Computers & Fluids 83 (2013) 164–169

Contents lists available at SciVerse ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

High-order remapping with piece-wise parabolic reconstruction

J. Velechovský [a], R. Liska [a,*], M. Shashkov [b]

[a] Czech Technical University in Prague, Faculty of Nuclear Sciences and Physical Engineering, Břehová 7, 115 19 Prague 1, Czech Republic
[b] Los Alamos National Laboratory, X-Computational Physics Division, Group XCP-4, Los Alamos, NM 87545, USA

ARTICLE INFO

Article history:
Received 17 January 2012
Received in revised form 4 June 2012
Accepted 5 June 2012
Available online 15 June 2012

Keywords:
Remapping
ALE
Piece-wise parabolic reconstruction
Barth–Jespersen limiter
Monotonicity

ABSTRACT

High-order remapping methods, using piece-wise parabolic reconstruction with different limiting techniques trying to keep monotonicity (defined in terms of bounds on remapped solution) in the neighborhood of discontinuities, are investigated and compared on cyclic remapping tests. Piece-wise parabolic remapping methods based on PPM and FCT approaches keep the solution bounds in all the cases. These methods provide more accurate results than the standard remapping method using piece-wise linear reconstruction, usually with Barth–Jespersen limiter.

1. Introduction

The Arbitrary Lagrangian Eulerian (ALE) method is an extension of Lagrangian hydrodynamical methods which allows to overcome the difficulties connected with moving Lagrangian mesh, which for some flow patterns, as e.g. shear or vortex flows, becomes distorted so much that the Lagrangian computation cannot continue. In such a case (or regularly after some number of Lagrangian time steps) the ALE method rezones the distorted mesh to a new smoother one and interpolates the conservative quantities (mass, momentum and energy) from the old mesh to the new one. The interpolation has to be conservative and is called remapping. The remapping methods [1–3] typically use piece-wise linear (or constant) reconstruction of the conserved quantities on the old mesh. During the remapping, the reconstruction is integrated over the new cells to get remapped quantities on the new mesh. The piece-wise constant reconstruction leads to the first-order remapping, while piece-wise linear reconstruction is second-order accurate in regions of smooth flow. Here we investigate the usage of piece-wise quadratic reconstruction for the remapping which should be third-order accurate.

It is well-known that using standard piece-wise linear reconstruction works fine in the regions of smooth flow, however produces overshoots and undershoots (or even oscillations) when employed for remapping in the vicinity of discontinuities. These monotonicity violations are usually treated by applying limiters which effectively reduce the slopes of linear reconstructions around discontinuities resulting in monotone remapping. In the remapping context the monotonicity requirement requests the remapped data to be monotone when the initial data are monotone. Monotonicity of remapping is usually formulated in terms of bounds preservation and is reasonably well understood for piece-wise linear reconstructions. Here we look into the issue of how to limit piece-wise parabolic reconstructions, so that remapping results will stay monotone, in bounds, where we use the standard definition of bounds. We try in the remapping context several types of limiting of piece-wise parabolic reconstruction [4–6]. Instead of limiters one can use flux corrected remapping [7,8] combining low-order and high-order numerical remapping fluxes in a way satisfying the bounds. An option to correct remapping results being out of bounds is to use repair techniques [3,2] which redistributes conservatively the quantities being out of bounds into the neighboring cells. Remapping with piece-wise quartic reconstruction, being the extension of the PPM method [6], has been investigated in [9] for ocean modeling.

The rest of the paper is organized as follows. Section 2 contains general introduction into the remapping in the flux form and definition of the remapping monotonicity. Different methods for piece-wise parabolic reconstruction as well as introduction to flux corrected approach for the remapping are described in Section 3. In the next section, the order of convergence and bound-preservation of the methods are verified numerically on a set of cyclic remapping tests for few types of smooth and discontinuous functions.

2. Remapping

In the selected approach the remapping can be divided into two stages. At the first stage the remapped quantity is reconstructed on

* Corresponding author.
E-mail address: liska@siduri.fjfi.cvut.cz (R. Liska).

0045-7930/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compfluid.2012.06.006

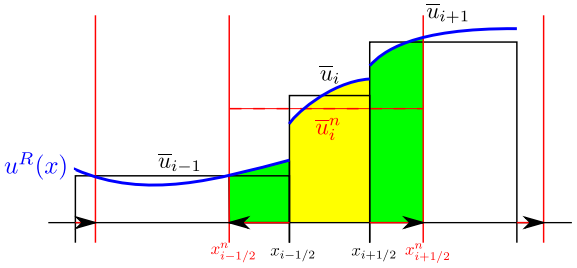

Fig. 1. Remapping in a single cell. The old computational mesh with nodes $x_{i\pm1/2}$ and means $\overline{u}_i$ are in black, the new ones are in red. The reconstructed piece-wise parabolic function $u(x)$ is in blue. The green areas correspond to the numerical fluxes $F_{i\pm1/2}$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the old mesh. At the second stage the reconstructed function is being integrated over all cells of the new mesh (see Fig. 1). We denote the nodes of the old mesh by $x_{i-1/2}$ and the nodes of the new mesh by $x^n_{i-1/2}$. The mean value $\overline{u}_i$ of the conserved quantity $u(x)$ in the old cell $(x_{i-1/2}, x_{i+1/2})$ is defined as

$$\overline{u}_i = \frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} u(x)\,\mathrm{d}x, \tag{1}$$

where $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ denotes the old cell volume. After the Lagrangian step one does not know the function $u(x)$. Only the old means $\overline{u}_i$ and old mesh $x_{i-1/2}$ are known. The new mesh $x^n_{i-1/2}$ is constructed during the rezone stage (here we assume that the rezoned nodes $x^n_{i-1/2}$ do not move outside the neighboring old cells, i.e. $x_{i-3/2} \leqslant x^n_{i-1/2} \leqslant x_{i+1/2}, \forall i$) and the remapping task is to compute the new means $\overline{u}^n_i$

$$\overline{u}^n_i \approx \frac{1}{\Delta x^n_i}\int_{x^n_{i-1/2}}^{x^n_{i+1/2}} u(x)\,\mathrm{d}x, \tag{2}$$

on the new mesh, where $\Delta x^n_i = x^n_{i+1/2} - x^n_{i-1/2}$ is the volume of the new cell. The quantity u is conservative, so we naturally require the remapping to be conservative, i.e.

