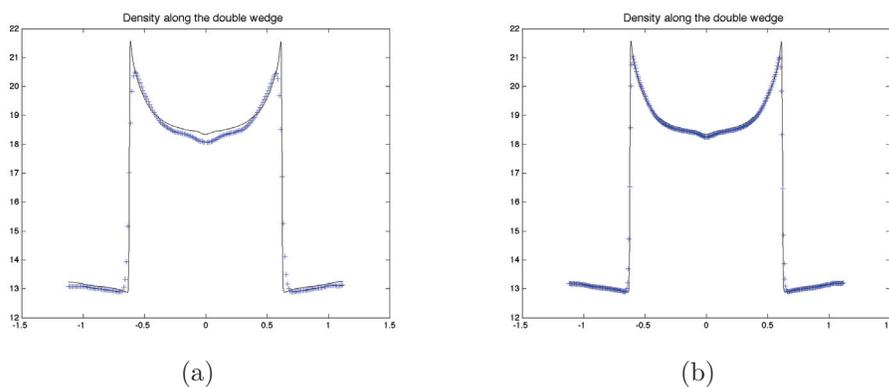


FIG. 6.4. Density along the double wedge at time  $t = 0.6$  computed on a mapped grid. The solid line is obtained from the refined reference solution. In (a) we show results from a computation using  $200 \times 200$  grid cells; in (b) we show results using  $400 \times 400$  grid cells.

$\Delta x = \Delta y = 1/200$ , respectively. In Figure 6.6 we plot the density in the cut cells along the embedded boundary and again compare these values with the refined mapped grid computation. Our results show that the accuracy obtained in the cut cells along the embedded boundary compares well with that of the mapped grid approach using a comparable resolution.

Finally, in Figure 6.7 we show contour plots of density at two different times using our cut cell method with the less diffusive Harten–Lax–van Leer contact (HLLC) Riemann solver (introduced by Toro, Spruce, and Speares [31]). This leads to a more highly resolved solution structure inside the domain without causing problems for the cut cell update.

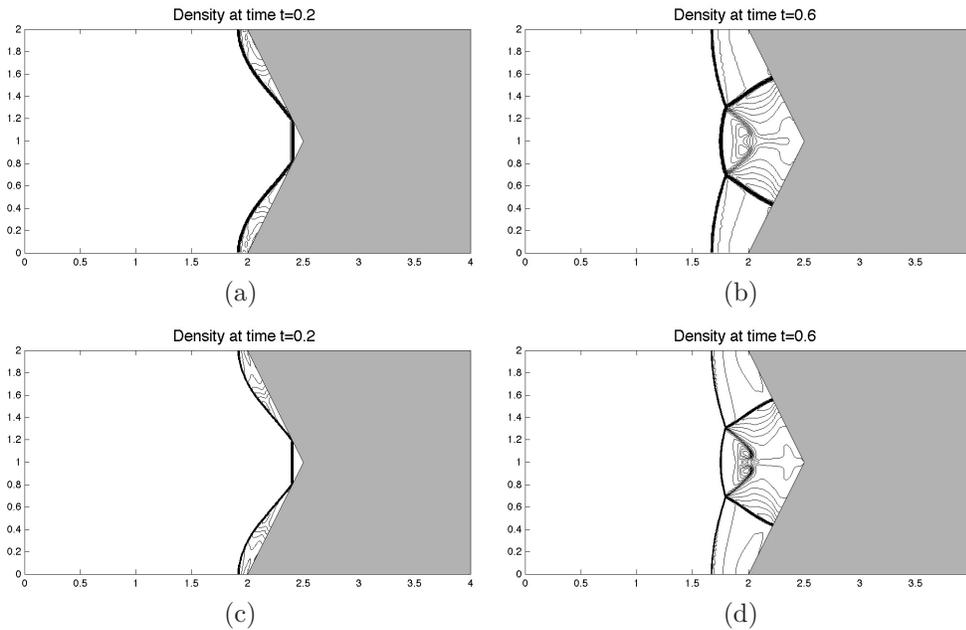


FIG. 6.5. Contour plots of density at two different times using the proposed cut cell method. In the regular grid cells, the grid-aligned finite volume method is used. For the cut cell fluxes away from the concave corner, the orthogonal rotated grid method is used. The two small triangular grid cells in the corner are updated using the nonorthogonal rotated grid method. (a), (b) Using  $\Delta x = \Delta y = 1/100$ . (c), (d) Using  $\Delta x = \Delta y = 1/200$ .

