

# Constrained-Optimization Based Data Transfer: A New Perspective on Flux Correction.

Pavel Bochev, Denis Ridzal, Guglielmo Scovazzi, and Mikhail Shashkov

**Abstract** We formulate a new class of optimization-based methods for data transfer (remap) of a scalar conserved quantity between two close meshes with the same connectivity. We present the methods in the context of the remap of a mass density field, which preserves global mass (the integral of the density over the computational domain). The key idea is to formulate remap as a global inequality-constrained optimization problem for mass fluxes between neighboring cells. The objective is to minimize the discrepancy between these fluxes and the given high-order *target mass fluxes*, subject to constraints that enforce physically motivated bounds on the associated primitive variable. In so doing, we separate accuracy considerations, handled by the objective functional, from the enforcement of physical bounds, handled by the constraints. The resulting second-order, conservative, and bound-preserving optimization-based remap (OBR) formulation is applicable to general, unstructured, heterogeneous grids. Under some weak requirements on grid proximity we prove that the OBR algorithm preserves linear fields in one, two and three dimensions. The chapter also examines connections between the OBR and the flux-corrected remap

---

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Pavel Bochev

Numerical Analysis and Applications, Sandia National Laboratories, MS-1320, Albuquerque, NM 87185-1320, USA, e-mail: pbboche@sandia.gov

Denis Ridzal

Optimization and Uncertainty Quantification, Sandia National Laboratories, MS-1320, Albuquerque, NM 87185-1320, USA, e-mail: dridzal@sandia.gov

Guglielmo Scovazzi

Numerical Analysis and Applications, Sandia National Laboratories, MS-1319, Albuquerque, NM 87185-1319, USA, e-mail: gscovaz@sandia.gov

Mikhail Shashkov

XCP-4, Methods and Algorithms, Los Alamos National Laboratory, MS-F644, Los Alamos, NM 87545, USA, e-mail: shashkov@lanl.gov

(FCR), which can be interpreted as a modified version of OBR (M-OBR), with the same objective but a smaller feasible set. The feasible set for M-OBR (FCR) is given by simple box constraints derived by using a “worst-case” scenario approach, which may result in loss of linearity preservation and ultimately accuracy for some grid motions. The optimality of the OBR solution means that, given a set of target fluxes and a distance measure, OBR finds the best possible approximations of these fluxes with respect to this measure, which also satisfy the physically motivated bounds. In this sense, OBR can serve as a natural benchmark for evaluating the accuracy of existing and future numerical methods for data transfer with respect to a given class of flux reconstruction methods and flux distance measures. In this context, we perform numerical comparisons between OBR, FCR and iFCR (a version of FCR which utilizes an iterative procedure to enhance the accuracy of FCR numerical fluxes).

## 1 Introduction

The problem of transferring data between computational grids under specific constraints arises in the computational sciences in many contexts (see, e.g., Laursen and Heinstejn (2003); Bochev and Day (2008); Carey et al (2001)). Among the main applications, we focus on Arbitrary Lagrangian-Eulerian (ALE) methods (see Hirt et al (1974)) as the primary motivation for this work.

ALE methods based on so-called continuous remap involve three separate phases: (i) the Lagrangian update of the solution, including displacements of the computational grid; (ii) rezoning (repositioning) of the computational grid in order to reduce grid distortion accrued during the Lagrangian motion; and (iii) conservative interpolation (remap) of the Lagrangian solution onto the rezoned grid. Formally, it is possible to run ALE algorithms primarily in the Lagrangian mode with the occasional rezone/remap taking place only when the grid becomes too distorted. However, an alternative computational strategy that combines the best properties of Eulerian and Lagrangian methods is to perform rezoning and remapping at every time step (from which the terminology, continuous remapping).

An important property of the *continuous rezone* strategy is that individual grid movements can be limited to small perturbations of the Lagrangian (old) mesh, and, in turn, that conserved quantities are exchanged only between neighboring cells. In this case, the remap step is localized to neighborhoods of old mesh cells and eliminates expensive global search operations required to locate new cells in the old mesh. Note also that, since remap is performed at every time step, the accuracy of the continuous-rezone ALE strongly depends on the quality of the remap phase.

In what follows, we focus on the second-order conservative and bound-preserving remap of a scalar conserved quantity between two close meshes with the same connectivity. On each cell of the old mesh we are given the mean value of the primitive variable that is an otherwise unknown positive scalar function (“density”). The conserved variable is the product of this mean value and the cell volume (“mass”). The

objective is to find an accurate approximation of the conserved variable on the new mesh such that the density, approximated by the remapped cell mass divided by the volume of the new cell, satisfies physically motivated bounds. In summary, we seek solutions to the remap problem which possess the following properties:

- P1. Conservation of total mass;
- P2. Preservation of linearity;
- P3. Preservation of local bounds for the primitive variable (namely, density).

Specifically, property (P1) is a fundamental requirement for remap, while property (P2) is a statement of accuracy. It requires the remap algorithm to recover exact masses in the new cells whenever the old masses correspond to a linear density function. Property (P3) accounts for the fact that physically motivated bounds are imposed on the primitive variable rather than on the conserved quantity. In the continuous rezone setting, every new cell is contained in the union of its Lagrangian prototype and its neighbors. The minimum and maximum mean density values on these Lagrangian cells provide natural lower and upper bounds for the mean density value on the new cell.

Conservation of total mass (P1) is guaranteed if the remap is discretely stated in mass flux form, as indicated by Margolin and Shashkov (2003).

Two strategies are commonly used in existing remappers to fulfill (P2) and (P3). The first one employs slope-limited bound-preserving reconstruction of the primitive variable, as presented in Dukowicz and Kodis (1987); Jones (1999); Miller et al (1996). This first approach suffers from two main drawbacks: On the one hand, many of the slope limiters in wide use today are not linearity-preserving on irregular grids, as shown in Berger et al (2005); on the other hand slope limiters usually impose geometric restrictions on the mesh (e.g., cell alignment, logically structured grids, etc.) The second strategy relaxes the bound-preserving requirement in the reconstruction, and in turn the geometric conditions on the mesh. The approach then proceeds with a mass re-distribution to satisfy (P3), see e.g., Kucharik et al (2003); Margolin and Shashkov (2004); Loubere and Shashkov (2005); Loubere et al (2006). Unfortunately, both bound-preserving reconstruction and mass “repair” tend to obscure the sources of discretization errors and make the analysis of accuracy more complex.

The alternative approach pursued here relies as well on the mass flux form of remap to provide (P1), but achieves (P2) and (P3) without bound-preserving reconstruction or mass post-processing. This is because the remap step is rephrased as a *global* inequality-constrained optimization problem for mass fluxes between neighboring cells. The objective is to minimize the discrepancy between these fluxes and the given *target* mass fluxes, subject to constraints that enforce physically motivated bounds on the primitive variable (density).

This strategy is expected to be more robust, flexible and asymptotically accurate than the other two approaches mentioned for the following reasons. First, optimization-based remap (OBR) finds a global optimal solution from a feasible set defined by the local bounds, i.e. OBR always finds *the best possible, with respect to the target fluxes, remapped quantity that also satisfies these bounds*. Therefore, it

does not rely on local “worst-case” assumptions, which can reduce the accuracy, as both bound-preserving reconstruction and mass redistribution.

Second, OBR can be easily adapted to different problems by choosing the most appropriate target fluxes and discrepancy measures (norms) for these problems.

Third, OBR enforces the local bounds (P3) by a set of linear inequalities, which are completely impervious to the shape of the cells in the mesh. Therefore, in principle, OBR can be applied to arbitrary grids, including grids comprising of polygons or polyhedra.

It is important to mention at this point that Rider and Kothe (1997) and Berger et al (2005) used constrained optimization in lieu of standard limiters to define a bound-preserving reconstruction method on general cells. In that work the least-squares gradient recovery on a cell is constrained by the local minimum and maximum of the data, i.e., the limiting remains based on local “worst-case” assumptions. In contrast, we pose the entire remap problem as a globally constrained minimization problem in which all bounds are considered simultaneously. This possibility was first brought up in Liska et al (2010). Using ideas from flux-corrected transport (FCT, see e.g. Kuzmin et al (2005)) these authors developed a flux-corrected remap (FCR) algorithm. Then, they interpreted FCR as “a process of replacing a global constrained optimization problem by series of local constrained optimization problems by considering the worst case scenario”. Liska et al (2010) did not examine in detail this connection, and left open the question about the preservation of linearity in FCR.

The material that follows is aimed at presenting the key components of the proposed approach in detail, and to ultimately demonstrate that the global inequality-constrained optimization strategy leads to robust, accurate and efficient remappers. For this reason, we use the Euclidean norm to measure the flux discrepancy and define the target fluxes using density reconstruction that is exact for linear functions. While not the only possible choices, the former leads to differentiable objectives and the latter provides the preservation of linearity (P2).

Furthermore, we show that under some fairly weak requirements on mesh proximity OBR satisfies (P2) on arbitrary unstructured grids in one, two and three dimensions, including grids with polygonal or polyhedral cells.

We also clarify the intuitive interpretation of FCR given in Liska et al (2010). We show that the FCR solution coincides with the solution of a modified version of OBR (M-OBR), which has the same objective but a simpler set of box constraints derived from the OBR constraints by using a worst-case scenario. FCR is then viewed as an approximate solution procedure for OBR, which seeks minimizers in a reduced feasible set. Because M-OBR (FCR) has a smaller feasible set, preservation of linearity may be lost and accuracy may suffer for some grid configurations.

Numerical studies confirm these conjectures, showing that for certain types of grids FCR defaults to a first-order accurate scheme, while OBR achieves the theoretically best possible accuracy (second order) for a linearity-preserving scheme. We also present examples of grids in one and two dimensions for which OBR is linearity preserving when FCR is not, and grids for which OBR preserves (P3) when FCR does not. These trends also extend to the case of *iterated* FCR (iFCR), a re-

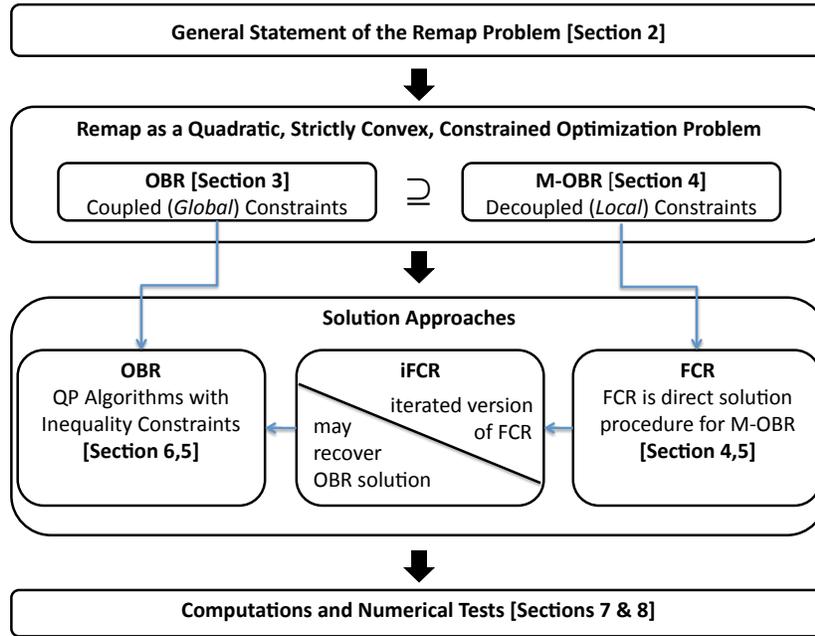


Fig. 1 Outline of the contents of the chapter and the main flow of the presentation.

cursive algorithm derived from standard FCR, in which the low-order remap fluxes are sequentially updated using the most recent FCR monotone iterate. The iFCR algorithm is clearly more expensive than the simple FCR algorithm, but provides a more challenging benchmark for testing the accuracy of OBR.

Our analysis also explains why the FCR fluxes are required to be *convex* combinations of low and high-order fluxes, without appealing to analogies with FCT. We show that the convexity requirement is introduced implicitly when the OBR constraints are approximated by simpler box constraints. This restricts the optimal solution of the global M-OBR problem to convex combinations of low-order and high-order fluxes. Because FCR is a solution procedure for the M-OBR problem, the convexity requirement becomes part of the “formula” for the optimal solution.

The chapter is organized as follows (see also Figure 1 for a roadmap of the presentation of the material). Notation and a formal statement of the remap problem is presented in Section 2, and the new optimization-based formulation of remap is developed in Section 3. There we also establish sufficient conditions for the preservation of linearity in OBR. Connections between OBR, FCR, and iFCR are examined in Section 4. Sections 5 and 6 discuss implementation details of OBR and FCR. Section 7 presents three instructive computational examples, and Section 8 focuses on numerical estimates of convergence rates and assessment of the OBR performance.

## 2 The remap problem

### 2.1 Notation

In what follows  $\Omega \subset \mathbb{R}^d$ ,  $d = 1, 2, 3$ , denotes an open bounded domain with a Lipschitz continuous boundary  $\partial\Omega$ . Bold face lower case Roman symbols denote points in the computational domain with  $\mathbf{x} \in \Omega$  reserved for the independent variable. The symbol  $K_h(\Omega)$  stands for a conforming partition of  $\Omega$  into  $K$  cells  $\kappa_i$ ,  $i = 1, \dots, K$ , with volumes and barycenters given by

$$V(\kappa_i) = \int_{\kappa_i} dV \quad \text{and} \quad \mathbf{b}_i = \frac{\int_{\kappa_i} \mathbf{x} dV}{V(\kappa_i)}, \quad (1)$$

respectively.  $S(K_h)$  is the set of all sides in the mesh  $K_h(\Omega)$ , and  $S(\kappa_i)$  is the subset of  $S(K_h)$  associated with cell  $\kappa_i$ . A side can be oriented in two different ways, which we refer to as positive and negative. We assume that each side  $\sigma_i \in S(K_h)$  is endowed with a unique positive or negative orientation  $\omega_i$ . It is convenient to associate  $\omega_i$  with the numeric values  $+1$  and  $-1$ , for positively and negatively oriented sides, respectively. We recall that conforming partitions of  $\Omega$  consist of cells that cover the domain without gaps or overlaps. The partition  $K_h(\Omega)$  can be uniform or nonuniform, and the cells are not required to have the same shape or to be convex. For instance, in two dimensions  $K_h(\Omega)$  can contain triangles, quadrilaterals and convex and non-convex polygons. This makes our approach applicable to a wide range of grids and methodologies. For example, we can think of a two-dimensional AMR grid (see, e.g., Berger and Colella (1989)) as consisting of quadrilaterals and (degenerate) polygons, while in three dimensions (see, e.g., Bell et al (1994)) such grids will contain cubes and polyhedra.

We assume that  $\Omega$  is endowed with two different grid partitions  $K_h(\Omega)$  and  $\tilde{K}_h(\Omega)$  having the same connectivity. In the context of ALE methods we refer to  $K_h(\Omega)$  as the old or Lagrangian grid and  $\tilde{K}_h(\Omega)$  as the new or rezoned<sup>1</sup> grid. Quantities defined on the new grid will have the tilde accent, e.g.  $\tilde{f}$ , whereas the quantities on  $K_h(\Omega)$  will have no accent. The cells on the new grid are denoted by  $\tilde{\kappa}_i$ , with barycenters  $\tilde{\mathbf{b}}_i$ ,  $i = 1, \dots, K$ . Because  $K_h(\Omega)$  and  $\tilde{K}_h(\Omega)$  have the same connectivity, it is convenient to assume that the new cells are numbered in the same order as the old cells. Therefore, the Lagrangian prototype of the rezoned cell  $\tilde{\kappa}_i$  is the cell  $\kappa_i$ .