$$\sum_i \overline{u}_i \Delta x_i = \sum_i \overline{u}^n_i \Delta x^n_i. \tag{3}$$

Standard way to proceed is to derive a piece-wise polynomial reconstruction function $u^R(x)$ from the old means $\overline{u}_i$ and the old mesh $x_{i-1/2}$ (the formula (1) has to be valid for $u^R(x)$ and all cells i). The reconstruction function $u^R(x)$ is then used in (2) to compute the new means $\overline{u}^n_i$. Now if we define the numerical remapping fluxes (corresponding to the green areas in Fig. 1) as

$$F_{i-1/2} = \begin{cases} \int_{x_{i-1/2}}^{x^n_{i-1/2}} u^R_i(x)\,\mathrm{d}x & \text{for } x^n_{i-1/2} > x_{i-1/2} \\ -\int_{x^n_{i-1/2}}^{x_{i-1/2}} u^R_{i-1}(x)\,\mathrm{d}x & \text{for } x^n_{i-1/2} < x_{i-1/2} \end{cases}, \tag{4}$$

where $u^R_i(x)$ stands for the polynomial reconstruction function in the cell i, then the remapping is given by (see Fig. 1)

$$\overline{u}^n_i \Delta x^n_i = \overline{u}_i \Delta x_i + F_{i+1/2} - F_{i-1/2}, \tag{5}$$

and it is conservative. To specify the monotonicity requirement we first define the bounds for the remapped values

$$u^{\min}_i = \min\{\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}\}, \quad u^{\max}_i = \max\{\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}\}. \tag{6}$$

We say that remapping is in bounds if for all cells i

$$\forall i \quad u^{\min}_i \leqslant \overline{u}^n_i \leqslant u^{\max}_i. \tag{7}$$

3. Piece-wise parabolic reconstruction

Inside each cell we choose a parabolic reconstruction

$$u^R_i(x) = u_i + u^x_i(x - x_i) + \frac{1}{2}u^{xx}_i(x - x_i)^2,$$

where x_i is the center of cell $x_i = (x_{i+1/2} + x_{i-1/2})/2$. The unknown coefficients of this reconstruction u_i, u^x_i, u^{xx}_i have to be computed from the old means $\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}$ on the old mesh. The conservation in the cell i implies $u_i = \overline{u}_i - \frac{1}{24}u^{xx}_i \Delta x_i^2$. The remaining unknown coefficients are computed by the least squares minimization of reconstruction error $\Phi(u^x_i, u^{xx}_i)$ in the neighboring cells $i-1$ and $i+1$, namely

$$\Phi(u^x_i, u^{xx}_i) = \sum_{j\in\{i-1,i+1\}} \left(\overline{u}_j - \frac{1}{\Delta x_j}\int_{x_{j-1/2}}^{x_{j+1/2}} u^R_i(x)\,\mathrm{d}x\right)^2.$$

This minimization, i.e. solution of the system $\frac{\partial\Phi(u^x_i,u^{xx}_i)}{\partial u^x_i} = 0, \frac{\partial\Phi(u^x_i,u^{xx}_i)}{\partial u^{xx}_i} = 0$ is equivalent to both zero contributions in the error sum $\Phi(u^x_i, u^{xx}_i)$ in the presented case (parabolic reconstruction in 1D). Resulting coefficients for the unlimited reconstruction are

$$u^x_i = 2\frac{(\overline{u}_{i+1} - \overline{u}_i)\Delta x_{i,i-1}\Delta_2 x_{i,i-1} + (\overline{u}_i - \overline{u}_{i-1})\Delta x_{i,i+1}\Delta_2 x_{i,i+1}}{\Delta x_{i,i-1}\Delta x_{i,i+1}(\Delta_2 x_{i,i-1} + \Delta_2 x_{i,i+1})}, \tag{8}$$

$$u^{xx}_i = 12\frac{(\overline{u}_{i+1} - \overline{u}_i)\Delta x_{i,i-1} - (\overline{u}_i - \overline{u}_{i-1})\Delta x_{i,i+1}}{\Delta x_{i,i-1}\Delta x_{i,i+1}(\Delta_2 x_{i,i-1} + \Delta_2 x_{i,i+1})}. \tag{9}$$

where $\Delta x_{i,i\pm1} = \Delta x_{i\pm1} + \Delta x_i, \Delta_2 x_{i,i\pm1} = 2\Delta x_{i\pm1} + \Delta x_i$. The reconstruction, as well as the complete remapping, is exact for quadratic function.

Limiting of these unlimited coefficients with respect to standard limiting of a piece-wise linear reconstruction is described in the following section.

