Accuracy considerations of mixed explicit implicit schemes for embedded boundary meshes

Sandra May¹*, Marsha Berger²**, and Fabian Laakmann³***

¹ Department of Mathematics, TU Dortmund University, Vogelpothsweg 87, 44227 Dortmund, Germany
² Courant Institute, New York University, 251 Mercer Street, New York, NY 10012, United States
³ Mathematical Institute, University of Oxford, Woodstock Road, Oxford OX2 6GG, United Kingdom

We consider a mixed explicit implicit finite volume scheme for solving the advection equation on a Cartesian cut cell mesh: the scheme couples an implicit time stepping scheme on cut cells with an explicit time stepping scheme on cells away from the cut cells by means of flux bounding. We examine the accuracy of the resulting mixed scheme to see if a second-order explicit and a second-order implicit scheme result in a second-order mixed scheme.

© 2019 The Authors Proceedings in Applied Mathematics & Mechanics published by Wiley-VCH Verlag GmbH & Co. KGaA Weinheim

1 The mixed explicit implicit scheme

For flow simulations in complex geometry, the use of cut cell meshes is typically cheaper and faster than employing body-fitted grids. In this approach the geometry is simply cut out of a Cartesian background grid. This results in so called cut cells around the boundary of the embedded object. These cells have irregular shape and may be very small. In the context of solving time-dependent, hyperbolic partial differential equations, probably the biggest issue caused by this approach is the small cell problem – standard explicit time stepping schemes are not stable on the arbitrarily small cut cells if the time step is chosen according to the background mesh.

The mixed explicit implicit scheme solves this problem by using implicit time stepping on cut cells and explicit time stepping otherwise. In space, a first- or second-order finite volume scheme is used. In order to ensure mass conservation and stability, the time stepping schemes are coupled using so called flux bounding [1]. While the mixed scheme also works well in 2D and 3D, we will focus on 1D for this contribution for simplicity. For an extended discussion of the scheme and its properties and its accuracy in 2D (and 3D), we refer the reader to [1,2].

We consider the linear advection equation

\[ s_t + u s_x = 0, \quad s(x,0) = s_0(x), \quad u > 0 \text{ constant}, \]

on a bounded interval \( I \) with periodic boundary conditions. To examine the behavior of cut cells in 1D, we use the following model problem: we consider a uniform mesh on \( I \) with mesh width \( h \), which contains one cell of length \( \alpha h \) with \( \alpha \in (0, 1] \). This cell is labeled as cell ‘0’. The idea of flux bounding is illustrated in figure 1. The important aspect is how the explicit and the implicit scheme are coupled: by reusing in Step 2 at edges between cells \(-2\) and \(-1\) and between cells \(1\) and \(2\), respectively, the flux from the explicit scheme (that was used in the updates of cells \(-2\) and \(2\) in Step 1) one ensures mass conservation. Further, for coupling the explicit second-order MUSCL scheme with implicit Euler time stepping with piecewise constant data, one can show a TVD stability result that is independent of the size of the volume fraction \( \alpha \) [1].

For obtaining a second-order scheme, it is necessary to also upgrade the implicit time stepping scheme to second order. We choose the implicit Trapezoidal rule in time with slope reconstruction in space. The question is whether the resulting mixed scheme, which we refer to as MUSCL-Trap, has second-order accuracy, too.

Fig. 1: Flux bounding [1]: One first updates the cells away from cell 0 using an explicit scheme (edges marked in pink). Then, the neighborhood of the cut cell is updated: the update on cell 0 uses fluxes from an implicit scheme (edges marked in bold black), the update on cells \(-1\) and \(1\) (which are called transition cells) employ both explicit and implicit fluxes.

For flow simulations in complex geometry, the use of cut cell meshes is typically cheaper and faster than employing body-fitted grids. In this approach the geometry is simply cut out of a Cartesian background grid. This results in so called cut cells around the boundary of the embedded object. These cells have irregular shape and may be very small. In the context of solving time-dependent, hyperbolic partial differential equations, probably the biggest issue caused by this approach is the small cell problem – standard explicit time stepping schemes are not stable on the arbitrarily small cut cells if the time step is chosen according to the background mesh.

The mixed explicit implicit scheme solves this problem by using implicit time stepping on cut cells and explicit time stepping otherwise. In space, a first- or second-order finite volume scheme is used. In order to ensure mass conservation and stability, the time stepping schemes are coupled using so called flux bounding [1]. While the mixed scheme also works well in 2D and 3D, we will focus on 1D for this contribution for simplicity. For an extended discussion of the scheme and its properties and its accuracy in 2D (and 3D), we refer the reader to [1,2].