The neighborhood  $N(\kappa_i)$  of  $\kappa_i$  comprises of the cell  $\kappa_i$  itself and all its neighbors, i.e. those cells in  $K_h(\Omega)$  that share a vertex (in 1D), vertex or edge (in 2D) and vertex, edge or face (in 3D) with  $\kappa_i$ . The remap problem is stated under the assumption that the rezoned grid satisfies the *locality condition*

$$\tilde{\kappa}_i \subset N(\kappa_i), \quad \text{for all } i = 1, \dots, K, \quad (2)$$

<sup>1</sup> Typically, in a continuous rezoned ALE the rezoned grid is close to the Lagrangian but has better geometric quality.

that is, each rezoned cell  $\tilde{\kappa}_i$  is contained in  $N(\kappa_i)$ , the neighborhood of its Lagrangian prototype. Here the relation  $\tilde{\kappa}_i \subset N(\kappa_i)$  is interpreted geometrically (in contrast to its set-relational definition).<sup>2</sup> In the context of ALE methods, assumption (2) corresponds to using the continuous rezone strategy. Finally,  $\mathcal{I}$  denotes the operator that returns the index of a cell, i.e.  $\mathcal{I}(\kappa_i) = \mathcal{I}(\tilde{\kappa}_i) = i$ . The extension of this operator to sets of cells is natural, e.g.

$$\mathcal{I}(N(\kappa_i)) = \{\mathcal{I}(\kappa_j) \mid \kappa_j \in N(\kappa_i)\}$$

is the set of all indices of the cells in  $N(\kappa_i)$ .

For completeness, we review the specialization of some notation to one-dimensional domains  $\Omega = [a, b]$  where  $a < b$  are real numbers. In this case  $K_h(\Omega)$  is defined by a set of  $K + 1$  points  $a = x_0 < x_1 < \dots < x_{K-1} < x_K = b$ , the Lagrangian cells are the intervals  $\kappa_i = [x_{i-1}, x_i]$  and their volumes are  $V(\kappa_i) = h_i = x_i - x_{i-1}$ . The new grid  $\tilde{K}_h(\Omega)$  comprises of rezoned cells  $\tilde{\kappa}_i = [\tilde{x}_{i-1}, \tilde{x}_i]$  such that  $a = \tilde{x}_0 < \tilde{x}_1 < \dots < \tilde{x}_{K-1} < \tilde{x}_K = b$ . In one dimension, (2) assumes a particularly simple form:

$$\begin{aligned} \tilde{\kappa}_i &\subset (\kappa_{i-1} \cup \kappa_i \cup \kappa_{i+1}) \quad \text{for } i = 2, \dots, K-1, \\ \tilde{\kappa}_1 &\subset (\kappa_1 \cup \kappa_2) \quad \text{and} \quad \tilde{\kappa}_K \subset (\kappa_{K-1} \cup \kappa_K), \end{aligned}$$

or

$$\begin{aligned} \tilde{\kappa}_i &\subset [x_{i-2}, x_{i+1}] \quad \text{for } i = 2, \dots, K-1, \\ \tilde{\kappa}_1 &\subset [a, x_2] \quad \text{and} \quad \tilde{\kappa}_K \subset [x_{K-2}, b]. \end{aligned}$$

An equivalent form of the locality condition is given by

$$x_{i-1} \leq \tilde{x}_i \leq x_{i+1}, \quad i = 1, \dots, K-1. \quad (3)$$

Material in this chapter also requires some notation for Euclidean spaces  $\mathbb{R}^n$ . We use Roman and Greek symbols with an arrow accent, and bold face Roman capitals for vectors and matrices, respectively, e.g.,  $\vec{c} \in \mathbb{R}^n$ ,  $\vec{F} \in \mathbb{R}^n$ ,  $\vec{\lambda} \in \mathbb{R}^n$ , and  $\mathbf{A} \in \mathbb{R}^{n \times m}$ . The superscript  $(\cdot)^T$  indicates vector and matrix transposition. The *Euclidean inner product*,  $\langle \cdot, \cdot \rangle : \mathbb{R}^N \rightarrow \mathbb{R}$ , is  $\langle \vec{a}, \vec{b} \rangle = \vec{a}^T \vec{b}$ , and the *Euclidian norm*  $\|\vec{a}\|_2^2 = \langle \vec{a}, \vec{a} \rangle = \vec{a}^T \vec{a}$ . We use the Euclidean space notation to state algebraic forms of the optimization problems and for various coefficient vectors.

---

<sup>2</sup> In this chapter, we use the set-relational definitions and the corresponding geometric interpretations of  $\subset$ ,  $\subseteq$ ,  $\cup$ ,  $\cap$ ,  $\setminus$  and  $\in$  interchangeably. Their meaning will be clear from the context. In particular, relations between entities defined on  $\tilde{K}_h(\Omega)$  and those defined on  $K_h(\Omega)$  only make sense when interpreted geometrically relative to the common domain  $\Omega$ .

## 2.2 Statement of the remap problem

We recall the formal statement of mass-density remap following Margolin and Shashkov (2003); Liska et al (2010). We assume that there is a positive function  $\rho(\mathbf{x}) > 0$ , referred to as *density*, that is defined on  $\Omega$  and whose values on the boundary  $\partial\Omega$  are known. The only information given about  $\rho(\mathbf{x})$  in the interior of  $\Omega$  is its mean value on the old cells:

$$\rho_i = \frac{\int_{\kappa_i} \rho(\mathbf{x}) dV}{V(\kappa_i)}.$$

Equivalently, we can write

$$\rho_i = \frac{m_i}{V(\kappa_i)} \quad \text{or} \quad m_i = \rho_i V(\kappa_i) \quad (4)$$

where

$$m_i = \int_{\kappa_i} \rho(\mathbf{x}) dV$$

is the (old) cell mass. Here we have implicitly assumed that the initial distribution of  $\rho(\mathbf{x})$  is known exactly, and that the previous integral represents the exact mass associated with cell  $i$ . The total mass is

$$M = \int_{\Omega} \rho(\mathbf{x}) dV = \sum_{i=1}^K \int_{\kappa_i} \rho(\mathbf{x}) dV = \sum_{i=1}^K m_i = \sum_{i=1}^K \rho_i V(\kappa_i).$$

For further reference we note that the mean density on every Lagrangian cell  $\kappa_i$  trivially satisfies the bounds

$$\rho_i^{\min} \leq \rho_i \leq \rho_i^{\max}, \quad (5)$$

where

$$\rho_i^{\min} = \begin{cases} \min_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\} & \text{if } \kappa_i \cap \partial\Omega = \emptyset \\ \min \left\{ \min_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\}, \min_{\mathbf{x} \in N(\kappa_i) \cap \partial\Omega} \rho(\mathbf{x}) \right\} & \text{if } \kappa_i \cap \partial\Omega \neq \emptyset \end{cases} \quad (6)$$

and

$$\rho_i^{\max} = \begin{cases} \max_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\} & \text{if } \kappa_i \cap \partial\Omega = \emptyset \\ \max \left\{ \max_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\}, \max_{\mathbf{x} \in N(\kappa_i) \cap \partial\Omega} \rho(\mathbf{x}) \right\} & \text{if } \kappa_i \cap \partial\Omega \neq \emptyset. \end{cases} \quad (7)$$

In words, for cells that do not intersect the boundary  $\partial\Omega$ , the values of  $\rho_i^{\min}$  and  $\rho_i^{\max}$  give the smallest and the largest mean densities in the neighborhood of  $\kappa_i$ , respectively. For cells adjacent to the boundary,  $\rho_i^{\min}$  is the smaller of the smallest mean cell density in the cell neighborhood and the smallest density on the boundary segment  $N(\kappa_i) \cap \partial\Omega$ ;  $\rho_i^{\max}$  is defined analogously. Bounds for the cell masses follow from (4) and (5):

$$\rho_i^{\min} V(\kappa_i) = m_i^{\min} \leq m_i \leq m_i^{\max} = \rho_i^{\max} V(\kappa_i) \quad \forall \kappa_i \in K_h(\Omega). \quad (8)$$

A formal statement of the mass-density remap problem is as follows.

**Definition 2.1 (Remapping of mass-density)** *Given mean density values  $\rho_i$  on the old grid cells  $\kappa_i$ , find accurate approximations  $\tilde{m}_i$  for the masses of the new cells  $\tilde{\kappa}_i$ ,*

$$\tilde{m}_i \approx \tilde{m}_i^{\text{ex}} = \int_{\tilde{\kappa}_i} \rho(\mathbf{x}) dV; \quad i = 1, \dots, K, \quad (9)$$

such that the following conditions hold:

R1. *The total mass is conserved:*

$$\sum_{i=1}^K \tilde{m}_i = \sum_{i=1}^K m_i = M.$$

R2. *If the exact density  $\rho(\mathbf{x})$  is a linear function on all of  $\Omega$ , then the remapped masses are exact:*

$$\tilde{m}_i = \tilde{m}_i^{\text{ex}} = \int_{\tilde{\kappa}_i} \rho(\mathbf{x}) dV; \quad i = 1, \dots, K. \quad (10)$$

R3. *Given approximate masses  $\tilde{m}_i$  on the new cells, define  $\tilde{\rho}_i = \tilde{m}_i/V(\tilde{\kappa}_i)$ . Let  $\rho_i^{\min}$  and  $\rho_i^{\max}$  be the quantities defined in (6)–(7). Then the bounds*

$$\rho_i^{\min} \leq \tilde{\rho}_i \leq \rho_i^{\max}$$

and

$$\rho_i^{\min} V(\tilde{\kappa}_i) = \tilde{m}_i^{\min} \leq \tilde{m}_i \leq \tilde{m}_i^{\max} = \rho_i^{\max} V(\tilde{\kappa}_i) \quad (11)$$

hold on every new cell  $\tilde{\kappa}_i$ .  $\square$

Requirements (R1–R3) in Definition 2.1 are derived from the desired remap properties (P1–P3). (R1) and (R2) are formal statements of (P1) and (P2), whereas (R3) follows from the bounds in (5) and (8), and the locality assumption (2). Therefore, the last requirement is specific to a continuous rezone strategy and may have to be modified for other settings. Such a modification is beyond the scope of this chapter.

### 3 A constrained optimization formulation of the remap problem

In this section we develop an inequality-constrained optimization formulation of remap that satisfies requirements (R1–R3). The conservation of total mass (R1) is the simplest one. For any two grids that satisfy the locality assumption (2), the new cells have the following representation (cf. Margolin and Shashkov (2003, Eq.(3.9))):

$$\tilde{\kappa}_i = \left( \kappa_i \cup \bigcup_{j \in \mathcal{J}(N(\kappa_i))} \tilde{\kappa}_i \cap \kappa_j \right) \setminus \left( \bigcup_{j \in \mathcal{J}(N(\kappa_i))} \kappa_i \cap \tilde{\kappa}_j \right), \quad (12)$$

Using (12) we can express the exact masses of the new cells in *flux form*

$$\tilde{m}_i^{\text{ex}} = m_i + \sum_{j \in \mathcal{J}(N(\kappa_i))} F_{ij}^{\text{ex}}, \quad (13)$$

where the (exact) fluxes are (cf. Margolin and Shashkov (2003, Eq.(3.12)))

$$F_{ij}^{\text{ex}} = \int_{\tilde{\kappa}_i \cap \kappa_j} \rho(\mathbf{x}) dV - \int_{\kappa_i \cap \tilde{\kappa}_j} \rho(\mathbf{x}) dV. \quad (14)$$

Formula (14) implies that the exact mass fluxes are antisymmetric:  $F_{ij}^{\text{ex}} = -F_{ji}^{\text{ex}}$ . Assume that  $F_{ij}$  are approximate mass fluxes that are also antisymmetric

$$F_{ij} = -F_{ji}. \quad (15)$$

Using these fluxes in (16) yields a formula for the approximation of the new cell masses

$$\tilde{m}_i = m_i + \sum_{j \in \mathcal{J}(N(\kappa_i))} F_{ij}, \quad (16)$$

which preserves the total mass, i.e. satisfies (R1) in Definition 2.1. To satisfy (R2) we introduce the notion of *high-order target* mass fluxes

$$F_{ij}^H = \int_{\tilde{\kappa}_i \cap \kappa_j} \rho_j^H(\mathbf{x}) dV - \int_{\kappa_i \cap \tilde{\kappa}_j} \rho_i^H(\mathbf{x}) dV, \quad (17)$$

where  $\rho_i^H(\mathbf{x})$  is a density reconstruction on  $\kappa_i$  that is exact for linear functions. If  $\rho(\mathbf{x})$  is linear, then  $F_{ij}^H = F_{ij}^{\text{ex}}$ , i.e., the target fluxes coincide<sup>3</sup> with the exact fluxes for linear functions. In this case, using (16) with the target fluxes gives the exact new cell masses, i.e., (R2) holds. However, if  $\rho(\mathbf{x})$  is not linear, using  $F_{ij}^H$  in (16) will likely lead to violation of (R3), especially when  $\rho(\mathbf{x})$  is not smooth. We then constrain the set of approximate fluxes  $F_{ij}$  introduced in (15)–(16) by the global system of linear inequalities

<sup>3</sup> In practice, this also means that the integrals in (17) should be approximated by quadratures that are exact for linear functions.

$$\tilde{m}_i^{\min} \leq m_i + \sum_{j \in \mathcal{S}(N(\kappa_i))} F_{ij} \leq \tilde{m}_i^{\max}; \quad i = 1, \dots, K, \quad (18)$$

obtained by substituting the approximate mass in (11) with the flux form formula (16). By construction, any  $F_{ij}$  that solves (18) produces new cell masses that satisfy (R3). To summarize,

- using the flux form (16) guarantees the conservation of total mass (R1);
- using (16) with the target fluxes  $F_{ij}^H$  ensures preservation of linearity (R2);
- using (16) with fluxes  $F_{ij}$  which solve (18) secures the preservation of local bounds (R3).

We use optimization to reconcile the last two properties. Let us regard the fluxes  $F_{ij}$  as the unknowns, the inequalities (18) as the constraints, and the minimization of the Euclidean distance<sup>4</sup> between the target and the unknown fluxes as the objective. The resulting constrained optimization problem reads

$$\left\{ \begin{array}{l} \min_{F_{ij}} \sum_{i=1}^K \sum_{j \in \mathcal{S}(N(\kappa_i))} (F_{ij} - F_{ij}^H)^2 \quad \text{subject to} \\ F_{ij} = -F_{ji} \quad i = 1, \dots, K, \quad j \in \mathcal{S}(N(\kappa_i)) \\ \tilde{m}_i^{\min} \leq m_i + \sum_{j \in \mathcal{S}(N(\kappa_i))} F_{ij} \leq \tilde{m}_i^{\max} \quad i = 1, \dots, K. \end{array} \right. \quad (19)$$

Explicit enforcement of the antisymmetry constraint by using only the fluxes  $F_{pq}$  for which  $p < q$  simplifies the optimization problem:

$$\left\{ \begin{array}{l} \min_{F_{ij}} \sum_{i=1}^K \sum_{\substack{j \in \mathcal{S}(N(\kappa_i)) \\ i < j}} (F_{ij} - F_{ij}^H)^2 \quad \text{subject to} \\ \tilde{m}_i^{\min} - m_i \leq \sum_{\substack{j \in \mathcal{S}(N(\kappa_i)) \\ i < j}} F_{ij} - \sum_{\substack{j \in \mathcal{S}(N(\kappa_i)) \\ i > j}} F_{ji} \leq \tilde{m}_i^{\max} - m_i \quad i = 1, \dots, K, \end{array} \right. \quad (20)$$

where we have also moved  $m_i$  to the left and right of the chain of inequalities. Any feasible point of (20) satisfies (R1) and (R3) by construction.