3.1. Reconstruction coefficients limiting

Minmod (MM) limiter's extension to piece-wise parabolic reconstruction [10] is done as a sequential application of minmod function to the second derivative approximation and then to the first one, i.e.

$${}^m u^{xx}_i = \mathrm{minmod}\left(u^{xx}_i, \beta\frac{u^x_{i+1} - u^x_i}{\Delta x_{i,i+1}/2}, \beta\frac{u^x_i - u^x_{i-1}}{\Delta x_{i,i-1}/2}\right) \quad \text{, where } \beta \in (1,2).$$

In our tests we use $\beta = 1.5$. If the limiting of second derivative is not necessary (i.e. ${}^m u^{xx}_i = u^{xx}_i$), then we set also ${}^m u^x_i = u^x_i$, otherwise

$${}^m u^x_i = \mathrm{minmod}\left(u^x_i, \beta\frac{\overline{u}_{i+1} - \overline{u}_i}{\Delta x_{i,i+1}/2}, \beta\frac{\overline{u}_i - \overline{u}_{i-1}}{\Delta x_{i,i-1}/2}\right). \tag{10}$$

Note that standard piece-wise linear MM limited reconstruction uses this slope (10) everywhere. The final formula for MM limited parabolic reconstruction is

$$u^{\mathrm{MM}}_i(x) = \overline{u}_i + {}^m u^x_i(x - x_i) + \frac{1}{2}{}^m u^{xx}_i\left[(x - x_i)^2 - \frac{1}{12}\Delta x_i^2\right]. \tag{11}$$

Kuzmin–Barth–Jespersen [4] (KBJ) limiter is based on the Barth–Jespersen (BJ) limiter [11] $BJ(\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}, u^x_i, \Delta x_i)$ returning limiting factor $\alpha_i = \min\left(\alpha_i^{i-1/2}, \alpha_i^{i+1/2}\right)$ in the cell i. The limiting factors $\alpha_i^{i\pm1/2}$ at two nodes $i \pm 1/2$ of the cell i are given by

$$\alpha_i^{i\pm1/2} = \begin{cases} \min\left(1, \frac{u^{\max}_{i\pm1/2} - \overline{u}_i}{u^u_{i,i\pm1/2} - \overline{u}_i}\right) & \text{for } u^u_{i,i\pm1/2} - \overline{u}_i > 0 \\ 1 & \text{for } u^u_{i,i\pm1/2} - \overline{u}_i = 0 \\ \min\left(1, \frac{u^{\min}_{i\pm1/2} - \overline{u}_i}{u^u_{i,i\pm1/2} - \overline{u}_i}\right) & \text{for } u^u_{i,i\pm1/2} - \overline{u}_i < 0 \end{cases}, \tag{12}$$

where the bounds at the nodes are $u^{\min}_{i-1/2} = \min(\overline{u}_{i-1}, \overline{u}_i)$, $u^{\max}_{i-1/2} = \max(\overline{u}_{i-1}, \overline{u}_i)$ and the unlimited reconstruction from the cell i at the nodes are $u^u_{i,i-1/2} = u^u_i(x_{i-1/2}) = \overline{u}_i - u^x_i\Delta x_i/2, u^u_{i,i+1/2} = u^u_i(x_{i+1/2}) = \overline{u}_i + u^x_i\Delta x_i/2$. Now the parabolic KBJ limiter is defined as

$$
\begin{aligned}
\alpha_i^{xx} &= \mathrm{BJ}(u_{i-1}^x, u_i^x, u_{i+1}^x, u_i^{xx}, \Delta x_i),\\
\alpha_i^{xo} &= \mathrm{BJ}(\overline{u}_{i-1}, \overline{u}_i, \overline{u}_{i+1}, u_i^x, \Delta x_i),\\
\alpha_i^{x} &= \max(\alpha_i^{xo}, \alpha_i^{xx}),\\
{}^{B}u_i^{xx} &= \alpha_i^{xx} u_i^{xx},\\
{}^{B}u_i^{x} &= \alpha_i^{x} u_i^{x},
\end{aligned}
\tag{13}
$$

and the KBJ limited parabolic reconstruction is

$$
u_i^{\mathrm{KBJ}}(x) = \overline{u}_i + {}^{B}u_i^x (x - x_i) + \frac{1}{2}\,{}^{B}u_i^{xx}\left[(x - x_i)^2 - \frac{1}{12}\Delta x_i^2\right].
$$

The standard piece-wise linear KB limited reconstruction uses the slope $\alpha_i^{xo} u_i^x$.

Nejat–Venkatakrishnan (NV) limiter [5] for piece-wise parabolic reconstruction is described as

$$
\sigma = \frac{1 - \tanh(S(\phi_0 - \phi_i))}{2},
$$

where the constants are set to $\phi_0 = 0.8$ and $S = 20$, and ϕ_i is the Venkatakrishnan smooth limiter [12,13], which is a smooth extension of the BJ limiter (12) with function $\min(1, y)$ replaced by $\frac{y^2+2y}{y^2+y+2}$. The limited coefficients of the parabolic reconstruction are given by

$$
\begin{aligned}
{}^{NV}u_i^{xx} &= \sigma u_i^{xx},\\
{}^{NV}u_i^{x} &= [(1-\sigma)\phi_i + \sigma] u_i^{x},
\end{aligned}
$$

and the NV limited reconstruction is

$$
u_i^{\mathrm{NV}}(x) = \overline{u}_i + {}^{NV}u_i^x (x - x_i) + \frac{1}{2}\,{}^{NV}u_i^{xx}\left[(x - x_i)^2 - \frac{1}{12}\Delta x_i^2\right]. \tag{14}
$$

The standard piece-wise linear Venkatakrishnan limited reconstruction uses the slope $\phi_i u_i^x$.

3.2. PPM method

This method, referred under abbreviation PPM (Piece-wise Parabolic Method) [6], presents another way how to find appropriate reconstruction. Instead of searching second- and first-derivative approximations u_i^{xx} and u_i^x at the cell centers, the parabolic interpolant can be described by the values at the left u_i^l and right u_i^r nodes of the cell. There is a simple relation between these coefficients

$$
\begin{aligned}
u_i^x &= \frac{u_i^r - u_i^l}{\Delta x_i},\\
u_i^{xx} &= 12\,\frac{(u_i^r + u_i^l)/2 - \overline{u}_i}{\Delta x_i^2}.
\end{aligned}
\tag{15}
$$