We consider the linear advection equation

\[ s_t + u s_x = 0, \quad s(x,0) = s_0(x), \quad u > 0 \text{ constant}, \]

on a bounded interval \( I \) with periodic boundary conditions. To examine the behavior of cut cells in 1D, we use the following model problem: we consider a uniform mesh on \( I \) with mesh width \( h \), which contains one cell of length \( \alpha h \) with \( \alpha \in (0, 1] \). This cell is labeled as cell ‘0’. The idea of flux bounding is illustrated in figure 1. The important aspect is how the explicit and the implicit scheme are coupled: by reusing in Step 2 at edges between cells \(-2\) and \(-1\) and between cells \(1\) and \(2\), respectively, the flux from the explicit scheme (that was used in the updates of cells \(-2\) and \(2\) in Step 1) one ensures mass conservation. Further, for coupling the explicit second-order MUSCL scheme with implicit Euler time stepping with piecewise constant data, one can show a TVD stability result that is independent of the size of the volume fraction \( \alpha \) [1].

For obtaining a second-order scheme, it is necessary to also upgrade the implicit time stepping scheme to second order. We choose the implicit Trapezoidal rule in time with slope reconstruction in space. The question is whether the resulting mixed scheme, which we refer to as MUSCL-Trap, has second-order accuracy, too.

For flow simulations in complex geometry, the use of cut cell meshes is typically cheaper and faster than employing body-fitted grids. In this approach the geometry is simply cut out of a Cartesian background grid. This results in so called cut cells around the boundary of the embedded object. These cells have irregular shape and may be very small. In the context of solving time-dependent, hyperbolic partial differential equations, probably the biggest issue caused by this approach is the small cell problem – standard explicit time stepping schemes are not stable on the arbitrarily small cut cells if the time step is chosen according to the background mesh.

The mixed explicit implicit scheme solves this problem by using implicit time stepping on cut cells and explicit time stepping otherwise. In space, a first- or second-order finite volume scheme is used. In order to ensure mass conservation and stability, the time stepping schemes are coupled using so called flux bounding [1]. While the mixed scheme also works well in 2D and 3D, we will focus on 1D for this contribution for simplicity. For an extended discussion of the scheme and its properties and its accuracy in 2D (and 3D), we refer the reader to [1,2].

We consider the linear advection equation

\[ s_t + u s_x = 0, \quad s(x,0) = s_0(x), \quad u > 0 \text{ constant}, \]

on a bounded interval \( I \) with periodic boundary conditions. To examine the behavior of cut cells in 1D, we use the following model problem: we consider a uniform mesh on \( I \) with mesh width \( h \), which contains one cell of length \( \alpha h \) with \( \alpha \in (0, 1] \). This cell is labeled as cell ‘0’. The idea of flux bounding is illustrated in figure 1. The important aspect is how the explicit and the implicit scheme are coupled: by reusing in Step 2 at edges between cells \(-2\) and \(-1\) and between cells \(1\) and \(2\), respectively, the flux from the explicit scheme (that was used in the updates of cells \(-2\) and \(2\) in Step 1) one ensures mass conservation. Further, for coupling the explicit second-order MUSCL scheme with implicit Euler time stepping with piecewise constant data, one can show a TVD stability result that is independent of the size of the volume fraction \( \alpha \) [1].

For obtaining a second-order scheme, it is necessary to also upgrade the implicit time stepping scheme to second order. We choose the implicit Trapezoidal rule in time with slope reconstruction in space. The question is whether the resulting mixed scheme, which we refer to as MUSCL-Trap, has second-order accuracy, too.

Fig. 1: Flux bounding [1]: One first updates the cells away from cell 0 using an explicit scheme (edges marked in pink). Then, the neighborhood of the cut cell is updated: the update on cell 0 uses fluxes from an implicit scheme (edges marked in bold black), the update on cells \(-1\) and \(1\) (which are called transition cells) employ both explicit and implicit fluxes.

For obtaining a second-order scheme, it is necessary to also upgrade the implicit time stepping scheme to second order. We choose the implicit Trapezoidal rule in time with slope reconstruction in space. The question is whether the resulting mixed scheme, which we refer to as MUSCL-Trap, has second-order accuracy, too.

Fig. 1: Flux bounding [1]: One first updates the cells away from cell 0 using an explicit scheme (edges marked in pink). Then, the neighborhood of the cut cell is updated: the update on cell 0 uses fluxes from an implicit scheme (edges marked in bold black), the update on cells \(-1\) and \(1\) (which are called transition cells) employ both explicit and implicit fluxes.

For obtaining a second-order scheme, it is necessary to also upgrade the implicit time stepping scheme to second order. We choose the implicit Trapezoidal rule in time with slope reconstruction in space. The question is whether the resulting mixed scheme, which we refer to as MUSCL-Trap, has second-order accuracy, too.
2 Accuracy considerations

Denote by $S^n_i$ the discrete solution on cell $i$ at time $t^n$. Let the MUSCL-Trap scheme be given by $S^{n+1}_i = \Phi(S^n_i, S^{n+1}_i)$ with

$$\Phi(S^n_i, S^{n+1}_i) = S^n_i - \frac{\Delta t}{h_i} \left[ F_{i+1/2}^{n+1/2, x}(S^n_i, S^{n+1}_i) - F_{i-1/2}^{n+1/2, x}(S^n_i, S^{n+1}_i) \right]$$

with $F_{i+1/2}^{n+1/2, x}$ representing $F_{i+1/2}^{n+1/2, M}$ (MUSCL flux) or $F_{i+1/2}^{n+1/2, T}$ (Trapezoidal flux), depending on $i$, and $h_i = \alpha h$ for $i = 0$ and $h_i = h$ otherwise. Choose $\Delta t$ such that $0 < \lambda = \frac{\alpha h}{h_i} \leq 1$, independent of $\alpha$.