We proceed to show that (20) has a non-empty feasible set, i.e., there is always a non-trivial optimal solution, and that the optimal solution preserves linear densities.

**Theorem 1.** *Assume that  $K_h(\Omega)$  and  $\tilde{K}_h(\Omega)$  are such that the locality condition (2) holds. For any given set of masses  $m_i$  and associated densities  $\rho_i = m_i/V(\kappa_i)$  on  $K_h(\Omega)$  there exist antisymmetric fluxes  $\{F_{ij}\}$  which satisfy the inequality constraints in (20), resp (19).*

*Proof.* We need to show that there are antisymmetric fluxes  $F_{ij}$  such that

<sup>4</sup> The Euclidean distance is used for simplicity. The objective can be defined using any valid distance function (or, equivalently, norm).

$$\rho_i^{\min} V(\tilde{\kappa}_i) \leq \rho_i V(\kappa_i) + \sum_{\kappa_j \in N_i} F_{ij} \leq \rho_i^{\max} V(\tilde{\kappa}_i)$$

Fix a cell index  $1 \leq i \leq K$ , and choose  $\hat{\rho}_j$ , for  $\kappa_j \in N_j$  according to

$$\rho_i^{\min} \leq \hat{\rho}_j \leq \rho_i^{\max} \quad \text{for } j \neq i \quad \text{and} \quad \hat{\rho}_i = \rho_i. \quad (21)$$

The representation formula (12) motivates the following definition:

$$F_{ij} = \hat{\rho}_j V(\tilde{\kappa}_i \cap \kappa_j) - \hat{\rho}_i V(\kappa_i \cap \tilde{\kappa}_j). \quad (22)$$

Clearly,  $F_{ij} = -F_{ji}$ . Using the fluxes (22)

$$\begin{aligned} \rho_i V(\kappa_i) + \sum_{\kappa_j \in N_i} F_{ij} &= \rho_i \left[ V(\kappa_i) - \sum_{j \neq i} V(\kappa_i \cap \tilde{\kappa}_j) \right] + \sum_{j \neq i} \hat{\rho}_j V(\tilde{\kappa}_i \cap \kappa_j) \\ &= \rho_i V(\tilde{\kappa}_i \cap \kappa_i) + \sum_{j \neq i} \hat{\rho}_j V(\kappa_i \cap \tilde{\kappa}_j) = \sum_{\kappa_j \in N_i} \hat{\rho}_j V(\tilde{\kappa}_i \cap \kappa_j). \end{aligned}$$

From  $\tilde{\kappa}_i = \cup_{\kappa_j \in N_i} (\tilde{\kappa}_i \cap \kappa_j)$  and the bounds in (21) it follows that

$$\begin{aligned} \sum_{\kappa_j \in N_i} \hat{\rho}_j V(\tilde{\kappa}_i \cap \kappa_j) &\leq \rho_i^{\max} \sum_{\kappa_j \in N_i} V(\tilde{\kappa}_i \cap \kappa_j) = \rho_i^{\max} V(\tilde{\kappa}_i); \\ \sum_{\kappa_j \in N_i} \hat{\rho}_j V(\tilde{\kappa}_i \cap \kappa_j) &\geq \rho_i^{\min} \sum_{\kappa_j \in N_i} V(\tilde{\kappa}_i \cap \kappa_j) = \rho_i^{\min} V(\tilde{\kappa}_i), \end{aligned}$$

which proves the theorem.

Preservation of linearity (R2) requires the target fluxes  $F_{ij}^H$  to be in the feasible set of (20) whenever  $\rho(\mathbf{x})$  is linear, i.e.,

$$\tilde{m}_i^{\min} - m_i \leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} F_{ij}^H - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} F_{ji}^H \leq \tilde{m}_i^{\max} - m_i \quad i = 1, \dots, K. \quad (23)$$

The proof of this fact requires a simple technical result.

**Lemma 3.1** *Let  $n > 0$  be an integer and let  $\vec{c} \in \mathbb{R}^n$  be an arbitrary fixed vector. For any closed and bounded set of points  $P \subset \mathbb{R}^n$*

$$\min_{\mathbf{x} \in P} (\vec{c}^\top \mathbf{x}) = \min_{\mathbf{x} \in \mathcal{H}(P)} (\vec{c}^\top \mathbf{x}) \quad \text{and} \quad \max_{\mathbf{x} \in P} (\vec{c}^\top \mathbf{x}) = \max_{\mathbf{x} \in \mathcal{H}(P)} (\vec{c}^\top \mathbf{x}), \quad (24)$$

where  $\mathcal{H}(P)$  is the convex hull of  $P$ .

*Proof.* The real-valued function  $\vec{c}^\top \mathbf{x}$  is continuous on  $\mathbb{R}^n$ . The set  $P$  is closed and bounded, which implies that  $\vec{c}^\top \mathbf{x}$  attains its minimum and maximum over  $P$ . Since the convex hull of a closed and bounded set is closed and bounded, see Rockafellar (1970, Theorem 17.2), the same is true for  $\mathcal{H}(P)$ .<sup>5</sup>

<sup>5</sup> This guarantees that taking min and max in (24) is well-defined. Otherwise, the correct statement of this result should involve inf and sup.

The function  $\bar{c}^\top \mathbf{x}$  is linear, hence both convex and concave. The claim of the lemma follows from a standard result on the supremum of convex (infimum of concave) functions, see e.g. Rockafellar (1970, Theorem 32.2).

The following theorem provides sufficient conditions on mesh movement for (23) to hold.

**Theorem 2.** *Assume the locality condition (2) and suppose that the exact density  $\rho(\mathbf{x})$  is linear in all of  $\Omega$ . Let  $B_i$  denote the set of barycenters of the Lagrangian cells in  $N(\kappa_i)$ ,*

$$B_i = \{\mathbf{b}_j \mid j \in \mathcal{J}(N(\kappa_i))\},$$

*and let  $\tilde{\mathbf{b}}_i$  be the barycenter of the rezoned cell  $\tilde{\kappa}_i$ . Sufficient conditions for the target fluxes to be in the feasible set of (20), that is for (23) to hold, are*

$$\tilde{\mathbf{b}}_i \in \mathcal{H}(B_i) \quad \text{if } \kappa_i \cap \partial\Omega = \emptyset, \quad (25)$$

$$\tilde{\mathbf{b}}_i \in \mathcal{H}(B_i \cup (N(\kappa_i) \cap \partial\Omega)) \quad \text{if } \kappa_i \cap \partial\Omega \neq \emptyset, \quad (26)$$

where  $\mathcal{H}(\cdot)$  denotes the convex hull.

*Proof.* Because  $\rho(\mathbf{x})$  is linear and the density reconstruction is exact for linear functions it follows that the remapped mass equals the exact mass on every rezoned cell  $\tilde{\kappa}_i$ :

$$\tilde{m}_i = m_i + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} F_{ij}^H - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} F_{ji}^H = m_i + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} F_{ij}^{\text{ex}} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} F_{ji}^{\text{ex}} = \tilde{m}_i^{\text{ex}}.$$

Therefore, proving that (23) holds reduces to showing that

$$\tilde{m}_i^{\min} \leq \tilde{m}_i^{\text{ex}} \leq \tilde{m}_i^{\max} \quad \text{for all } i = 1, \dots, K. \quad (27)$$

Recalling  $\rho(\mathbf{x}) = c_0 + \bar{c}^\top \mathbf{x}$  and using the barycenter formula (1) yields

$$\begin{aligned} \tilde{m}_i^{\text{ex}} &= \int_{\tilde{\kappa}_i} (c_0 + \bar{c}^\top \mathbf{x}) dV = c_0 V(\tilde{\kappa}_i) + \bar{c}^\top \left[ \int_{\tilde{\kappa}_i} \mathbf{x} dV \right] \\ &= c_0 V(\tilde{\kappa}_i) + \bar{c}^\top \left[ \frac{\int_{\tilde{\kappa}_i} \mathbf{x} dV}{V(\tilde{\kappa}_i)} \right] V(\tilde{\kappa}_i) = (c_0 + \bar{c}^\top \tilde{\mathbf{b}}_i) V(\tilde{\kappa}_i). \end{aligned}$$

We consider two cases,  $\kappa_i \cap \partial\Omega = \emptyset$  and  $\kappa_i \cap \partial\Omega \neq \emptyset$ .

Case 1: *Suppose  $\kappa_i \cap \partial\Omega = \emptyset$*

Using

$$\rho_i^{\min} = \min_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\} \quad \text{and} \quad \rho_i^{\max} = \max_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\},$$

the barycenter formula yields

$$\tilde{m}_i^{\min} = \min_{j \in \mathcal{J}(N(\kappa_i))} \left[ \frac{\int_{\kappa_j} (c_0 + \bar{c}^T \mathbf{x}) dV}{V(\kappa_j)} \right] V(\tilde{\kappa}_i) = \min_{\mathbf{b}_j \in B_i} (c_0 + \bar{c}^T \mathbf{b}_j) V(\tilde{\kappa}_i)$$

for the lower bound and

$$\tilde{m}_i^{\max} = \max_{j \in \mathcal{J}(N(\kappa_i))} \left[ \frac{\int_{\kappa_j} (c_0 + \bar{c}^T \mathbf{x}) dV}{V(\kappa_j)} \right] V(\tilde{\kappa}_i) = \max_{\mathbf{b}_j \in B_i} (c_0 + \bar{c}^T \mathbf{b}_j) V(\tilde{\kappa}_i)$$

for the upper bound in (27). From Lemma 3.1 it follows that

$$\min_{\mathbf{b}_j \in B_i} (c_0 + \bar{c}^T \mathbf{b}_j) = \min_{\mathbf{x} \in \mathcal{H}(B_i)} (c_0 + \bar{c}^T \mathbf{x}) \quad (28)$$

and

$$\max_{\mathbf{b}_j \in B_i} (c_0 + \bar{c}^T \mathbf{b}_j) = \max_{\mathbf{x} \in \mathcal{H}(B_i)} (c_0 + \bar{c}^T \mathbf{x}). \quad (29)$$

Consequently, whenever  $\kappa_i \cap \partial\Omega = \emptyset$ , (27) is equivalent to

$$\min_{\mathbf{x} \in \mathcal{H}(B_i)} (c_0 + \bar{c}^T \mathbf{x}) \leq (c_0 + \bar{c}^T \tilde{\mathbf{b}}_i) \leq \max_{\mathbf{x} \in \mathcal{H}(B_i)} (c_0 + \bar{c}^T \mathbf{x}). \quad (30)$$

A sufficient condition for (30) is given by (25).

Case 2: *Suppose  $\kappa_i \cap \partial\Omega \neq \emptyset$*

We have

$$\rho_i^{\min} = \min \left\{ \min_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\}, \min_{\mathbf{x} \in N(\kappa_i) \cap \partial\Omega} (c_0 + \bar{c}^T \mathbf{x}) \right\}$$

and

$$\rho_i^{\max} = \max \left\{ \max_{j \in \mathcal{J}(N(\kappa_i))} \{\rho_j\}, \max_{\mathbf{x} \in N(\kappa_i) \cap \partial\Omega} (c_0 + \bar{c}^T \mathbf{x}) \right\}.$$

Using again the barycenter formula, we obtain

$$\rho_i^{\min} = \min \left\{ \min_{\mathbf{x} \in B_i} (c_0 + \bar{c}^T \mathbf{x}), \min_{\mathbf{x} \in N(\kappa_i) \cap \partial\Omega} (c_0 + \bar{c}^T \mathbf{x}) \right\}$$

and

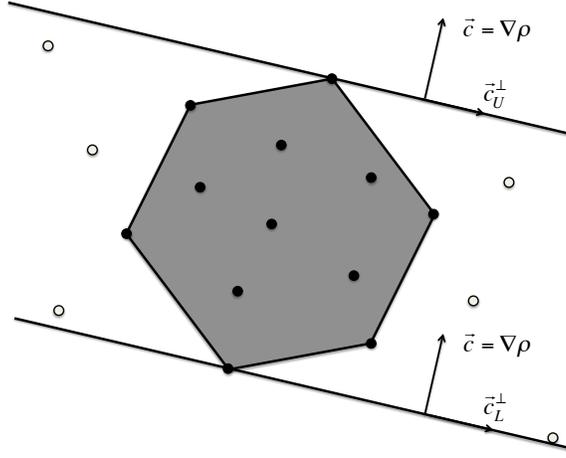
$$\rho_i^{\max} = \max \left\{ \max_{\mathbf{x} \in B_i} (c_0 + \bar{c}^T \mathbf{x}), \max_{\mathbf{x} \in N(\kappa_i) \cap \partial\Omega} (c_0 + \bar{c}^T \mathbf{x}) \right\}.$$

In other words,

$$\tilde{m}_i^{\min} = \min_{\mathbf{x} \in B_i \cup (N(\kappa_i) \cap \partial\Omega)} (c_0 + \bar{c}^T \mathbf{x}) V(\tilde{\kappa}_i)$$

and

$$\tilde{m}_i^{\max} = \max_{\mathbf{x} \in B_i \cup (N(\kappa_i) \cap \partial\Omega)} (c_0 + \bar{c}^T \mathbf{x}) V(\tilde{\kappa}_i).$$



**Fig. 2** The level sets of  $\rho(\mathbf{x}) = c_0 + \vec{c}^T \mathbf{x}$  are perpendicular to  $\nabla \rho(\mathbf{x}) = \vec{c}$  and the extrema of  $\rho(\mathbf{x})$  are achieved along the parallel lines  $\vec{c}_L^\perp$  and  $\vec{c}_U^\perp$  shown in the plot. Therefore, inequality (30) holds for all points between the two lines, while (25) requires  $\tilde{\mathbf{b}}_i$  to remain in the convex hull  $\mathcal{H}(B_i)$  (the gray hexagon).

Treating  $B_i \cup (N(\kappa_i) \cap \partial \Omega)$  as a set of points in  $\mathbb{R}^n$ , another application of Lemma 3.1 gives

$$\tilde{m}_i^{\min} = \min_{\mathbf{x} \in \mathcal{H}(B_i \cup (N(\kappa_i) \cap \partial \Omega))} (c_0 + \vec{c}^T \mathbf{x}) V(\tilde{\kappa}_i)$$

and

$$\tilde{m}_i^{\max} = \max_{\mathbf{x} \in \mathcal{H}(B_i \cup (N(\kappa_i) \cap \partial \Omega))} (c_0 + \vec{c}^T \mathbf{x}) V(\tilde{\kappa}_i).$$

Therefore, whenever  $\kappa_i \cap \partial \Omega \neq \emptyset$ , a sufficient condition for (27) is given by (26). This concludes the proof.