The third-order nodal values approximations of u_i^l and u_i^r are constructed in a way which guarantees that they satisfy the bounds (7) from the neighboring cells. Initial guess for u_i^r, resp. u_{i+1}^l is given by

$$
\begin{aligned}
u_{i+1/2} = \overline{u}_i &+ \frac{\Delta x_i}{\Delta x_i + \Delta x_{i+1}}(\overline{u}_{i+1} - \overline{u}_i) + \frac{1}{\Delta x_{i-1} + \Delta x_i + \Delta x_{i+1} + \Delta x_{i+1}}\\
&\times \left\{ \frac{2\Delta x_{i+1}\Delta x_i}{\Delta x_i + \Delta x_{i+1}} \left[\frac{\Delta x_{i-1} + \Delta x_i}{2\Delta x_i + \Delta x_{i+1}} - \frac{\Delta x_{i+2} + \Delta x_{i+1}}{2\Delta x_{i+1} + \Delta x_i} \right] (\overline{u}_{i+1} - \overline{u}_i) \right.\\
&\left. - \Delta x_i \frac{\Delta x_{i-1} + \Delta x_i}{2\Delta x_i + \Delta x_{i+1}} \delta\overline{u}_{i+1} + \Delta x_{i+1} \frac{\Delta x_{i+1} + \Delta x_{i+2}}{\Delta x_i + 2\Delta x_{i+1}} \delta\overline{u}_i \right\},
\end{aligned}
\tag{16}
$$

where $\delta\overline{u}_i$ is the jump of u in the cell i and is computed as

$$
\delta\overline{u}_i = \begin{cases} 0 & \text{if } (\overline{u}_{i+1} - \overline{u}_i)(\overline{u}_i - \overline{u}_{i-1}) < 0,\\ \min(|\Delta x_i u_i^x|, 2|(\overline{u}_i - \overline{u}_{i-1})|, 2|(\overline{u}_{i+1} - \overline{u}_i)|)\,\mathrm{sign}(\Delta x_i u_i^x) & \text{otherwise,} \end{cases}
\tag{17}
$$

and u_i^x in previous expression comes from (8). This condition guarantees that $u_{i+1/2}$ stays in limits of $\overline{u}_i$ and $\overline{u}_{i+1}$ and leads to steeper representation of shocks [6].

Now one-side values u_i^l, u_i^r are initialized as $u_i^r = u_{i+1}^l = u_{i+1/2}$. To avoid oscillations in the reconstruction additional limiting is applied. If $(u_i^r - \overline{u}_i)(\overline{u}_i - u_i^l) \leqslant 0$ then $\overline{u}_i$ is local extrema (of the three values) and we set $u_i^r = u_i^l = \overline{u}_i$ so that the reconstruction is constant in this cell. Further we require the parabolic reconstruction to be monotone in each cell. If

$$
(u_i^r - u_i^l)\left(\overline{u}_i - \frac{1}{2}(u_i^r + u_i^l)\right) > \frac{(u_i^r - u_i^l)^2}{6},
$$

then the monotonicity is violated and we set $u_i^l = 3\overline{u}_i - 2u_i^r$ and similarly if

$$
-\frac{(u_i^r - u_i^l)^2}{6} > (u_i^r - u_i^l)\left(\overline{u}_i - \frac{1}{2}(u_i^r + u_i^l)\right),
$$

then we set $u_i^r = 3\overline{u}_i - 2u_i^l$. This gives us the final values of u_i^l, u_i^r which define the coefficients of PPM parabolic reconstruction (15). Another limiter for PPM, preserving accuracy at smooth extrema, has been proposed in [14] for reconstruction on uniform meshes.

3.3. Flux corrected approach

Flux corrected remapping is based on FCT (Flux Corrected Transport) method [15] modified for remapping. Assume we have low-order (usually donor from piece-wise constant reconstruction) remapping fluxes $F_{i+1/2}^L$ which preserve the bounds (7) and high-order (here from unlimited piece-wise parabolic reconstruction) remapping fluxes $F_{i+1/2}^H$. The flux corrected (FCT) remapping fluxes are given as a weighted combination of low-order and high-order fluxes

$$
F_{i+1/2}^{FCT} = F_{i+1/2}^L (1 - C_{i+1/2}) + C_{i+1/2} F_{i+1/2}^H, \tag{18}
$$

where the weights $C_{i+1/2} \in (0, 1)$ are constructed in a way that guarantees that the remapping with the FCT fluxes will stay in bounds.

Here we provide a short description showing how the FCT weights $C_{i+1/2}$ are computed. In the FCT context it is useful to introduce the numerical fluxes into the cell i from the neighboring cells $i \pm 1$ as

$$
G_{i,i+1}^P = F_{i+1/2}^P, \quad G_{i,i-1}^P = -F_{i-1/2}^P, \quad \text{for } P \in \{L, H, FCT\},
$$

so that the remapping (5) is written as

$$
\overline{u}_i^{n,P} \Delta x_i^n = \overline{u}_i \Delta x_i + \sum_{k=i\pm1} G_{i,k}^P \quad \text{for } P \in \{L, H, FCT\}.
$$

The FCT remapping can be transformed to

$$
\overline{u}_i^{n,FCT} \Delta x_i^n = \overline{u}_i^{n,L} \Delta x_i^n + \sum_{k=i\pm1} C_{i,k}\, dG_{i,k},
$$

where $dG_{i,k}$ are the anti-diffusive fluxes $dG_{i,k} = G_{i,k}^H - G_{i,k}^L$ and where for the FCT weights we use the notation $C_{i,i+1} = C_{i+1,i} = C_{i+1/2}$ to simplify the formula. We define the available space in the bounds for the low-order remapping as

$$
Q_i^{\max} = u_i^{\max} \Delta x_i^n - \overline{u}_i^{n,L} \Delta x_i^n \geqslant 0, \quad Q_i^{\min} = u_i^{\min} \Delta x_i^n - \overline{u}_i^{n,L} \Delta x_i^n \leqslant 0,
$$

and the total anti-diffusive fluxes incoming and outgoing from the cell i as

$$
P_i^+ = \sum_{k, dG_{i,k} > 0} dG_{i,k} \geqslant 0, \quad P_i^- = \sum_{k, dG_{i,k} < 0} dG_{i,k} \leqslant 0.
$$