**Definition 2.1** (One step error) Denote by $\tilde{s}^n_i$ the true cell averages on cell $i$ at time $t^n$. We define the one step error $L(\tilde{s}^n, \tilde{s}^{n+1})$, on cell $i$ for the time step $t^n \rightarrow t^{n+1}$ by

$$L(\tilde{s}^n_i, \tilde{s}^{n+1}_i) = \Phi(\tilde{s}^n_i, \tilde{s}^{n+1}_i) - \tilde{s}^{n+1}_i$$

Note that for standard schemes on uniform meshes the one step error is typically one order higher than the overall order of the scheme. One can show the following result [1,2]:

**Lemma 2.2** Consider MUSCL-Trap with unlimited slopes for the advection equation with smooth $s_0$ for the model problem. For the one step error, there holds:

- $\text{cell} = 1$ and 1: $L(\tilde{s}^0_i, \tilde{s}^{n+1}_i) = O(h^2)$ and $L(\tilde{s}^n_i, \tilde{s}^{n+1}_i) = O(h^2)$ due to the switch in the time stepping scheme and due to cell 0 having a different length than the other cells;
- $\text{cell} = 0$: $L(\tilde{s}^0_i, \tilde{s}^{n+1}_i) = O(h^2)$ due to cell 0 having a different length than the other cells;
- all other cells $i$: $L(\tilde{s}^n_i, \tilde{s}^{n+1}_i) = O(h^2)$ for $i = 2$ and $L(\tilde{s}^n_i, \tilde{s}^{n+1}_i) = O(h^3)$ otherwise.

Therefore, one might expect to see only a reduced accuracy of first order for the mixed scheme at time $T$. Fortunately, one can show the following result [2]:

**Theorem 2.3** Let the MUSCL-Trap scheme be stable with respect to the $L^1$ and the $L^\infty$ norm and use unlimited forward difference quotients for slope reconstruction. Then the scheme is second-order accurate with respect to the $L^1$ and the $L^\infty$ norm for the model problem for smooth data $s_0$.

This results also holds true if the number of cut cells (cells of type 0) increases with $O(h^{-1})$ as $h \rightarrow 0$. In other words: the result does not rely on a counting argument of the form that there are only $O(1)$ cut cells in the model problem. Instead, the proof exploits the structure of the one step error on irregular grids.

3 Numerical results

We support our theoretical findings by numerical results: we consider the model problem with $\alpha = 10^{-5}$, $\nu = 1$, and $\lambda = 0.8$ on $I = [0, 1 + \alpha h]$. The small cell is located in the interval $[0.5, 0.5 + \alpha h]$. The $L^1$ error has been normalized to adjust for the changing domain length. We use smooth initial data $s_0(x) = \sin(\frac{2\pi x}{1+\alpha h})$ and choose $T$ such that the test function is back to its original position.

**Table 1**: Error for unlimited mixed scheme MUSCL-Trap for taking one time step and computing up to time $T$.

| $h$      | $L^1$ error | order | $L^\infty$ error | order | $L^1$ error | order | $L^\infty$ error | order |
|----------|-------------|-------|------------------|-------|-------------|-------|------------------|-------|
| 1/80     | 2.39e-05    | –     | 2.28e-03         | –     | 2.10e-04    | –     | 1.68e-03         | –     |
| 1/160    | 2.95e-06    | 3.02  | 5.52e-04         | 2.04  | 5.68e-05    | 1.89  | 3.94e-04         | 2.09  |
| 1/320    | 3.65e-07    | 3.01  | 1.36e-04         | 2.02  | 1.48e-05    | 1.94  | 9.40e-05         | 2.07  |

The results for the errors after 1 time step and at time T, respectively, are shown in Table 1: for the one step error we observe second-order convergence in the $L^\infty$ norm due to limited accuracy of the cells around cell 0. Due to their number being $O(1)$, this does not affect the third order convergence in the $L^1$ norm. At time $T$, we observe second-order convergence both in the $L^1$ and in the $L^\infty$ norm, which is consistent with the statement of theorem 2.3.

References

[1] S. May and M. Berger. An explicit implicit scheme for cut cells in embedded boundary meshes, J. Sci. Comput., 71(3), 919-943 (2017).
[2] S. May and F. Laakmann. Explicit implicit finite volume schemes for complex geometry. In preparation.

© 2019 The Authors. *Proceedings in Applied Mathematics & Mechanics* published by Wiley-VCH Verlag GmbH & Co. KGaA Weinheim www.gamm-proceedings.com