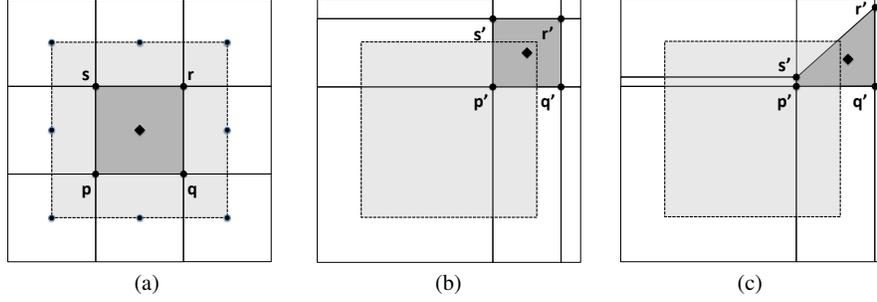
**Remark 3.1** The sufficient condition (26) can be replaced by more restrictive conditions of the type

$$\tilde{\mathbf{b}}_i \in \mathcal{H}(B_i \cup S_i) \quad \text{if } \kappa_i \cap \partial \Omega \neq \emptyset,$$

where  $S_i \subseteq (N(\kappa_i) \cap \partial \Omega)$ , i.e.  $S_i$  is any (for example, finite) set of points taken from the boundary segment  $N(\kappa_i) \cap \partial \Omega$ .

**Remark 3.2** The sufficient conditions (25) and (26) are not in any way dependent on the cell shape. As a result, the statement of Theorem 2 applies to general grids, including grids which contain, e.g., non-convex polytopes. This allows to use OBR for a wider range of mesh partitions of  $\Omega$ .

Simple examples showing mesh motions that comply with or violate condition (25) are shown in Figure 3. It is worth pointing out that a similar but more restrictive condition  $\tilde{\kappa}_i \subset \mathcal{H}(B_i)$  is necessary and sufficient for linear functions to



**Fig. 3** Examples of mesh motions which satisfy and violate, respectively, the sufficient condition for the preservation of linearity in Theorem 2. (a) the neighborhood  $N(\kappa_i)$  consisting of 9 square cells, the Lagrangian prototype of  $\tilde{\kappa}_i$  with vertices  $(\mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{s})$ , its barycenter (the diamond), the set  $B_i$  (the solid dots), and its convex hull  $\mathcal{H}(B_i)$  (the dotted square); (b) an *admissible* rezoned grid for which  $\tilde{\mathbf{b}}_i \in \mathcal{H}(B_i)$ ; (c) an *inadmissible* rezoned grid for which  $\tilde{\mathbf{b}}_i \notin \mathcal{H}(B_i)$ . In (b) and (c)  $\tilde{\kappa}_i$  is the cell with vertices  $(\mathbf{p}', \mathbf{q}', \mathbf{r}', \mathbf{s}')$ . All cells in (a)–(c) satisfy the locality condition (2). Note that the rezoned cell in (b) violates  $\tilde{\kappa}_i \subset \mathcal{H}(B_i)$  which is necessary and sufficient for Van Leer slope limiting to recover linear functions as shown by Swartz (1999), but which is not required for the OBR formulation.

be preserved under Van Leer slope limiting; see Swartz (1999). The center pane in Figure 3 provides an example for which  $\tilde{\kappa}_i \not\subset \mathcal{H}(B_i)$  but  $\tilde{\mathbf{b}}_i \in \mathcal{H}(B_i)$ , i.e. Van Leer slope limiting does not preserve linear functions whereas OBR does.

#### 4 OBR, modified-OBR (M-OBR), and connection with flux-corrected remap (FCR)

In this section we establish connections between the global OBR problem (20) and the FCR algorithm by Liska et al (2010). The FCR algorithm is formulated by defining the mass fluxes in (16) to be convex combinations of so-called low-order and high-order fluxes; the low-order fluxes are assumed to satisfy the local bounds. We will have more to say about this assumption later. The first step is to rewrite (20) in terms of the same low-order and high-order fluxes as in FCR. The reformulation of OBR amounts to a change of variables that leaves the solution of (20) intact but places the OBR problem in a form that can be compared with FCR. The second step approximates the constraints in OBR by a set of inequalities which are sufficient for the original constraints to hold but have a simpler structure. This step gives rise to a modified version of OBR, termed M-OBR, in which the original objective is minimized over a subset of the original OBR feasible set. The final step entails showing that the optimal solution of M-OBR coincides with the FCR solution.

### 4.1 Reformulation of the optimization-based remap

The low-order fluxes in FCR are defined by the formula

$$F_{ij}^L = \int_{\tilde{\kappa}_i \cap \kappa_j} \rho_j^L(\mathbf{x}) dV - \int_{\kappa_i \cap \tilde{\kappa}_j} \rho_i^L(\mathbf{x}) dV, \quad (31)$$

using a piecewise constant reconstruction  $\rho_i^L(\mathbf{x})$  of the old mesh values  $\rho_i$ , i.e.

$$\rho_i^L(\mathbf{x}) = \rho_i \quad \forall \mathbf{x} \in \kappa_i, \quad i = 1, \dots, K.$$

Using these fluxes in formula (16) gives a low-order approximation of the mass in the rezoned cell  $\tilde{\kappa}_i$ .

$$\tilde{m}_i^L = m_i + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} F_{ij}^L - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} F_{ji}^L. \quad (32)$$

Because  $F_{ij}^L$  are computed using exact cell intersections, the new masses satisfy the local bounds, see Margolin and Shashkov (2003, Section 3)

$$\tilde{m}_i^{\min} \leq \tilde{m}_i^L \leq \tilde{m}_i^{\max}. \quad (33)$$

The high-order fluxes in the FCR are defined by the same formula (17) as our target fluxes. Therefore we change the variables in (20) according to

$$F_{ij} = (1 - a_{ij})F_{ij}^L + a_{ij}F_{ij}^H = F_{ij}^L + a_{ij}dF_{ij}, \quad (34)$$

where  $dF_{ij} = F_{ij}^H - F_{ij}^L$ . The coefficients  $a_{ij}$  are the new variables for the optimization problem. It easy to see that antisymmetry of the fluxes implies symmetry of the new variables:  $a_{ij} = a_{ji}$ . However, the change of variables does not introduce any additional constraints on  $a_{ij}$ . As before, we enforce the symmetry constraint by using only coefficients  $a_{pq}$  for which  $p < q$ .

Under the change of variables (34) the terms in the objective functional assume the form

$$F_{ij} - F_{ij}^H = F_{ij}^L + a_{ij}dF_{ij} - F_{ij}^H = (a_{ij} - 1)dF_{ij}.$$

Using (32) and (34) we rewrite the constraints as follows:

$$\begin{aligned} \tilde{m}_i &= m_i + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} F_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} F_{ji} \\ &= m_i + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} (F_{ij}^L + a_{ij}dF_{ij}) - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} (F_{ji}^L + a_{ji}dF_{ji}) \\ &= \left( m_i + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} F_{ij}^L - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} F_{ji}^L \right) + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} a_{ij}dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} a_{ji}dF_{ji} \\ &= \tilde{m}_i^L + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} a_{ij}dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} a_{ji}dF_{ji}. \end{aligned}$$

From (33) it follows that

$$\tilde{Q}_i^{\min} := \tilde{m}_i^{\min} - \tilde{m}_i^L \leq 0 \quad \text{and} \quad \tilde{Q}_i^{\max} := \tilde{m}_i^{\max} - \tilde{m}_i^L \geq 0. \quad (35)$$

We write the transformed constraints using these quantities as

$$\tilde{Q}_i^{\min} \leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} a_{ji} dF_{ji} \leq \tilde{Q}_i^{\max} \quad i = 1, \dots, K. \quad (36)$$

In summary, after changing variables according to (34), the OBR problem (20) assumes the form

$$\left\{ \begin{array}{l} \min_{a_{ij}} \sum_{i=1}^K \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} (1 - a_{ij})^2 (dF_{ij})^2 \quad \text{subject to} \\ \tilde{Q}_i^{\min} \leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} a_{ji} dF_{ji} \leq \tilde{Q}_i^{\max} \quad i = 1, \dots, K. \end{array} \right. \quad (37)$$

Problems (20) and (37) are completely equivalent. For example, the global minimizer  $a_{ij} = 1$  of (37), sans constraints, corresponds to  $F_{ij} = F_{ij}^H$ , which is the global minimizer of (20), sans constraints. Note also that the choice  $a_{ij} = 0$  satisfies the constraints, due to (35). The sufficient conditions in Theorem 2 guarantee that  $a_{ij} = 1$  are in the feasible set of (37) when the exact density  $\rho(\mathbf{x})$  is a linear function in all of  $\Omega$ .

## 4.2 The M-OBR formulation

In this section we modify (37) to another inequality-constrained optimization problem, termed M-OBR, in which the same objective is minimized subject to a set of simple box constraints. The box constraints are sufficient for the original inequality constraints in (37) to hold and are derived by following the same reasoning as in Liska et al (2010). To this end, we define the quantities

$$P_i^- = \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} dF_{ji} \leq 0; \quad P_i^+ = \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \geq 0} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \leq 0} dF_{ji} \geq 0; \quad (38)$$

$$D_i^- = \begin{cases} \frac{\tilde{Q}_i^{\min}}{P_i^-} & \text{if } P_i^- < 0 \\ 0 & \text{if } P_i^- = 0 \end{cases} \quad \text{and} \quad D_i^+ = \begin{cases} \frac{\tilde{Q}_i^{\max}}{P_i^+} & \text{if } P_i^+ > 0 \\ 0 & \text{if } P_i^+ = 0 \end{cases}. \quad (39)$$

Using these quantities we reduce the constraints in (37) to a set of box constraints in three steps.

In the first step we replace the upper and lower bounds in the constraints of (37) by  $D_i^- P_i^-$  and  $D_i^+ P_i^+$ , respectively:

$$D_i^- P_i^- \leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}} a_{ji} dF_{ji} \leq D_i^+ P_i^+ \quad i = 1, \dots, K. \quad (40)$$

In the second step we split (40) into two parts, according to the signs of the flux differentials:

$$\begin{aligned} (a) \quad D_i^- P_i^- &\leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} a_{ji} dF_{ji} \leq 0 \\ (b) \quad 0 &\leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \geq 0} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \leq 0} a_{ji} dF_{ji} \leq D_i^+ P_i^+ \end{aligned} \quad i = 1, \dots, K. \quad (41)$$

Finally, using definition (38), we reduce (41) to a set of box constraints by applying the upper and the lower bounds componentwise:

$$\begin{aligned} (a) \quad &\begin{cases} D_i^- dF_{ij} \leq a_{ij} dF_{ij} \leq 0 & \text{for } i < j, dF_{ij} \leq 0 \\ D_i^- dF_{ji} \geq a_{ji} dF_{ji} \geq 0 & \text{for } i > j, dF_{ji} \geq 0 \end{cases} \quad i = 1, \dots, K \\ (b) \quad &\begin{cases} 0 \leq a_{ij} dF_{ij} \leq D_i^+ dF_{ij} & \text{for } i < j, dF_{ij} \geq 0 \\ 0 \geq a_{ji} dF_{ji} \geq D_i^+ dF_{ji} & \text{for } i > j, dF_{ji} \leq 0 \end{cases} \quad j \in \mathcal{J}(N(\kappa_i)) \end{aligned} \quad (42)$$

Using the box constraints (42) in lieu of the original set of inequalities in (37) yields the modified OBR problem (M-OBR)

$$\left\{ \begin{array}{l} \min_{a_{ij}} \sum_{i=1}^K \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}} (1 - a_{ij})^2 (dF_{ij})^2 \quad \text{subject to} \\ (a) \quad \begin{cases} D_i^- dF_{ij} \leq a_{ij} dF_{ij} \leq 0 & \text{for } i < j, dF_{ij} \leq 0 \\ D_i^- dF_{ji} \geq a_{ji} dF_{ji} \geq 0 & \text{for } i > j, dF_{ji} \geq 0 \end{cases} \quad i = 1, \dots, K \\ (b) \quad \begin{cases} 0 \leq a_{ij} dF_{ij} \leq D_i^+ dF_{ij} & \text{for } i < j, dF_{ij} \geq 0 \\ 0 \geq a_{ji} dF_{ji} \geq D_i^+ dF_{ji} & \text{for } i > j, dF_{ji} \leq 0 \end{cases} \quad j \in \mathcal{J}(N(\kappa_i)) \end{array} \right. \quad (43)$$

We are now ready to study the connections of the global M-OBR formulation (43) with the OBR problem (37). The first result shows that (43) always has a solution.

**Proposition 4.1** *The feasible set of the modified OBR problem (43) is non-empty.*

*Proof.* The inequalities in (43) are always satisfied for  $a_{ij} = 0$  because  $D_i^- \geq 0$  and  $D_i^+ \geq 0$  for all  $i = 1, \dots, K$ . Therefore, the feasible set of (43) always contains at least one point.

We note that  $a_{ij} = 0$  results in  $F_{ij} = F_{ij}^L$ , which corresponds to a low-order mass remap or, using an advection parlance, to a “donor-cell” solution of the remap problem. Thus, at the least, the M-OBR problem admits the same solution as a conventional low-order local remapper.

The following theorem examines the relationship between M-OBR and OBR.

**Theorem 3.** *The feasible set of the M-OBR formulation (43) is a subset of the feasible set of the OBR formulation (37).*

*Proof.* The feasible sets of the OBR and M-OBR problems are given by

$$\mathcal{U}_O = \{a_{ij} \in \mathbb{R} \mid (36) \text{ hold for } i = 1, \dots, K \text{ and } j \in \mathcal{J}(N(\kappa_i))\},$$

and

$$\mathcal{U}_M = \{a_{ij} \in \mathbb{R} \mid (42) \text{ hold for } i = 1, \dots, K \text{ and } j \in \mathcal{J}(N(\kappa_i))\},$$

respectively. To show that  $\mathcal{U}_M \subseteq \mathcal{U}_O$  define the intermediate sets

$$\mathcal{U}_A = \{a_{ij} \in \mathbb{R} \mid (40) \text{ hold for } i = 1, \dots, K \text{ and } j \in \mathcal{J}(N(\kappa_i))\},$$

and

$$\mathcal{U}_B = \{a_{ij} \in \mathbb{R} \mid (41) \text{ hold for } i = 1, \dots, K \text{ and } j \in \mathcal{J}(N(\kappa_i))\},$$

corresponding to the first and the second stages in the transformation of the OBR constraints to the box constraints of M-OBR.

To prove the theorem we will show that

$$\mathcal{U}_M \subseteq \mathcal{U}_B \subseteq \mathcal{U}_A \subseteq \mathcal{U}_O.$$

Step 1:  $\mathcal{U}_M \subseteq \mathcal{U}_B$ .