The FCT weights are now given by

$$
\begin{aligned}
C_{i+1/2} &= \min\left(Q_i^{\max}/P_i^+, Q_{i+1}^{\min}/P_{i+1}^-, 1\right), \quad \text{for } dG_{i,i+1} > 0,\\
C_{i+1/2} &= \min\left(Q_i^{\min}/P_i^-, Q_{i+1}^{\max}/P_{i+1}^+, 1\right), \quad \text{for } dG_{i,i+1} < 0,
\end{aligned}
$$

where $|dG_{i,i+1}| > 0$ guarantees nonzero P^+ or P^- in the denominators. For zero anti-diffusive flux $dG_{i,i+1} = 0$ the low and high-order flux are the same and the weight $C_{i+1/2}$ can be arbitrary. For the full description of the weights computation, including also reasoning proving that FCT remapping satisfies the bounds, see [7] or [8]. The flux corrected approach is not constructing a reconstruction on the old mesh, it rather works directly with numerical fluxes, i.e. using the fluxes (18) in the flux form remapping (5).

4. Numerical tests and results

The remapping methods described in the previous section are compared on several tests. The tests include cyclic remapping of several smooth and discontinuous functions.

In the following figures and tables we use these abbreviations for different remapping methods: *donor* – remapping with piece-wise constant reconstruction, *pl.* – remapping with piece-wise linear reconstruction, *pp.* – remapping with piece-wise parabolic reconstruction, *unl.* – unlimited reconstruction (either pl. or pp.), *MM* – minmod limited pp. reconstruction (11) (with $\beta = 1.5$) *BJ* – Barth–Jespersen (12) limited pl. reconstruction, *KBJ* – Kuzmin–Barth–Jespersen (13) limited pp. reconstruction, *NV* – Nejat–Venkatakrishnan limited pp. reconstruction (14), *PPM* – pp. reconstruction by PPM method [6] (see Section 3.2), *FCT* – Flux Corrected Transport based (see Section 3.3) remapping using pp. unl. fluxes as high-order fluxes.

The only piece-wise parabolic reconstruction which always stay in bounds (i.e. (7) with $\overline{u}_i^n$ replaced by $u_i^R(x)$ holds for all x in each cell i) is the PPM reconstruction. For all the other reconstruction (i.e. pp. unl., pp. MM, pp. KBJ and pp. NV) it is not difficult to prepare an example (as a shock wave) for which the reconstruction in some cell does not stay in bounds. Having such example one can prepare a new mesh for which the single remapping will not satisfy the bounds (7). FCT remapping stays in bounds by its construction.

4.1. Cyclic remapping

Cyclic remapping [1,16] consists in repeated remapping of initial function given by means $\overline{u}_i^0$ on a prescribed sequence of meshes with n nodes $\{x^j_{i-1/2}, i = 1,\dots,n; j = 0,\dots,n_t\}$ satisfying at each pseudo-time step j the condition that the new nodes $x^j_{i-1/2}$ do not move outside two old neighboring cells, i.e. $x^{j-1}_{i-3/2} \leqslant x^j_{i-1/2} \leqslant x^{j-1}_{i+1/2}, \forall i,j$. At the j-th step the remapping computes the new means $\overline{u}_i^j$ from the old means $\overline{u}_i^{j-1}$, the old mesh $x^{j-1}_{i+1/2}$ and the new mesh $x^j_{i+1/2}$. We employ the mesh sequence $x^j_{i-1/2} = x(\xi_i, t^j)$ on the interval $x \in (0, 1)$ from Refs. [1,16] where

$$x(\xi,t) = (1-\alpha(t))\xi + \alpha(t)\xi^3, \quad \alpha(t) = \frac{\sin(4\pi t)}{2},$$
$$0 \leqslant \xi \leqslant 1; \;\; t_{min} \leqslant t \leqslant t_{max}, \tag{19}$$
$$t^j = t_{min} + \frac{j(t_{max}-t_{min})}{n_t}, \quad j = 0,\dots,n_t;$$
$$\xi_i = \frac{i-1}{n-1}, \;\; i = 1,\dots,n. \tag{20}$$

The number of time steps n_t for the unitary pseudo-time interval $(t_{min}, t_{max}) = (0, 1)$ is set to $n_t = 5n$ to satisfy an analog of the CFL condition. Two examples of these mesh sequences with a few cells are presented in Fig. 2.

To compare how much the bounds (7) are violated by different remapping methods in cyclic remapping we define the bounds violation error in the j-th time-step as L_1 violation

$$E^b_j = \sum_{i=1}^{n} \max\left(0, \overline{u}^j_i - {}^{\max}\overline{u}^{j-1}_i\right)\Delta x^j_i + \sum_{i=1}^{n} \max\left(0, {}^{\min}\overline{u}^{j-1}_i - \overline{u}^j_i\right)\Delta x^j_i,$$

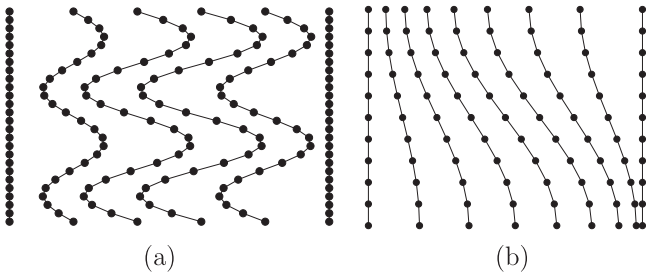

Fig. 2. Computational mesh sequence for 5 cells and 25 pseudo-time steps for $t \in (0, 1)$ (a) and for 8 cells and 10 pseudo-time steps for $t \in (-1/8, 1/8)$ (b). Horizontal axis corresponds to space axis x and vertical one to pseudo-time t.

where ${}^{\max}\overline{u}^j = \max\{\overline{u}^j_{i-1}, \overline{u}^j_i, \overline{u}^j_{i+1}\}$ a ${}^{\min}\overline{u}^j = \min\{\overline{u}^j_{i-1}, \overline{u}^j_i, \overline{u}^j_{i+1}\}$. The bounds violation error for the whole cyclic remapping is then L_1 error in pseudo-time

$$E^b = \sum_{j=1}^{n_t} E^b_j \Delta t, \tag{21}$$

where the pseudo-time step is $\Delta t = (t_{max} - t_{min})/n_t$.