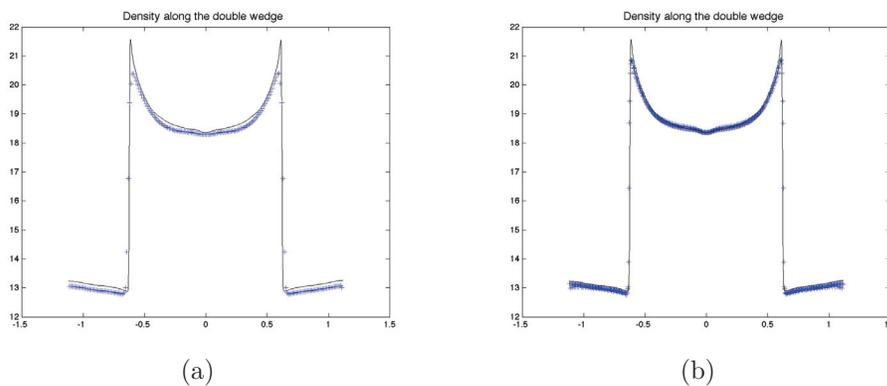


FIG. 6.6. Density along the cut cells at time  $t = 0.6$ . The solid line is obtained from the refined mapped grid reference solution. In (a) we show results from a cut cell computation using  $\Delta x = \Delta y = 1/100$ ; in (b) we show results using  $\Delta x = \Delta y = 1/200$ .

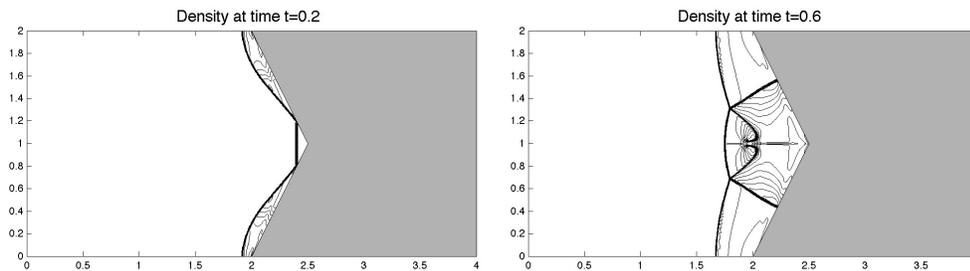


FIG. 6.7. Contour plots of density at two different times computed with the cut cell method by using the HLLC Riemann solver. The regular grid cells have the size  $\Delta x = \Delta y = 1/200$ .

**Conclusions.** We have presented a Cartesian grid embedded boundary method for hyperbolic PDEs based on a method of lines approach using SSP Runge–Kutta integration in time. Using a spatial reconstruction based on a rotated grid  $h$ -box approach, our method is stable for a time step based on the regular grid size, and is linearly exact for the arbitrarily small cut cells that appear where the Cartesian grid intersects the boundary. With a second order SSP method, the overall scheme retains second order accuracy. The method of lines approach leads to several simplifications and a much reduced stencil size compared to the MUSCL-type approach considered previously.

The introduction of the nonorthogonal rotated grid approach makes the method more flexible for nonsmoothly varying embedded boundaries. This needs to be explored for more complicated geometries in two dimensions. The extension of this algorithm for three-dimensional geometries is likely to add many programming complications.

We would also like to investigate extensions to incompressible flow. The use of the “virtual grid” to regularize the difference scheme at the cut cells might be especially useful for the approximation of higher derivatives in viscous flows. Furthermore, we are also working on higher order spatial discretizations for  $h$ -box methods. Finally, the method of lines approach allows the use of advanced splitting techniques, which can, for instance, be used to approximate problems with very different time scales.

#### REFERENCES

- [1] T. BARTH AND P. FREDERICKSON, *Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction*, AIAA Paper 90-0013, American Institute of Aeronautics and Astronautics, Reston, VA, 1990.
- [2] S.A. BAYYUK, K.G. POWELL, AND B. VAN LEER, *A Simulation Technique for 2-D Unsteady Inviscid Flows around Arbitrarily Moving and Deforming Bodies of Arbitrary Geometry*, AIAA Paper 93-2291-CP, American Institute of Aeronautics and Astronautics, Reston, VA, 1993.
- [3] M.J. BERGER, C. HELZEL, AND R.J. LEVEQUE,  *$h$ -box methods for the approximation of hyperbolic conservation laws on irregular grids*, SIAM J. Numer. Anal., 41 (2003), pp. 893–918.
- [4] M.J. BERGER AND R.J. LEVEQUE, *Stable boundary conditions for Cartesian grid calculations*, Comput. Systems Engrg., 1 (1990), pp. 305–311.
- [5] D.A. CALHOUN, C. HELZEL, AND R.J. LEVEQUE, *Logically rectangular grids and finite volume methods for PDEs in circular and spherical domains*, SIAM Rev., 50 (2008), pp. 723–752.
- [6] I.L. CHERN AND P. COLELLA, *A Conservative Front Tracking Method for Hyperbolic Conservation Laws*, LLNL Report UCRL-97 200, Lawrence Livermore National Laboratory, Livermore, CA, 1987.
- [7] W. COIRIER AND K. POWELL, *An accuracy assessment of Cartesian-mesh approaches for the Euler equations*, J. Comput. Phys., 117 (1995), pp. 121–131.