Let  $\{a_{ij}\} \in \mathcal{U}_M$ . Summing up the inequalities in (42) yields

$$\begin{aligned} \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} D_i^- dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} D_i^- dF_{ji} &\leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} a_{ji} dF_{ji} \leq 0, \\ 0 \leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} a_{ji} dF_{ji} &\leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} D_i^+ dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} D_i^+ dF_{ji}. \end{aligned}$$

From (38) we see that the left hand side in the first inequality equals  $D_i^- P_i^-$  and the right hand side in the second inequality is  $D_i^+ P_i^+$ . Therefore, inequalities (41) hold for  $\{a_{ij}\}$ , i.e.  $\{a_{ij}\} \in \mathcal{U}_B$ . This proves the inclusion  $\mathcal{U}_M \subseteq \mathcal{U}_B$ .

Step 2:  $\mathcal{U}_B \subseteq \mathcal{U}_A$ .

Assume that  $\{a_{ij}\} \in \mathcal{U}_B$ . Summing up inequalities (a) and (b) in (41) gives

$$D_i^- P_i^- \leq \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \leq 0} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \geq 0} a_{ji} dF_{ji} + \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i < j}}^{dF_{ij} \geq 0} a_{ij} dF_{ij} - \sum_{\substack{j \in \mathcal{J}(N(\kappa_i)) \\ i > j}}^{dF_{ji} \leq 0} a_{ji} dF_{ji} \leq D_i^+ P_i^+$$

from where it follows that (40) hold for  $\{a_{ij}\}$ , i.e.  $\{a_{ij}\} \in \mathcal{U}_A$ . This proves the inclusion  $\mathcal{U}_B \subseteq \mathcal{U}_A$ .

Step 3:  $\mathcal{U}_B \subseteq \mathcal{U}_O$ .

Finally, let  $\{a_{ij}\} \in \mathcal{U}_A$ . Note that

$$\tilde{Q}_i^{\min} \leq D_i^- P_i^- \quad \text{and} \quad D_i^+ P_i^+ \leq \tilde{Q}_i^{\max}.$$

Therefore, inequalities (36) hold for  $\{a_{ij}\}$ , i.e.  $\{a_{ij}\} \in \mathcal{U}_O$ . This proves the inclusion  $\mathcal{U}_A \subseteq \mathcal{U}_O$ .

**Remark 4.1** *Since the M-OBR feasible set is contained in the OBR feasible set due to Theorem 3, it follows that the OBR solution is always at least as accurate as the M-OBR solution.*

### 4.3 FCR and M-OBR: Two equivalent algorithms

In this section we show that the M-OBR formulation is equivalent to the FCR algorithm in Liska et al (2010). For convenience, below we summarize the FCR formulation for the mass-density remap. Full details can be found in Liska et al (2010, Section 3).

The original motivation for FCR is to replace a global optimization problem such as OBR by a series of local problems. To this end, FCR restricts the mass fluxes in (16) to *convex* combinations of the low-order and the high-order target fluxes, i.e.

$$F_{ij} = (1 - a_{ij})F_{ij}^L + a_{ij}F_{ij}^H = F_{ij}^L + a_{ij}dF_{ij}, \quad (44)$$

where  $a_{ij} = a_{ji}$  and  $0 \leq a_{ij} \leq 1$ . The convexity assumption is motivated by analogies with the FCT approach of Kuzmin et al (2005) for advection. Except for this requirement, formula (44) is identical to the change of variables in (34). In the FCR algorithm the approximate mass fluxes in (44) are computed using the following values for the unknown coefficients:

$$a_{ij} = \begin{cases} \min\{D_i^+, D_j^-, 1\} & \text{if } dF_{ij} > 0 \\ \min\{D_i^-, D_j^+, 1\} & \text{if } dF_{ij} < 0 \end{cases} \quad \begin{matrix} 1 \leq i, j \leq K \\ \text{and } i < j \end{matrix}. \quad (45)$$

For completeness, one can set  $a_{ij} = 1$  whenever  $dF_{ij} = 0$ . In Liska et al (2010) it is shown that (45) is sufficient for the local mass-density bounds in (36) to hold.

We proceed to show that the solution of the global M-OBR problem is also given by (45). This fact establishes the equivalence of FCR and M-OBR and is a direct consequence of the following theorem.

**Theorem 4.** *The M-OBR formulation (43) is equivalent to the following set of independent, single-variable, constrained optimization problems: for  $1 \leq i, j \leq K$  and  $i < j$  solve*

$$\begin{cases} \min_{a_{ij}} (1 - a_{ij})^2 (dF_{ij})^2 & \text{subject to} \\ 0 \leq a_{ij} \leq \begin{cases} \min\{D_i^+, D_j^-\} & \text{if } dF_{ij} > 0 \\ \min\{D_i^-, D_j^+\} & \text{if } dF_{ij} < 0. \end{cases} \end{cases} \quad (46)$$

*Proof.* A flux differential  $dF_{ij}$ ,  $i < j$ , can be negative, zero or positive. If  $dF_{ij} = 0$ , we denote the variable  $a_{ij}$  as *free*, because the box constraint in (42) holds for any value of  $a_{ij}$ . Note that the terms associated with free variables do not contribute to the objective, because  $(1 - a_{ij})^2 (dF_{ij})^2 = 0$ . It follows that all free variables can be eliminated<sup>6</sup> from the optimization problem. Thus, without loss of generality we may assume that  $dF_{ij} \neq 0$ .

It is easy to see that whenever  $dF_{ij} \neq 0$ , the associated variable  $a_{ij}$  enters in exactly one constraint of type (a) and one constraint of type (b). Solving the inequalities for  $a_{ij}$  gives

$$0 \leq a_{ij} \leq D_i^+ \quad \text{and} \quad 0 \leq a_{ij} \leq D_j^-$$

for  $i < j$  and  $dF_{ij} > 0$ , and

$$0 \leq a_{ij} \leq D_i^- \quad \text{and} \quad 0 \leq a_{ij} \leq D_j^+$$

for  $i < j$  and  $dF_{ij} < 0$ . Succinctly,

$$0 \leq a_{ij} \leq \begin{cases} \min\{D_i^+, D_j^-\} & \text{if } dF_{ij} > 0 \\ \min\{D_i^-, D_j^+\} & \text{if } dF_{ij} < 0 \end{cases} \quad \begin{matrix} 1 \leq i, j \leq K \\ \text{and } i < j \end{matrix}$$

is a new set of box constraints that is completely equivalent to (43). Because each of the terms in the objective functional depends on only one variable, it follows that (43) decouples into the set of independent, single-variable, constrained optimization problems given in (46).

The equivalence of FCR and M-OBR easily follows.

<sup>6</sup> For a complete match with FCR we can set all free variables to 1.

**Corollary 4.2** *The solution  $\{a_{ij}\}$  of the M-OBR problem (43) is given by the FCR formula (45).*

*Proof.* To find the solution of the M-OBR problem we set all free variables to 1. The rest of the variables are computed by solving the decoupled optimization problems in (46). For a given pair of indices  $i < j$  let  $D_{ij} \geq 0$  denote the upper bound in the constraint of the optimization problem for the variable  $a_{ij}$ . The cost functional  $(1 - a_{ij})^2 (dF_{ij})^2$  in this problem represents a parabola with the vertex at (1,0). Therefore, the constrained minimum is achieved at the smaller of the two values  $a_{ij} = 1$  or  $a_{ij} = D_{ij}$ . It follows that whenever  $dF_{ij} \neq 0$ , the solution of the optimization problem in (46) is given by formula (45).

#### 4.4 iFCR: An iterative extension of FCR

For the purpose of numerical comparisons, we introduce a variation of the standard FCR algorithm, called iterated FCR (iFCR), originally proposed by Schär and Smolarkiewicz (1996). The key idea of iFCR is that, by definition, FCR fluxes ensure monotonicity of the solution, and can be reused as base low-order fluxes for an additional FCR flux computation. This process can be repeated *ad infinitum*. The advantage of iFCR over FCR is mainly in accuracy, at the price of increased computational cost, as the FCR flux computation has to be repeated at each iteration of the method. iFCR represents a more challenging benchmark in the analysis of performance of the OBR approach, and, of course - in the limit for a large number of iterations - may easily surpass in cost the OBR algorithm itself. The iFCR approach is described in Table 1.

**Table 1** Outline of the iFCR algorithm.

<p>Initialize solution field with initial conditions.          Predictor: Compute FCR fluxes <math>F_{ij}</math> using (44) and (45).              Define <math>F_{ij}^{(0)} = F_{ij}</math> and <math>F_{ij}^{L_i(0)} = F_{ij}^L</math>.  <b>For</b> <math>k = 0, \dots, k_{\max}</math> (<i>iFCR loop begins</i>)              Replace <math>F_{ij}^{L_i(k+1)} = F_{ij}^{(k)}</math>.              Corrector: Compute <math>F_{ij}^{(k+1)}</math> using (44) and (45).  <b>End</b> (<i>iFCR loop ends</i>)  <b>Exit</b></p>
---

## 5 Algorithms I: Exact cell intersection versus swept region flux computations

Until now all our considerations were based on the exact cell intersection formula (12). This means that in order to implement the corresponding OBR and FCR algorithms we would have to find the intersections between the cells on the old and new meshes, which is computationally expensive. Instead, we implement the OBR and FCR algorithms using *swept regions* as in Margolin and Shashkov (2003, Section 4). These are the regions swept by the movement of the sides of the old cells. As a result, the swept regions are completely determined by the coordinates of the old and new nodes and do not require the computation of cell intersections.

Recall that  $S(\kappa_i)$  is the set of all sides in cell  $\kappa_i$ . Each side  $\sigma_j$  has unique orientation  $\omega_j = +1$ , or  $-1$ , which induces orientation on the associated swept region  $\Sigma_j$ . The idea of the swept region approximation is to allow mass exchanges only between cells that share a side. In this case, the new cell masses can be approximated by the formula

$$\tilde{m}_i = m_i + \sum_{j \in \mathcal{S}(S(\kappa_i))} \omega_j F_j, \quad (47)$$

where summation is over the sides of the cell and  $F_j$  are mass fluxes corresponding to the (signed) swept regions  $\Sigma_j$  associated with side  $\sigma_j$ .

Our implementation of OBR and FCR uses (47) in lieu of the cell-intersection formula (16). Let  $\Sigma_j$  denote the swept region associated with side  $\sigma_j$  of cell  $\kappa_i$ . We define the target (high-order) fluxes as<sup>7</sup>

$$F_j^H = \int_{\Sigma_j} \rho_j^H(\mathbf{x}) dV, \quad (48)$$

where  $\rho_j^H$  is a density reconstruction that is exact for linear functions. One can show that using formula (47) with the fluxes defined in (48) gives the exact cell masses whenever the density is linear (see Margolin and Shashkov (2003)). This means that the preservation of linearity in OBR remains in full force when the method is implemented using swept regions, instead of exact cell intersections.

The situation with FCR is somewhat more complicated. In addition to the high-order fluxes (48) this method also uses the low order fluxes

$$F_j^L = \int_{\Sigma_j} \rho_j^L(\mathbf{x}) dV. \quad (49)$$

---

<sup>7</sup> Because side nodes can move in different directions swept regions are not simple extrusions of the sides, which can complicate the computation of integrals. Using Green's theorem, integrals of polynomials over swept regions can be replaced by integrals of higher-degree polynomials over the (lower-dimensional) boundaries of these regions, see Margolin and Shashkov (2003); Dukowicz and Kodis (1987). This provides an efficient way to compute the fluxes, regardless of the shape of the swept regions.

It turns out that when the low-order approximations of the new cell masses are computed using (47) and (49), instead of (16) and (31), there is no guarantee that these masses will satisfy the bounds (33), see Margolin and Shashkov (2003). Additional restrictions on the mesh movement are required to ensure that these bounds hold. A sufficient condition for (33) is that the area of the old cell  $\kappa_i$  is greater than the sum of the absolute values of all negatively signed swept regions (see Margolin and Shashkov (2003, p.279)).

The fact that (33) can be violated when FCR is implemented using swept regions has important consequences. Without (33) holding, the two OBR formulations (20) and (37) are still equivalent. However, we cannot carry out the steps in Section 4.2, which reduced (37) to the M-OBR formulation (43). Therefore, violation of (33) invalidates Proposition 4.1, Theorems 3–4, and Corollary 4.2. What this means in practice is that the feasible set in (43) may become void, in which case the M-OBR problem has no solution. As a result, the FCR solution defined in (45) ceases to be connected to the global OBR optimization problem (20) and is not guaranteed to be in its feasible set. The practical dimension of this fact is that the FCR solution may violate the local bounds. Section 7.3 provides an instructive example in two dimensions that shows the loss of monotonicity when FCR is implemented using swept regions.

## 6 Algorithms II: Solution techniques for the OBR problem

We discuss optimization techniques for the solution of the OBR problem assuming a swept-region approximation. In compact matrix / vector notation problem (20) has the form

$$\begin{aligned} \min_{\vec{F} \in \mathbb{R}^M} \quad & \frac{1}{2} (\vec{F} - \vec{F}^H)^\top (\vec{F} - \vec{F}^H) \quad \text{subject to} \\ & \vec{b}_{\min} \leq \mathbf{A}\vec{F} \leq \vec{b}_{\max}, \end{aligned} \quad (50)$$

where  $M$  denotes the number of unique flux variables,  $\vec{F}_{ij}^h$ . We also define  $\vec{F} \in \mathbb{R}^M$ ,  $\vec{F}^H \in \mathbb{R}^M$ ,  $\vec{b}_{\min} \in \mathbb{R}^K$  and  $\vec{b}_{\max} \in \mathbb{R}^K$  such that  $\vec{F}_{\iota(i,j)} = \vec{F}_{ij}^h$ ,  $\vec{F}_{\iota(i,j)}^H = \vec{F}_{ij}^T$ ,  $(\vec{b}_{\min})_i = m_i^{\min} - \tilde{m}_i$  and  $(\vec{b}_{\max})_i = m_i^{\max} - \tilde{m}_i$ , respectively, where  $\iota$  is an indexing function. Finally we let  $\mathbf{A} \in \mathbb{R}^{K \times M}$  be a matrix with entries  $-1$ ,  $0$  and  $1$  defining the inequality constraints in (20) or a related proxy (see swept-region approximation, Bochev et al (2011, Sec. 4.1,4.2)). The matrix  $\mathbf{A}$  is typically very sparse, with  $M > K$  in 2D and 3D. We abbreviate the *nonnegative orthant* as  $\mathbb{R}_+^m = \{\mathbf{x} \in \mathbb{R}^m : \mathbf{x} \geq 0\}$ .

Rather than solving (50) directly, we focus on its dual formulation. This allows us to reformulate the problem into a simpler, *bound-constrained* optimization problem.