During the cyclic remapping of smooth functions on the unitary pseudo-time interval $t \in (0, 1)$ the prescribed mesh movement and its return to the initial position leads to cancellation of some truncation errors and the piece-wise linear unlimited remapping method is showing the third-order of accuracy. To avoid this unwanted feature we have changed the pseudo-time interval to $t \in (-1/8, 1/8)$ on which this false super-convergence does not appear. For this pseudo-time interval we set the number of pseudo-time steps to $n_t = \frac{5}{4}n$.

For the pseudo-time interval $t \in (0, 1)$ the cyclic remapping of smooth functions gives the L_1 errors ratio 8.0 for piece-wise linear unlimited remapping, while for the other methods the results are close to those presented below for the pseudo-time interval $t \in (-1/8, 1/8)$. For discontinuous functions the cyclic remapping results for the pseudo-time interval $t \in (0, 1)$ are close to those for the pseudo-time interval $t \in (-1/8, 1/8)$ for all methods.

4.1.1. Smooth functions

For simple smooth monotone functions without local extrema as exponential e^x the cyclic remapping gives the expected results, donor shows the first-order convergence, all piece-wise linear methods are second-order accurate and all piece-wise parabolic methods are third-order, except pp. NV which is only second-order. All remapping methods stay in bounds in this case.

The situation is more complicated for remapping of smooth functions with local extrema. In Table 1 we present the convergence results for cyclic remapping of function $\sin(2\pi x) + 1$. Donor is first-order, piece-wise linear methods are second-order with pl. BJ staying in bounds and unlimited pl. violating the bounds. Unlimited, KBJ and MM piece-wise parabolic methods are third-order, however violate the bounds, while pp. NV is second-order and violates the bounds. Both pp. PPM and FCT are better than second-order and satisfy the bounds. FCT is slightly more precise than PPM and both are more precise than pl. BJ. In this case the bounds (7) are over-restrictive in the vicinity of local extrema and cause the loss of precision for bounds preserving pl. BJ, pp. PPM and pp. FCT methods.

We have tested also the cyclic remapping with random mesh movement [1,16], which results, for the non-monotone smooth function as $\sin(2\pi x) + 1$, in slower convergence for most methods. The ratio of errors on meshes with 256 and 512 nodes is 3.2 for pl. unl., 3.7 for pl. BJ, around 6 for pp. unl., KBJ and MM, 1.6 (worse than first-order) for pp. NV and around 5.5 for pp. PPM and FCT

Table 1
Cyclic remapping of $\sin(2\pi x)+1$ for pseudo-time $t \in (-1/8, 1/8)$ on meshes with $n = 64, 128, 256, 512$ nodes. Columns E^c_n present L_1 errors of cyclic remapping on mesh with n nodes, columns headed by ratios n_1/n_2 present the ratios of L_1 errors on meshes with n_1 and n_2 nodes and E^b_n columns present the bounds violation errors (21).

Remap	$\frac{64}{128}$	$\frac{128}{256}$	$\frac{256}{512}$	E^c_{64}	E^c_{512}	E^b_{64}	E^b_{512}
Donor	1.9	2.0	2.0	4.20×10^{-2}	5.53×10^{-3}	0	0
pl. unl.	4.1	4.0	4.0	3.99×10^{-4}	6.11×10^{-6}	7.0×10^{-6}	1.4×10^{-8}
pl. BJ	4.2	4.4	4.4	1.42×10^{-3}	1.72×10^{-5}	0	0
pp. unl.	8.0	8.0	8.0	9.66×10^{-5}	1.91×10^{-7}	7.0×10^{-6}	1.4×10^{-8}
pp. KBJ	7.9	8.0	8.0	9.66×10^{-5}	1.92×10^{-7}	7.0×10^{-6}	1.4×10^{-8}
pp. MM	7.9	8.0	8.0	9.75×10^{-5}	1.93×10^{-7}	7.0×10^{-6}	1.4×10^{-8}
pp. NV	3.7	4.0	4.1	4.47×10^{-4}	7.35×10^{-6}	5.4×10^{-6}	1.7×10^{-8}
pp. PPM	5.3	5.0	5.2	1.04×10^{-3}	7.52×10^{-6}	0	0
pp. FCT	5.6	5.6	5.5	4.39×10^{-4}	2.57×10^{-6}	0	0

Table 2
Cyclic remapping of single step for pseudo-time $t \in (-1/8, 1/8)$ n meshes with $n = 64, 128, 256, 512$ nodes. Columns E^c_n present L_1 errors of cyclic remapping on mesh with n nodes, columns headed by ratios n_1/n_2 present the ratios of L_1 errors on meshes with n_1 and n_2 nodes and E^b_n columns present the bounds violation errors (21).