- [8] P. COLELLA, D.T. GRAVES, B.J. KEEN, AND D. MODIANO, *A Cartesian grid embedded boundary method for hyperbolic conservation laws*, J. Comput. Phys., 211 (2006), pp. 347–366.
- [9] A. DADONE AND B. GROSSMAN, *Ghost-cell method for analysis of inviscid three-dimensional flows on Cartesian-grids*, Comput. & Fluids, 36 (2007), pp. 1513–1528.
- [10] J. FALCOVITZ, G. ALFANDAR, AND G. HANOCH, *A two-dimensional conservation law scheme for compressible flows with moving boundaries*, J. Comput. Phys., 138 (1997), pp. 83–102.
- [11] H. FORRER AND R. JELTSCH, *A high-order boundary treatment for Cartesian-grid methods*, J. Comput. Phys., 140 (1998), pp. 259–277.
- [12] S. GOTTLIEB, C.-W. SHU, AND E. TADMOR, *Strong stability-preserving high-order time discretization methods*, SIAM Rev., 43 (2001), pp. 89–112.
- [13] B. GUSTAFSSON, H.-O. KREISS, AND A. SUNDSTRÖM, *Stability theory of difference approximations for mixed initial boundary value problems*, Math. Comp., 26 (1972), pp. 649–689.
- [14] A. HARTEN, *High resolution schemes for hyperbolic conservation laws*, J. Comput. Phys., 49 (1983), pp. 357–397.
- [15] A. HARTEN, P.D. LAX, AND B. VAN LEER, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Rev., 25 (1983), pp. 35–61.
- [16] C. HELZEL, M.J. BERGER, AND R.J. LEVEQUE, *A high-resolution rotated grid method for conservation laws with embedded geometries*, SIAM J. Sci. Comput., 26 (2005), pp. 785–809.
- [17] S. JEBENS, O. KNOTH, AND R. WEINER, *Partially implicit peer methods for compressible Euler equations*, J. Comput. Phys., 230 (2011), pp. 4955–4974.
- [18] B. KEEN AND S. KARNI, *A second order kinetic scheme for gas dynamics on arbitrary grids*, J. Comput. Phys., 205 (2005), pp. 108–130.
- [19] R. KLEIN, K.R. BATES, AND N. NIKIFORAKIS, *Well-balanced compressible cut-cell simulation of atmospheric flow*, Philos. Trans. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci., 367 (2009), pp. 4559–4575.
- [20] M. KUPIAINEN AND B. SJÖGREEN, *A Cartesian embedded boundary method for the compressible Navier-Stokes equations*, J. Sci. Comput., 41 (2009), pp. 94–117.
- [21] R.J. LEVEQUE, *Wave propagation algorithms for multidimensional hyperbolic systems*, J. Comput. Phys., 131 (1997), pp. 327–353.
- [22] R.J. LEVEQUE, *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, Cambridge, UK, 2002.
- [23] D. LEVY, K. POWELL, AND B. VAN LEER, *Use of a rotated Riemann solver for the two-dimensional Euler equations*, J. Comput. Phys., 106 (1993), pp. 201–214.
- [24] R.B. PEMBER, J.B. BELL, P. COLELLA, AND W.Y. CRUTCHFIELD, *An adaptive Cartesian grid method for unsteady compressible flow in irregular regions*, J. Comput. Phys., 120 (1995), pp. 278–304.
- [25] J.J. QUIRK, *An alternative to unstructured grids for computing gas dynamic flows around arbitrarily complex two-dimensional bodies*, Comput. & Fluids, 23 (1994), pp. 125–142.
- [26] J.J. QUIRK, *A contribution to the great Riemann solver debate*, Internat. J. Numer. Methods Fluids, 18 (1994), pp. 555–574.
- [27] P.L. ROE AND D. SIDILKOVER, *Optimum positive linear schemes for advection in two and three dimensions*, SIAM J. Numer. Anal., 29 (1992), pp. 1542–1568.
- [28] C.-W. SHU AND S. OSHER, *Efficient implementation of essentially non-oscillatory shock-capturing schemes*, J. Comput. Phys., 77 (1988), pp. 439–471.
- [29] B. SJÖGREEN AND N.A. PETERSSON, *A Cartesian embedded boundary method for hyperbolic conservation laws*, Commun. Comput. Phys., 2 (2007), pp. 1199–1219.
- [30] A.K. TORNBERG, B. ENGQUIST, B. GUSTAFSSON, AND P. WAHLUND, *A new type of boundary treatment for wave propagation*, BIT, 46 (2006), pp. S145–S170.
- [31] E.F. TORO, M. SPRUCE, AND W. SPEARES, *Restoration of the contact surface in the HLL-Riemann solver*, Shock Waves, 4 (1994), pp. 25–34.
- [32] B. WENDROFF AND A.B. WHITE, *A supraconvergent scheme for nonlinear hyperbolic systems*, Comput. Math. Appl., 18 (1989), pp. 761–767.
- [33] S. XU, T. ASLAM, AND D. STEWART, *High resolution numerical simulation of ideal and nonideal compressible reacting flows with embedded internal boundaries*, Combust. Theory Model., 1 (1997), pp. 113–142.