**Theorem 5.** *Given the definitions of  $\vec{F}^H \in \mathbb{R}^M$ ,  $\vec{b}_{\min} \in \mathbb{R}^K$ ,  $\vec{b}_{\max} \in \mathbb{R}^K$ , and  $\mathbf{A} \in \mathbb{R}^{K \times M}$  from above, let us define  $J_p : \mathbb{R}^M \rightarrow \mathbb{R}$  and  $J_d : \mathbb{R}^{2K} \rightarrow \mathbb{R}$  as*

$$J_p(\vec{F}) = \frac{1}{2} \|\vec{F} - \vec{F}^H\|_2^2$$

and

$$J_d(\vec{\lambda}, \vec{\mu}) = \frac{1}{2} \|\mathbf{A}^\top \vec{\lambda} - \mathbf{A}^\top \vec{\mu}\|_2^2 - \langle \vec{\lambda}, \vec{b}_{\min} - \mathbf{A}\vec{F}^H \rangle - \langle \vec{\mu}, -\vec{b}_{\max} + \mathbf{A}\vec{F}^H \rangle.$$

Then, we have that

$$\min_{F \in \mathbb{R}^M} \left\{ J_p(\vec{F}) : \vec{b}_{\min} \leq \mathbf{A}\vec{F} \leq \vec{b}_{\max} \right\} = \min_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ J_d(\vec{\lambda}, \vec{\mu}) \right\}$$

where we call the first problem the primal and the second problem the dual. Furthermore,

$$\{\vec{F}^H + \mathbf{A}^\top(\vec{\lambda}^* - \vec{\mu}^*)\} = \arg \min_{F \in \mathbb{R}^M} \left\{ J_p(\vec{F}) : \vec{b}_{\min} \leq \mathbf{A}\vec{F} \leq \vec{b}_{\max} \right\}$$

whenever

$$(\vec{\lambda}^*, \vec{\mu}^*) \in \arg \min_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ J_d(\vec{\lambda}, \vec{\mu}) \right\}.$$

*Proof.* We begin with the observation that  $J_p$  denotes a strictly convex, continuous function and that  $\{\vec{F} \in \mathbb{R}^M : \vec{b}_{\min} \leq \mathbf{A}\vec{F} \leq \vec{b}_{\max}\}$  denotes a bounded, closed, convex set. Therefore, a unique minimum exists and is attained. Furthermore, since there exists an  $\vec{F}$  such that  $\vec{b}_{\min} < \mathbf{A}\vec{F} < \vec{b}_{\max}$ , we satisfy Slater's constraint qualification. This tells us that strong duality holds, which implies that the Lagrangian dual exists and possesses the same optimal value as the original problem.

Based on this knowledge, we notice that

$$\begin{aligned} & \min_{F \in \mathbb{R}^M} \left\{ J_p(\vec{F}) : \vec{b}_{\min} \leq \mathbf{A}\vec{F} \leq \vec{b}_{\max} \right\} \\ &= \min_{F \in \mathbb{R}^M} \max_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ J_p(\vec{F}) - \langle \mathbf{A}\vec{F} - \vec{b}_{\min}, \vec{\lambda} \rangle - \langle \vec{b}_{\max} - \mathbf{A}\vec{F}, \vec{\mu} \rangle \right\} \\ &= \max_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \min_{F \in \mathbb{R}^M} \left\{ J_p(\vec{F}) - \langle \vec{F}, \mathbf{A}^\top(\vec{\lambda} - \vec{\mu}) \rangle + \langle \vec{b}_{\min}, \vec{\lambda} \rangle - \langle \vec{b}_{\max}, \vec{\mu} \rangle \right\}. \end{aligned}$$

Next, we consider the function  $J : \mathbb{R}^M \rightarrow \mathbb{R}$  where

$$J(\vec{F}) = J_p(\vec{F}) - \langle \vec{F}, \mathbf{A}^\top(\vec{\lambda} - \vec{\mu}) \rangle$$

and  $(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}$  are fixed. We see that  $J$  is strictly convex. Therefore, it attains its unique minimum when  $\nabla J = 0$ . Specifically, when

$$\vec{F} - \vec{F}^H - \mathbf{A}^\top(\vec{\lambda} - \vec{\mu}) = 0,$$

which occurs if and only if

$$\vec{F} = \vec{F}^H + \mathbf{A}^\top(\vec{\lambda} - \vec{\mu}).$$

Therefore, we may find the optimal solution to our original problem with this equation when  $(\vec{\lambda}, \vec{\mu})$  are optimal. In addition, we may use this knowledge to simplify our derivation of the dual. Let  $\omega = \mathbf{A}^\top(\vec{\lambda} - \vec{\mu})$  and notice that

$$\begin{aligned}
 & \max_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \min_{F \in \mathbb{R}^M} \left\{ J_p(\vec{F}) - \langle \vec{F}, \mathbf{A}^\top(\vec{\lambda} - \vec{\mu}) \rangle + \langle b_{\min}, \vec{\lambda} \rangle - \langle b_{\max}, \vec{\mu} \rangle \right\} \\
 &= \max_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ J_p(\vec{F}^H + \omega) - \langle \vec{F}^H + \omega, \omega \rangle + \langle b_{\min}, \vec{\lambda} \rangle - \langle b_{\max}, \vec{\mu} \rangle \right\} \\
 &= \max_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ \frac{1}{2} \|\omega\|_2^2 - \langle \vec{F}^H, \omega \rangle - \|\omega\|_2^2 + \langle b_{\min}, \vec{\lambda} \rangle - \langle b_{\max}, \vec{\mu} \rangle \right\} \\
 &= \max_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ -\frac{1}{2} \|\mathbf{A}^\top(\vec{\lambda} - \vec{\mu})\|_2^2 - \langle \mathbf{A}\vec{F}^H, \vec{\lambda} - \vec{\mu} \rangle + \langle b_{\min}, \vec{\lambda} \rangle - \langle b_{\max}, \vec{\mu} \rangle \right\} \\
 &= \min_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ \frac{1}{2} \|\mathbf{A}^\top(\vec{\lambda} - \vec{\mu})\|_2^2 + \langle \mathbf{A}\vec{F}^H, \vec{\lambda} - \vec{\mu} \rangle - \langle b_{\min}, \vec{\lambda} \rangle + \langle b_{\max}, \vec{\mu} \rangle \right\} \\
 &= \min_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ \frac{1}{2} \|\mathbf{A}^\top \vec{\lambda} - \mathbf{A}^\top \vec{\mu}\|_2^2 - \langle \vec{\lambda}, \vec{b}_{\min} - \mathbf{A}\vec{F}^H \rangle - \langle \vec{\mu}, -\vec{b}_{\max} + \mathbf{A}\vec{F}^H \rangle \right\} \\
 &= \min_{(\vec{\lambda}, \vec{\mu}) \in \mathbb{R}_+^{2K}} \left\{ J_d(\vec{\lambda}, \vec{\mu}) \right\}.
 \end{aligned}$$

Hence, we see the equivalence between our two optimization problems and note that the equation  $\vec{F} = \vec{F}^H + \mathbf{A}^\top(\vec{\lambda} - \vec{\mu})$  allows us to find an optimal primal solution given an optimal solution to the dual.

Although the primal problem is strictly convex and possesses a unique optimal solution, the dual formulation does not. Rather, the dual problem is convex, but not strictly convex, so multiple minima may exist. Second, our formula for reconstructing the primal solution from the dual depends on an optimal dual solution. If the solution to the dual is not optimal, the reconstruction formula may generate infeasible solutions. With these points in mind, we require two additional definitions before we may proceed to our optimization algorithm.

**Definition 6.1** We define the diagonal operator,  $\text{Diag} : \mathbb{R}^m \rightarrow \mathbb{R}^{m \times m}$ , as

$$[\text{Diag}(\mathbf{x})]_{ij} = \begin{cases} \mathbf{x}_i & i = j \\ 0 & i \neq j \end{cases}.$$

**Definition 6.2** For some symmetric, positive semidefinite  $\mathbf{H} \in \mathbb{R}^{m \times m}$  and some  $\vec{b} \in \mathbb{R}^m$ , we define the operator  $v_{\mathbf{H}, \vec{b}} : \mathbb{R}^m \rightarrow \mathbb{R}^m$  as

$$v_{\mathbf{H}, \vec{b}}(\mathbf{x}) = \begin{cases} \mathbf{x}_i & [\mathbf{H}\mathbf{x} + \vec{b}]_i \geq 0 \\ 1 & [\mathbf{H}\mathbf{x} + \vec{b}]_i < 0 \end{cases}.$$

When both  $\mathbf{H}$  and  $\vec{b}$  are clear from the context, we abbreviate this function as  $v$ .

In order to solve the dual optimization problem, we use a simplified version of the locally convergent Coleman-Li algorithm (Coleman and Li (1996)). The key to this algorithm follows from the following lemma.

**Lemma 6.1** *Let  $\mathbf{H} \in \mathbb{R}^{m \times m}$  be symmetric, positive semidefinite and let  $\vec{b} \in \mathbb{R}^m$ . Then, for some  $\mathbf{x}^* \geq 0$ , we have that*

$$\mathbf{x}^* \in \arg \min_{\mathbf{x} \in \mathbb{R}_+^m} \left\{ \frac{1}{2} \langle \mathbf{H}\mathbf{x}, \mathbf{x} \rangle + \langle \vec{b}, \mathbf{x} \rangle \right\} \iff \text{Diag}(v(\mathbf{x}^*))(\mathbf{H}\mathbf{x}^* + \vec{b}) = 0.$$

*Proof.* We begin with the observation that since  $\mathbf{H}$  is symmetric, positive semidefinite, the problem

$$\min_{\mathbf{x} \in \mathbb{R}_+^m} \left\{ \frac{1}{2} \langle \mathbf{H}\mathbf{x}, \mathbf{x} \rangle + \langle \vec{b}, \mathbf{x} \rangle \right\}$$

represents a convex optimization problem with a coercive objective and a closed, convex set of constraints. Therefore, a minimum exists and the first order optimality conditions become sufficient for optimality.

In the forward direction, we assume that we have an optimal pair  $(\mathbf{x}^*, \vec{\lambda}^*)$  that satisfy the first order optimality conditions,

$$\begin{aligned} \mathbf{H}\mathbf{x}^* + \vec{b} - \vec{\lambda}^* &= 0 \\ \mathbf{x}^* &\geq 0, \vec{\lambda}^* \geq 0 \\ \text{Diag}(\mathbf{x}^*)\vec{\lambda}^* &= 0. \end{aligned}$$

According to these equations,  $\vec{\lambda}^* = \mathbf{H}\mathbf{x}^* + \vec{b}$  and  $\vec{\lambda}^* \geq 0$ . This implies that  $\mathbf{H}\mathbf{x}^* + \vec{b} \geq 0$ . Therefore, according to the definition of  $v$ ,  $[\text{Diag}(v(\mathbf{x}^*))]_{ii} = \mathbf{x}_i^*$  for all  $i$ . This tells us that

$$[\text{Diag}(v(\mathbf{x}^*))(\mathbf{H}\mathbf{x}^* + \vec{b})]_i = \mathbf{x}_i^* [\mathbf{H}\mathbf{x}^* + \vec{b}]_i = \mathbf{x}_i^* \vec{\lambda}_i^* = 0$$

where the final equality follows from our fourth optimality condition, complementary slackness.

In the reverse direction, we assume that  $\text{Diag}(v(\mathbf{x}^*))(\mathbf{H}\mathbf{x}^* + \vec{b}) = 0$  for some  $\mathbf{x}^* \in \mathbb{R}_+^m$ . Since the problem

$$\min_{\mathbf{x} \in \mathbb{R}_+^m} \left\{ \frac{1}{2} \langle \mathbf{H}\mathbf{x}, \mathbf{x} \rangle + \langle \vec{b}, \mathbf{x} \rangle \right\}$$

represents a convex optimization problem, it is sufficient to show that the first order optimality conditions hold for  $\mathbf{x}^*$  and some  $\vec{\lambda}^*$ . Of course, we immediately see that we satisfy primal feasibility since  $\mathbf{x}^* \geq 0$  by assumption.

Due to the definition of  $v$ , our initial assumption implies that  $\mathbf{H}\mathbf{x}^* + \vec{b} \geq 0$ . If this were not the case, then there would exist an  $i$  such that  $[\mathbf{H}\mathbf{x}^* + \vec{b}]_i < 0$ . In this case, we see that  $[v(\mathbf{x}^*)]_i = 1$  and that  $[\text{Diag}(v(\mathbf{x}^*))(\mathbf{H}\mathbf{x}^* + \vec{b})]_i = [\mathbf{H}\mathbf{x}^* + \vec{b}]_i < 0$ , which contradicts our initial assumption. Therefore,  $\mathbf{H}\mathbf{x}^* + \vec{b} \geq 0$ . As a result, let us set  $\vec{\lambda}^* = \mathbf{H}\mathbf{x}^* + \vec{b}$ . This allows us to satisfy our first optimality condition,  $\mathbf{H}\mathbf{x}^* + \vec{b} - \vec{\lambda}^* = 0$  as well as our third,  $\vec{\lambda}^* \geq 0$ .

In order to show that we satisfy complementary slackness, we combine our initial assumption as well as our knowledge that  $\mathbf{H}\mathbf{x}^* + \vec{b} \geq 0$  to see that

$$\begin{aligned} 0 &= \text{Diag}(v(\mathbf{x}^*))(\mathbf{H}\mathbf{x}^* + \vec{b}) \\ &= \text{Diag}(\mathbf{x}^*)(\mathbf{H}\mathbf{x}^* + \vec{b}) \\ &= \text{Diag}(\mathbf{x}^*)\vec{\lambda}^*. \end{aligned}$$

Therefore, we satisfy our final optimality condition and, hence,  $\mathbf{x}^*$  denotes an optimal solution to the optimization problem.

The above lemma allows us to recast a bound-constrained, convex quadratic optimization problem into a piecewise differentiable system of equations. In order to solve this system of equations, we apply Newton's method. Before we do so, we require one additional definition and a lemma.

**Definition 6.3** For some symmetric, positive semidefinite  $\mathbf{H} \in \mathbb{R}^{m \times m}$  and some  $\vec{b} \in \mathbb{R}^m$ , we define the operator  $K_{\mathbf{H}, \vec{b}} : \mathbb{R}^m \rightarrow \mathbb{R}^{m \times m}$  as

$$[K_{\mathbf{H}, \vec{b}}(\mathbf{x})]_{ij} = \begin{cases} 1 & [\mathbf{H}\mathbf{x} + \vec{b}]_i \geq 0 \\ 0 & [\mathbf{H}\mathbf{x} + \vec{b}]_i < 0 \end{cases}.$$

When both  $\mathbf{H}$  and  $\vec{b}$  are clear from the context, we abbreviate this operator as  $K$ .