Remap	$\frac{64}{128}$	$\frac{128}{256}$	$\frac{256}{512}$	E^c_{64}	E^c_{512}	E^b_{64}	E^b_{512}
Donor	1.4	1.4	1.4	1.41×10^{-1}	5.38×10^{-2}	0	0
pl. unl.	1.5	1.6	1.5	5.03×10^{-2}	1.32×10^{-2}	7.7×10^{-5}	3.5×10^{-6}
pl. BJ	1.5	1.6	1.6	3.85×10^{-2}	9.65×10^{-3}	0	0
pp. unl.	1.6	1.6	1.6	4.97×10^{-2}	1.19×10^{-2}	9.1×10^{-5}	4.0×10^{-6}
pp. KBJ	1.6	1.6	1.7	5.05×10^{-2}	1.20×10^{-2}	6.5×10^{-5}	3.2×10^{-6}
pp. MM	1.6	1.6	1.7	4.22×10^{-2}	9.71×10^{-3}	2.7×10^{-6}	1.7×10^{-8}
pp. NV	1.6	1.6	1.6	4.14×10^{-2}	1.03×10^{-2}	3.7×10^{-5}	1.5×10^{-6}
pp. PPM	1.7	1.6	1.7	2.45×10^{-2}	5.32×10^{-3}	0	0
pp. FCT	1.6	1.6	1.6	3.60×10^{-2}	8.88×10^{-3}	0	0

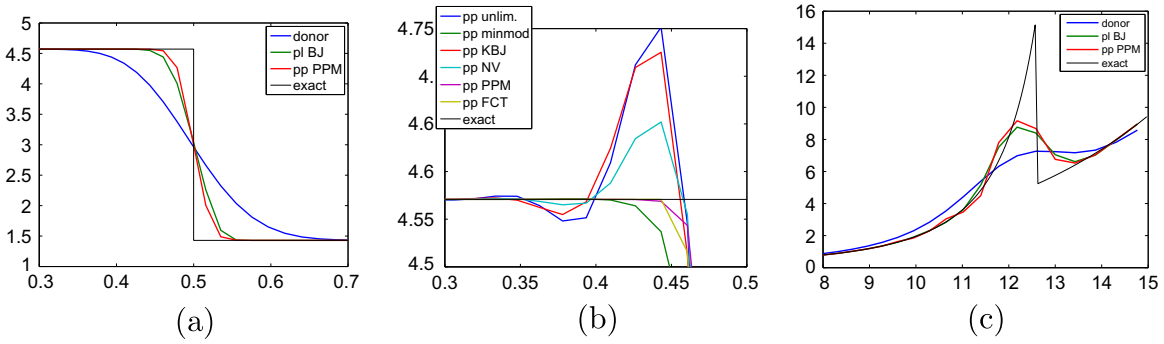

Fig. 3. Cyclic remapping results of step function for pseudo-time $t \in (-1/8, 1/8)$ on mesh with 16 nodes for donor, pl. BJ and pp. PPM methods (a) and for all pp. methods zoomed on the left of the step where bounds violating methods have overshoots (b). Cyclic remapping results of exponential shock for pseudo-time $t \in (-1/8, 1/8)$ and on mesh with 64 nodes for donor, pl. BJ and pp. PPM methods (c).

(compare with Table 1 for smooth mesh movement). Note that bounds preserving pp. PPM and FCT convergence is not slower compared to smooth mesh movement. Bounds violation error E^b_{512} is of the order 5×10^{-9}, similar for all bounds violating methods (pl. unl., pp. unl., KBJ, MM, NV).

4.1.2. Step function

Table 2 and Fig. 3(a) and (b) present results for cyclic remapping of a step function $u(x) = 3 + \pi/2$ for $x < 1/2$ and $u(x) = 3 - \pi/2$ for $x > 1/2$ (which is the limit of the function $u(x) = 3 + \arctan(c(1/2 - x))$ for large c; this function has been used in other tests with moderate values of c) emulating a shock wave. The convergence rate is very similar for all methods, however the errors are different for various methods. The best is pp. PPM followed by pp. FCT and pl. BJ. All these three methods stay in bounds. All other pp. methods (i.e. unl., KBJ, MM and NV) violate the bounds, however note that bounds violation for pp. MM is much less than that for the other three bounds violating methods and pp. MM error is very close to the error of pl. BJ (and further pp. MM is third-order for smooth function, see Table 1). The bounds violation of pp. MM is not visible in Fig. 3(b) on mesh with only 16 nodes.

Convergence results of cyclic remapping for step function with random mesh movement are in all aspects similar to those presented in Table 2 for smooth mesh movement.

4.1.3. Exponential shock

For the exponential shock test [7,8] the initial profile is given by

$$u(x) = \begin{cases} 3\rho_0 \exp\left(\frac{x_F - x_0}{\delta}\right)(1+2\eta)^{-5/2} & \text{for } 0 \leqslant x \leqslant x_F \\ \rho_0 \exp\left(\frac{x - x_0}{\delta}\right) & \text{for } x_F < x \leqslant 15 \end{cases}, \tag{22}$$

Table 3
Cyclic remapping of exponential shock for pseudo-time $t \in (-1/8, 1/8)$ on meshes with $n = 64, 128, 256, 512$ nodes. Columns E_n^c present L_1 errors of cyclic remapping on mesh with n nodes, columns headed by ratios n_1/n_2 present the ratios of L_1 errors on meshes with n_1 and n_2 nodes and E_n^b columns present the bounds violation errors (21).

Remap	$\frac{64}{128}$	$\frac{128}{256}$	$\frac{256}{512}$	E_{64}^c	E_{512}^c	E_{64}^b	E_{512}^b
Donor	1.1	1.3	1.4	3.40×10^{-1}	1.61×10^{-1}	0	0
lin. unl.	0.9	1.6	1.7	1.11×10^{-1}	4.35×10^{-2}	2.8×10^{-4}	1.6×10^{-5}
lin. BJ	1.1	1.8	1.8	1.42×10^{-1}	4.06×10^{-2}	0	0
pp unl.	1.0	1.6	1.7	1.14×10^{-1}	3.91×10^{-2}	2.9×10^{-4}	1.7×10^{-5}
pp KBJ	1.0	1.6	1.7	1.15×10^{-1}	3.93×10^{-2}	2.2×10^{-4}	1.6×10^{-5}
pp MM	1.2	1.8	1.9	1.45×10^{-1}	3.76×10^{-2}	9.7×10^{-5}	9.0×10^{-6}
pp NV	1.1	1.7	1.9	1.25×10^{-1}	3.60×10^{-2}	1.4×10^{-4}	7.8×10^{-6}
pp PPM	1.1	1.8	2.1	1.30×10^{-1}	3.09×10^{-2}	0	0
pp FCT	1.1	1.7	1.9	1.31×10^{-1}	3.58×10^{-2}	0	0