**Lemma 6.2** Let  $\mathbf{H} \in \mathbb{R}^{m \times m}$  be symmetric, positive definite,  $\vec{b} \in \mathbb{R}^m$ , and define the function  $J : \mathbb{R}^m \rightarrow \mathbb{R}$  as

$$J(\mathbf{x}) = \text{Diag}(v(\mathbf{x}))(\mathbf{H}\mathbf{x} + \vec{b}).$$

Then, we have that

$$J'(\mathbf{x}) = K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + \text{Diag}(v(\mathbf{x}))\mathbf{H}.$$

*Proof.* Let us begin by assessing the derivative of  $v$ . We notice that

$$[v(\mathbf{x} + t\vec{\eta})]_i = \begin{cases} \mathbf{x}_i + t\vec{\eta}_i & [\mathbf{H}\mathbf{x} + \vec{b}]_i \geq 0 \\ 1 & [\mathbf{H}\mathbf{x} + \vec{b}]_i < 0 \end{cases}.$$

Therefore, from a piecewise application of Taylor's theorem, we see that

$$[v'(\mathbf{x})\vec{\eta}]_i = \begin{cases} \vec{\eta}_i & [\mathbf{H}\mathbf{x} + \vec{b}]_i \geq 0 \\ 0 & [\mathbf{H}\mathbf{x} + \vec{b}]_i < 0 \end{cases}.$$

Next, we apply a similar technique to  $J$ . Let us define  $g : \mathbb{R}^m \rightarrow \mathbb{R}$  so that  $g(\mathbf{x}) = \mathbf{H}\mathbf{x} + \vec{b}$ . Then, we see that

$$\begin{aligned}
J(\mathbf{x} + t\vec{\eta}) &= \text{Diag}(v(\mathbf{x} + t\vec{\eta}))(\mathbf{H}(\mathbf{x} + t\vec{\eta}) + \vec{b}) \\
&= \text{Diag}(v(\mathbf{x}) + tv'(\mathbf{x})\vec{\eta} + o(|t|))(\mathbf{H}\mathbf{x} + \vec{b} + t\vec{\eta}) \\
&= \text{Diag}(v(\mathbf{x}))g(\vec{x}) + t(\text{Diag}(v(\mathbf{x}))\mathbf{H}\vec{\eta} + \text{Diag}(v'(\mathbf{x})\vec{\eta})g(\vec{x})) + o(|t|).
\end{aligned}$$

Hence, from a piecewise application of Taylor's theorem, we have that

$$\begin{aligned}
J'(\mathbf{x})\vec{\eta} &= \text{Diag}(v(\mathbf{x}))\mathbf{H}\vec{\eta} + \text{Diag}(v'(\mathbf{x})\vec{\eta})(\mathbf{H}\mathbf{x} + \vec{b}) \\
&= \text{Diag}(v(\mathbf{x}))\mathbf{H}\vec{\eta} + K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b})\vec{\eta}.
\end{aligned}$$

Therefore,  $J'(\mathbf{x}) = K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + \text{Diag}(v(\mathbf{x}))\mathbf{H}$ .

The preceding lemma allows us to formulate Newton's method where we seek a step  $\vec{p} \in \mathbb{R}^m$  such that  $J'(\mathbf{x})\vec{p} = -J(\mathbf{x})$ . Although the operator  $J'(\mathbf{x})$  is well structured, it is nonsymmetric. We symmetrize the system as follows.

**Definition 6.4** For some symmetric, positive semidefinite  $\mathbf{H} \in \mathbb{R}^{m \times m}$  and some  $\vec{b} \in \mathbb{R}^m$ , we define the operator  $D_{\mathbf{H}, \vec{b}} : \mathbb{R}_+^m \rightarrow \mathbb{R}^{m \times m}$  as

$$D_{\mathbf{H}, \vec{b}}(\mathbf{x}) = \text{Diag}(v_{\mathbf{H}, \vec{b}}(\mathbf{x}))^{1/2}.$$

When both  $\mathbf{H}$  and  $\vec{b}$  are clear from the context, we abbreviate this operator as  $D$ .

**Lemma 6.3** Let  $\mathbf{H} \in \mathbb{R}^{m \times m}$  be symmetric, positive semidefinite and let  $\vec{b} \in \mathbb{R}^m$ . Then, we have that

$$\begin{aligned}
(K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + \text{Diag}(v(\mathbf{x}))\mathbf{H})\vec{p} &= -\text{Diag}(v(\mathbf{x}))(\mathbf{H}\mathbf{x} + \vec{b}) \\
\iff (K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})\mathbf{H}D(\mathbf{x}))\vec{q} &= -D(\mathbf{x})(\mathbf{H}\mathbf{x} + \vec{b})
\end{aligned}$$

where  $\vec{p} = D(\mathbf{x})\vec{q}$ .

*Proof.* Notice that

$$\begin{aligned}
0 &= (K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + \text{Diag}(v(\mathbf{x}))\mathbf{H})\vec{p} + \text{Diag}(v(\mathbf{x}))(\mathbf{H}\mathbf{x} + \vec{b}) \\
&= (K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})^2\mathbf{H})\vec{p} + D(\mathbf{x})^2(\mathbf{H}\mathbf{x} + \vec{b}) \\
&= D(\mathbf{x})((D(\mathbf{x})^{-1}K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})\mathbf{H})\vec{p} + D(\mathbf{x})(\mathbf{H}\mathbf{x} + \vec{b})) \\
&= D(\mathbf{x})((D(\mathbf{x})^{-1}K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})\mathbf{H})D(\mathbf{x})\vec{q} + D(\mathbf{x})(\mathbf{H}\mathbf{x} + \vec{b})) \\
&= D(\mathbf{x})((K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})\mathbf{H}D(\mathbf{x}))\vec{q} + D(\mathbf{x})(\mathbf{H}\mathbf{x} + \vec{b})),
\end{aligned}$$

which occurs if and only if

$$0 = (K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})\mathbf{H}D(\mathbf{x}))\vec{q} + D(\mathbf{x})(\mathbf{H}\mathbf{x} + \vec{b})$$

since  $D(\mathbf{x})$  is nonsingular.

Properly, we require a line search to ensure feasible iterates. However, we can be far more aggressive in practice. In order to initialize the algorithm, we use the starting iterate of  $(\vec{\lambda}, \vec{\mu}) = (\vec{0}, \vec{0})$ . This corresponds to a primal solution where  $\vec{F} = \vec{F}^H$ . Since the optimal solution to the primal problem is close to the target  $\vec{F}^H$ , we expect the optimal solution to the dual problem to reside in a neighborhood close to zero. As a result, Newton's method should converge quadratically to the solution with a step size equal to one. Therefore, we ignore the feasibility constraint and always use a unit step size. Sometimes, this allows the dual solution to become slightly infeasible, but the amount of infeasibility tends to be small. In practice, the corresponding primal solution is always feasible and produces good results. In order to allow infeasible solutions, we must use the original formulation of Newton's method rather than the symmetric reformulation. Namely, the operator  $D$  becomes ill-defined for infeasible points.

When we combine the above pieces, we arrive at the final algorithm.

**Table 2** Dual algorithm for the solution of the remap problem

<p>1. Define <math>H \in \mathbb{R}^{2K \times 2K}</math> and <math>b \in \mathbb{R}^{2K}</math> as</p> $\mathbf{H} = \begin{bmatrix} \mathbf{A}\mathbf{A}^T & -\mathbf{A}\mathbf{A}^T \\ -\mathbf{A}\mathbf{A}^T & \mathbf{A}\mathbf{A}^T \end{bmatrix} \quad \vec{b} = \begin{bmatrix} \mathbf{A}\vec{F}^H - \vec{b}_{\min} \\ -\mathbf{A}\vec{F}^H + \vec{b}_{\max} \end{bmatrix}.$ <p>2. Initialize <math>\mathbf{x} = \vec{0}</math>.</p> <p>3. Until <math>\ \text{Diag}(v(\mathbf{x}))(\mathbf{H}\mathbf{x} + \vec{b})\ </math> becomes small or we exceed a fixed number of iterations.</p> <p>a. When feasible, solve</p> $(K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + D(\mathbf{x})\mathbf{H}D(\mathbf{x}))\vec{q} = -D(\mathbf{x})(\mathbf{H}\mathbf{x} + \vec{b})$ <p>and set <math>\vec{p} = D(\mathbf{x})\vec{q}</math>. Otherwise, solve</p> $(K(\mathbf{x})\text{Diag}(\mathbf{H}\mathbf{x} + \vec{b}) + \text{Diag}(v(\mathbf{x}))\mathbf{H})\vec{p} = -\text{Diag}(v(\mathbf{x}))(\mathbf{H}\mathbf{x} + \vec{b}).$ <p>b. Set <math>\mathbf{x} = \mathbf{x} + \vec{p}</math>.</p>
---

## 7 A few instructive examples

In this section we present three numerical examples that illustrate the advantages of the OBR formulation in comparison to the M-OBR formulation. Because, as shown in Corollary 4.2, the solution of the M-OBR problem (43) is equivalent to the one given by the FCR algorithm, our study effectively compares and contrasts the fundamental properties of OBR and FCR; henceforth, we denote the M-OBR / FCR methods and algorithms by the common acronym M-OBR (FCR).

Most notably, the three examples reveal that the conditions on the mesh motion for OBR, given in Theorem 2, are much less restrictive than those for M-OBR (FCR). First, we demonstrate on a simple three-cell example in one spatial dimension that for certain mesh motions M-OBR (FCR) does not preserve the shape of a given density function, while OBR does. Second, we construct a related example for which M-OBR (FCR) does not preserve linear density functions under mesh motions admissible by OBR. Finally, we give a 9-cell example in two spatial dimensions for which a commonly used M-OBR (FCR) algorithm based on swept regions, see Section 5, does not preserve monotonicity, while OBR does. In the following, we refer to Section 3 for relevant notation.

We will also compare some of the numerical results with the iFCR algorithm. In particular, unless otherwise specified, iFCR(k) indicates the k-th iterate of the iFCR algorithm.

### ***7.1 An example of mesh movement in which OBR preserves shape and M-OBR (FCR) does not***

The goal of this section is to show that the smaller feasible set of the M-OBR (FCR) formulation (43) can limit its ability to accurately preserve the shape of a given density function. To this end we design a “torture” test example that shows how the shape of a given “peak” density function can be changed by M-OBR (FCR) into a step-function profile. Of course, because M-OBR (FCR) and FCR are equivalent, the same will hold true for the FCR solution.

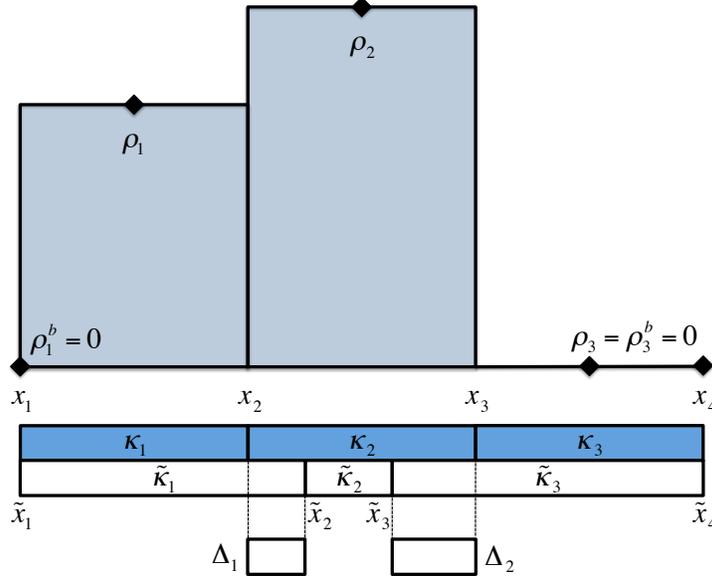
A schematic of the torture test is shown in Figure 4. The computational domain is given by the unit interval,  $\Omega = [0, 1]$ . The old mesh  $K_h(\Omega)$  is defined by a uniform partition of the unit interval into 3 cells using the vertices  $x_1 = 0, x_2 = 1/3, x_3 = 2/3$  and  $x_4 = 1$ . The nodes of the new mesh  $\tilde{K}_h(\Omega)$  are set to  $\tilde{x}_1 = x_1, \tilde{x}_2 = x_2 + \Delta_1, \tilde{x}_3 = x_3 - \Delta_2$  and  $\tilde{x}_4 = x_4$ , where  $\Delta_1 > 0$  and  $\Delta_2 > 0$  are such that  $\Delta_1 + \Delta_2 < 1/3$ ; see Figure 4. In other words, the new mesh is defined by compressing the middle cell of the old mesh. Note that  $\tilde{K}_h(\Omega)$  satisfies the locality assumption (3) and that

$$x_2 < \tilde{x}_2 \quad \text{and} \quad \tilde{x}_3 < x_3. \quad (51)$$

To complete the specification of the torture test we prescribe the mean density values  $\rho_1, \rho_2, \rho_3$  on the old cells and boundary values  $\rho_1^b = 0, \rho_3^b = 0$  at the endpoints. The mean density values are subject to the conditions

$$\rho_1 > \rho_3, \quad \rho_2 \geq \rho_1, \quad \text{and} \quad \rho_2 \geq \rho_3. \quad (52)$$

Specific numbers will be given momentarily. To explain these choices it is necessary to examine the structure of the feasible set of (37) and its modification (43), specialized to the torture test. As before, we follow the rule that the antisymmetry of fluxes and the symmetry of coefficients  $a_{ij}$  are enforced by using index pairs  $\{i, j\}$  for



**Fig. 4** Specification of the “torture” test for shape preservation. The new mesh is defined by compressing the middle cell of the old mesh. The mean density values are subject to the conditions that  $\rho_1 > \rho_3$  and that  $\rho_2$  is the largest value. The results reported in this section correspond to  $\Delta_1 = \Delta_2 = 0.14$ ,  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ .

which  $i < j$ . In the case of the torture test, which has three cells, there are two such pairs, given by  $\{1, 2\}$  and  $\{2, 3\}$ . Therefore, the independent fluxes are  $F_{12}$  and  $F_{23}$ , the unknown coefficients are  $a_{12}$  and  $a_{23}$ , and the OBR problem (37) specializes to

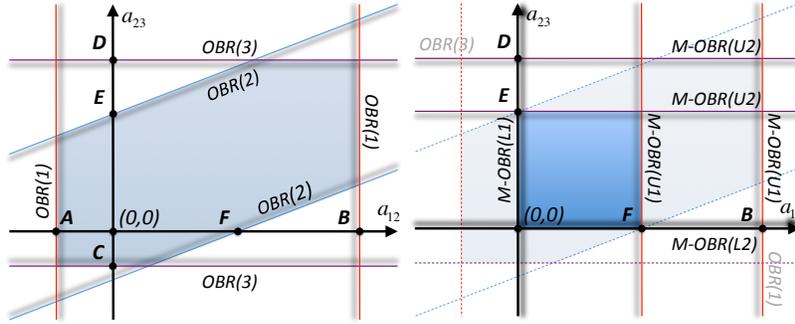
$$\begin{cases} \min_{a_{12}, a_{23}} \{(1 - a_{12})^2 (dF_{12})^2 + (1 - a_{23})^2 (dF_{23})^2\} & \text{subject to} \\ \tilde{Q}_1^{\min} \leq a_{12} dF_{12} \leq \tilde{Q}_1^{\max} & (1) \\ \tilde{Q}_2^{\min} \leq a_{23} dF_{23} - a_{12} dF_{12} \leq \tilde{Q}_2^{\max} & (2) \\ \tilde{Q}_3^{\min} \leq -a_{23} dF_{23} \leq \tilde{Q}_3^{\max} & (3) \end{cases} \quad (53)$$

Regarding the M-OBR (FCR) formulation (37), a simple but tedious calculation shows that  $dF_{12} > 0$  and  $dF_{23} > 0$  whenever (i) the middle cell is compressed, i.e. (51) holds, and (ii) the first condition in (52) holds, i.e.  $\rho_1 > \rho_3$ . As a result, the M-OBR (FCR) problem assumes the form

Point	A	B	C	D	E	F
Definition	$\frac{\tilde{Q}_1^{\min}}{dF_{12}}$	$\frac{\tilde{Q}_1^{\max}}{dF_{12}}$	$\frac{\tilde{Q}_3^{\max}}{-dF_{23}}$	$\frac{\tilde{Q}_3^{\min}}{-dF_{23}}$	$\frac{\tilde{Q}_2^{\max}}{dF_{23}}$	$\frac{\tilde{Q}_2^{\min}}{-dF_{12}}$
Value	-25.04	4.10	-20.53	8.62	0.00	3.28

**Table 3** Control points for the feasible sets of the OBR (53) and the M-OBR (FCR) (54) problems and their values for  $\Delta_1 = \Delta_2 = 0.14$ ,  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ .

$$\begin{cases} \min_{a_{12}, a_{23}} \{(1 - a_{12})^2 (dF_{12})^2 + (1 - a_{23})^2 (dF_{23})^2\} & \text{subject to} \\ 0 \leq a_{12} \leq \min\{D_1^+, D_2^-\} & (1) \\ 0 \leq a_{23} \leq \min\{D_2^+, D_3^-\} & (2) \end{cases} \quad (54)$$



**Fig. 5** Structure of the OBR (left pane) and M-OBR (FCR) (right pane) feasible sets when  $dF_{12}$  and  $dF_{23}$  are positive. The strips between the pairs of lines marked by  $OBR(1)$ ,  $OBR(2)$  and  $OBR(3)$  correspond to the three inequality constraints in (53). The lines marked by  $OBR(2)$  have positive slopes given by  $dF_{12}/dF_{23}$ . The lines marked by  $M-OBR(U1)$  and  $M-OBR(U2)$  represent the two upper upper bounds in the two inequality constraints in (54), respectively. The lower bounds correspond to the coordinate axes and are identified by  $M-OBR(L1)$  and  $M-OBR(L2)$ , respectively. The shadows point towards the interiors of the domains defined by the constraints. It is evident that the feasible set of M-OBR (FCR) is a subset of the feasible set of OBR.