with $x_F = x_0 + 3/2\delta \log(t'/t_0), \eta = (x_F - x)/\delta, t_0 = 2, x_0 = 6, \rho_0 = 1, \delta = 4, t' = 6$. The convergence of cyclic remapping for the exponential shock, presented in Table 3, is similar to the convergence for step function, only the exponential shock needs finer mesh to start to converge (errors on under-resolved meshes with 64 and 128 nodes are almost the same) and pp. MM bounds violation errors are similar to those of the bounds violating pp. methods (unl., KBJ and NV). Fig. 3c shows results of cyclic remapping of exponential shock on a mesh with 64 nodes with donor, pl. BJ and pp. PPM methods.

5. Conclusion

Several high-order remapping methods based on piece-wise parabolic reconstruction and using different limiting approaches to preserve bounds (monotonicity) has been presented. The cyclic remapping of monotone smooth functions by these methods is third-order accurate. The only two methods preserving bounds for remapping of discontinuous functions are PPM and FCT methods. The accuracy of bounds preserving PPM and FCT methods for cyclic remapping of smooth functions with local extrema decreases around extrema due to over-restrictive bounds. One would need a better bounds in vicinity of local extrema, however it is difficult to distinguish local extrema of smooth function from local extrema in numerical oscillations around discontinuities.

We have made preliminary experiments with applying the remapping with piece-wise parabolic reconstruction in a full 1D ALE method based on the cell-centered scheme [17] and the results are promising.

The methods described in Section 3.1 (MM, KBJ, NV) can be generalized to multiple dimensions, however their results do not stay in bounds on discontinuities. Generalization of PPM reconstruction would be very complicated even in 2D on e.g. non-orthogonal quadrilateral cells. Piece-wise parabolic FCT based remapping method is easily extensible to multiple dimensions as its numerical flux construction is local at each interface between two computational cells [7].

Acknowledgments

We thank Pierre-Henri Maire for fruitful discussion and anonymous reviewers for their constructive comments. The work of the last author was performed under the auspices of the US Department of Energy's National Nuclear Security Administration by Los Alamos National Security, LLC, at Los Alamos National Laboratory, under contract DE-AC52-06NA25396. The authors gratefully acknowledge the partial support of the US DOE NNSA's Advanced Simulation and Computing (ASC) Program and the partial support of the US DOE Office of Science Advanced Scientific Computing Research (ASCR) Program in Applied Mathematics Research. The first and the second author have been supported in part by the Czech Science Foundation Grant P205/10/0814, Czech Ministry of Education grants MSM 6840770022, LC528, RVO 68407700 and Czech Technical University Grant SGS10/299/OHK4/3T/14.

References

[1] Margolin L, Shashkov M. Second-order sign-preserving conservative interpolation (remapping) on general grids. J Comput Phys 2003;184:266–98.
[2] Kuchařík M, Shashkov M, Wendroff B. An efficient linearity-and-bound-preserving remapping method. J Comput Phys 2003;188(2):462–71.
[3] Loubère R, Shashkov M. A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian–Eulerian methods. J Comput Phys 2005;209(1):105–38.
[4] Kuzmin D. A vertex-based hierarchical slope limiter for p-adaptive discontinuous Galerkin methods. J Comput Appl Math 2010;233(12):3077–85.
[5] Nejat A. A higher-order accurate unstructured finite volume Newton–Krylov algorithm for inviscid compressible flows. PhD thesis, University of British Columbia; 2007.
[6] Colella P, Woodward P. The piecewise parabolic method (PPM) for gas-dynamical simulations. J Comput Phys 1984;54:174–201.
[7] Liska R, Shashkov M, Váchal P, Wendroff B. Optimization-based synchronized flux-corrected conservative interpolation (remapping) of mass and momentum for arbitrary Lagrangian–Eulerian methods. J Comput Phys 2010;229(5):1467–97.
[8] Liska R, Shashkov M, Váchal P, Wendroff B. Synchronized flux corrected remapping for ALE methods. Comput Fluids 2011;46:312–7.
[9] White L, Adcroft A. A high-order finite volume remapping scheme for nonuniform grids: the piecewise quartic method (PQM). J Comput Phys 2008;227:7394–422.
[10] Yang M, Wang Z. A parameter-free generalized moment limiter for high- order methods on unstructured grids. In: Booktitle 47th AIAA aerospace sciences meeting including the new horizons forum and aerospace exposition. No. 2009-605 in series AIAA. AIAA; 2009. p. 1–22.
[11] Barth T, Jespersen D. The design and application of upwind schemes on unstructured meshes. Tech. rep. AIAA-89-0366. AIAA, NASA Ames Research Center; 1989.
[12] Venkatakrishnan V. Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. J Comput Phys 1995;118:120–30.
[13] Michalak C, Ollivier-Gooch C. Accuracy preserving limiter for the high-order accurate solution of the Euler equations. J Comput Phys 2009;228:8693–711.
[14] Colella P, Sekora M. A limiter for PPM that preserves accuracy at smooth extrema. J Comput Phys 2008;227:7069–76.
[15] Boris J, Book D. Flux-corrected transport I: SHASTA, a fluid transport algorithm that works. J Comput Phys 1973;11:38–69.
[16] Margolin LG, Shashkov M. Second-order sign-preserving remapping on general grids. Tech. rep. LA-UR-02-525. Los Alamos National Laboratory; 2002.
[17] Maire PH. A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. J Comput Phys 2009;228(7):2391–425.