The left and the right panes in Figure 5 show cartoons of the feasible sets of (53) and (54), respectively. The horizontal and the vertical axes in these plots correspond to the unknowns  $a_{12}$  and  $a_{23}$ , respectively. The strips between the pairs of lines marked by  $OBR(1)$ ,  $OBR(2)$  and  $OBR(3)$  correspond to the three inequality constraints in (53). Note that the slope of the lines marked by  $OBR(2)$  is given by  $dF_{12}/dF_{23}$  and is therefore positive. The lines marked by  $M-OBR(U1)$  and  $M-OBR(U2)$  represent the two upper upper bounds in the two inequality constraints in (54), respectively. The lower bounds coincide with the coordinate axes and are marked by  $M-OBR(L1)$  and  $M-OBR(L2)$ , respectively.

	$i = 1$	$i = 2$	$i = 3$
$\tilde{Q}_i^{\min}$	-40.66	-5.33	-14.00
$\tilde{Q}_i^{\max}$	6.66	0.00	33.33
$dF_{i,i+1}$	1.62	1.62	—

**Table 4** Numerical values for the lower and the upper bounds and the flux differentials in (53)–(54) corresponding to  $\Delta_1 = \Delta_2 = 0.14$ ,  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ .

The relation between the two feasible sets can be understood by examining the points **A**, **B**, **C**, **D**, **E** and **F**. The first pair of points corresponds to the lower and upper bounds on  $a_{12}$  imposed by the first constraint in (53). The second pair, i.e., **C**, and **D**, corresponds to the lower and upper bounds on  $a_{23}$  imposed by the third constraint in (53). The last two points correspond to the intercepts of the lines associated with the upper and lower bounds in the second constraint in (53) with the vertical and horizontal coordinate axes, respectively. The definitions of these points and their values corresponding to the actual test data used in the study are summarized in Table 3.

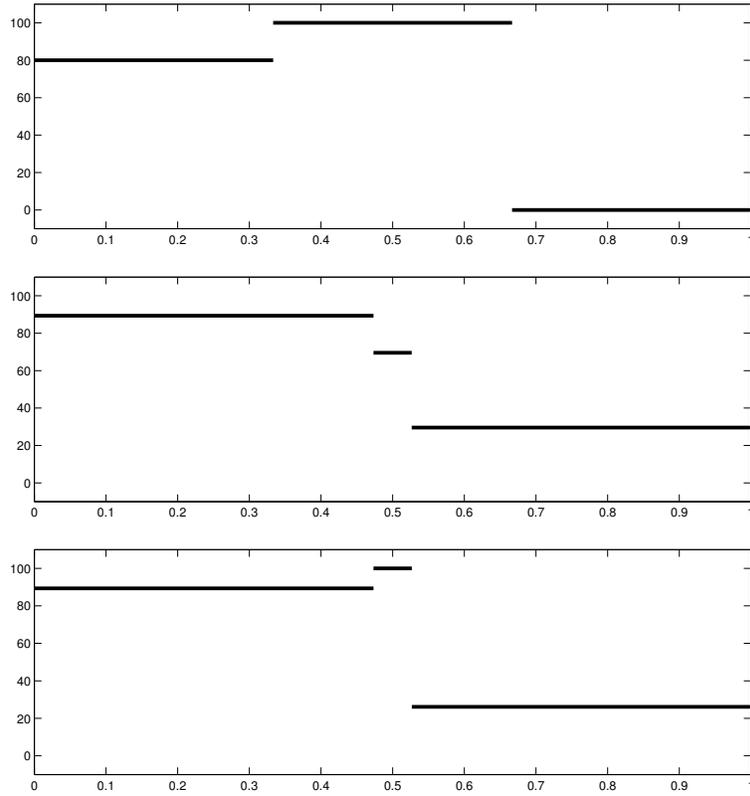
To explain the construction of the torture test, note that the shape of the M-OBR (FCR) feasible set is completely determined by the positions of **E** and **F** along the vertical and the horizontal coordinate axes. This is a consequence of the worst-case analysis used to derive the constraints of (54). Consequently, by moving **E** to the origin the M-OBR (FCR) feasible set can be reduced to a line extending from the origin to point **F**. This removes the point  $(1, 1)$  from the feasible set and forces the M-OBR (FCR) formulation to pick a solution that corresponds to remap by low-order fluxes. By moving **E** to the origin we also shrink the OBR feasible set. However, because the lines corresponding to the second constraint have positive slopes, they can be chosen in such a way that  $(1, 1)$  remains in this feasible set.

In order to move **E** to the origin we need to set  $\tilde{Q}_2^{\max}/dF_{23} = 0$ . It is not hard to see that this is true whenever (i) the middle cell is compressed, i.e. (51) holds, and (ii) the second condition in (52), i.e.  $\rho_2^{\max} = \rho_2$  holds.

Figure 6 compares the OBR and M-OBR (FCR) solutions on the new mesh for  $\Delta_1 = \Delta_2 = 0.14$ ,  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and boundary values  $\rho_1^b = \rho_3^b = 0$ . Table 4 shows the corresponding values of the lower and the upper inequality bounds as well as the values of the flux differentials in (53)–(54).

The initial density function has the shape of a “peak” and is shown in the top pane of Figure 6. The bottom pane in Figure 6 shows clearly that the OBR solution preserves this shape on the new mesh. However, as one can see from the middle pane in Figure 6, the M-OBR (FCR) solution changes the shape of the peak to a step-function profile on the new mesh. We note that the iFCR(2) method delivers results identical to those of the OBR method.

The constraint sets of (53) and (54) for this example are compared in Figure 7. We see that  $(1, 1)$  is included in the former but not in the latter. This is a consequence of the worst-case analysis used to obtain the constraint set in (54).



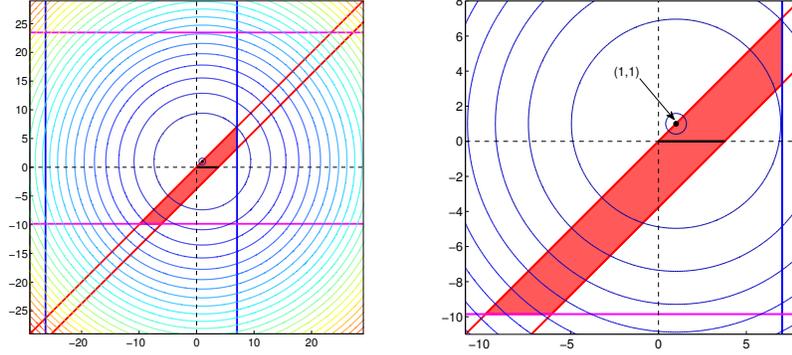
**Fig. 6** Initial density function (top pane), M-OBR (FCR) solution (middle pane) and OBR solution (bottom pane) for  $\Delta_1 = \Delta_2 = 0.14$ ,  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ . The OBR solution preserves the shape of the original density function, while the M-OBR (FCR) solution does not. The iFCR(2) method delivers results identical to those of the OBR method.

## 7.2 An example in which OBR preserves linear densities and M-OBR (FCR) does not

In this section, we investigate the differences between OBR and M-OBR (FCR) concerning the preservation of linear density functions. The basic setup is closely related to the previous example. The specification of the computational mesh is identical. The density function is given by

$$\rho(x) = x \quad 0 \leq x \leq 1,$$

i.e.  $\rho_1 = 1/6$ ,  $\rho_2 = 1/2$ ,  $\rho_3 = 5/6$ ,  $\rho_1^b = 0$  and  $\rho_2^b = 1$ . We consider a series of compression increments  $\Delta_1, \Delta_2$  given by



**Fig. 7** Level sets of the objective functional and the feasible sets of problems (53) and (54) for  $\Delta_1 = \Delta_2 = 0.14$ ,  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ . The regions between horizontal (magenta), slanted (red) and vertical (blue) lines on the left pane correspond to the first, second and third constraints in the OBR problem (53). Their intersection (red region) gives the OBR feasible set which contains the point  $(1, 1)$ . The feasible set of M-OBR (FCR) is given by the solid horizontal segment (black) and does not contain the point  $(1, 1)$ . The right pane shows a zoom of the OBR and M-OBR (FCR) feasible sets.

$$\Delta_1 = \Delta_2 = \frac{\ell - 1}{6\ell},$$

where  $\ell = \{7, 8, 9, 10, 100, 1000\}$ , resulting in  $\ell$ -fold compressions of the middle cell.

The initial linear density function is remapped onto the compressed mesh and then back onto the original mesh, where we record the  $L_2$  error between the thus obtained and the original density. Table 5 clearly shows that while OBR preserves linear densities for arbitrary compressions of the middle cell, M-OBR (FCR) is linearity-preserving only for  $\ell \leq 8$ . The iFCR algorithm, for a large number of iterations, recaptures the behavior of OBR.

The root cause for the loss of the linearity preservation in this example is the same as for the loss of shape preservation in the last section. The M-OBR (FCR) problem (54) preserves linearity if and only if the unconstrained minimizer  $(1, 1)$  of the functional in (54) is included in its feasible set. When the middle cell is compressed the feasible set of M-OBR (FCR) shrinks and eventually ceases to contain the point  $(1, 1)$ .

Ultimately, the loss of linearity preservation in the M-OBR (FCR) is a function of the mesh movement. To prevent the loss of this important property we recommend that M-OBR (FCR) implementations include the following test to determine the admissible mesh motions. Given a candidate new mesh, compute the quantities  $P_i^-$ ,  $D_i^-$  and  $P_i^+$ ,  $D_i^+$  defined in (38)–(39), for the monomial  $x$  in one dimension, monomials  $x$  and  $y$  in two dimensions and monomials  $x$ ,  $y$  and  $z$  in three dimensions. Accept the mesh if and only if  $D_i^- \geq 1$  and  $D_i^+ \geq 1$ , whenever  $P_i^- < 0$  and

$P_i^+ > 0$ , respectively. This condition guarantees that  $a_{ij} = 1$  are in the feasible set of the M-OBR (FCR) problem (46).

	$\ell = 7$	$\ell = 8$	$\ell = 9$	$\ell = 10$	$\ell = 100$	$\ell = 1000$
OBR	<b>1.67e-16</b>	<b>0</b>	<b>3.20e-17</b>	<b>3.58e-17</b>	<b>1.63e-16</b>	<b>1.95e-14</b>
FCR	<b>4.53e-17</b>	<b>3.58e-17</b>	2.32e-03	4.46e-03	2.09e-02	2.25e-02
iFCR(2)	<b>1.24e-16</b>	<b>1.57e-16</b>	<b>1.57e-16</b>	<b>1.57e-16</b>	3.31e-02	3.87e-02
iFCR(20)	<b>1.24e-16</b>	<b>1.57e-16</b>	<b>1.57e-16</b>	<b>1.57e-16</b>	<b>1.92e-16</b>	3.30e-02
iFCR(200)	<b>1.24e-16</b>	<b>1.57e-16</b>	<b>1.57e-16</b>	<b>1.57e-16</b>	<b>1.92e-16</b>	<b>1.69e-15</b>

**Table 5**  $L_2$  errors in the OBR, M-OBR (FCR) and iFCR remap of a linear density function in one dimension, for different compression ratios  $\ell : 1$  of the middle cell. Errors close to machine precision are highlighted. OBR preserves linear densities for arbitrarily compressed middle cells, while M-OBR (FCR) does not. iFCR is linearity preserving given a sufficient, possibly very large, number of iterations.

We also investigated whether the good performance of the iFCR algorithm for large number of iterations depends on OBR recovering the high-order flux in computations. Figures 8–9 show detailed computations in which it is clear that similar results between iFCR and OBR can be obtained also in the case in which OBR does not recover the target fluxes. This indicates that for a sufficiently large number of iterations the iFCR solution converges to the OBR solution. Our conjecture is that iFCR may be a solution procedure for OBR.

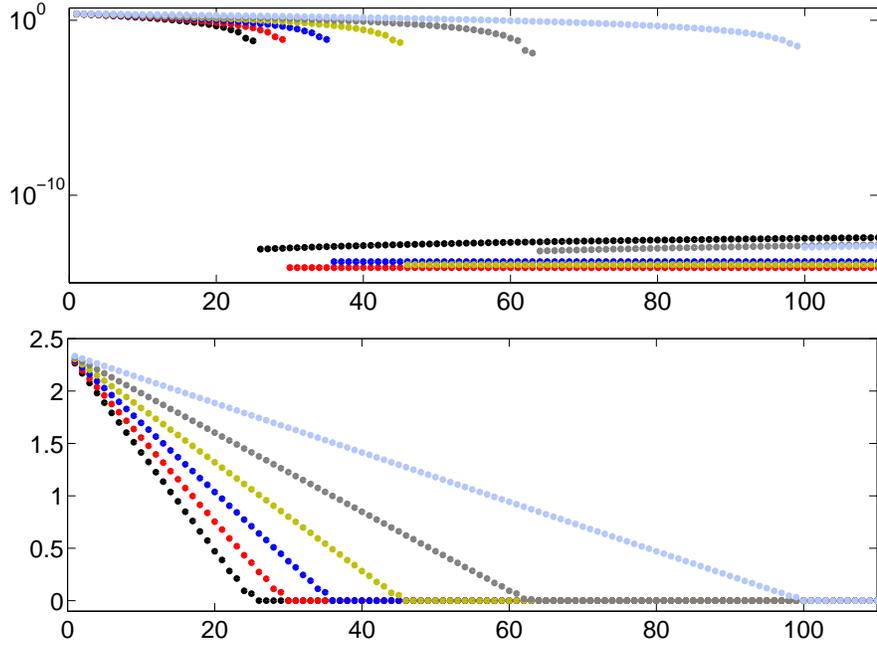
### 7.3 OBR preserves monotonicity when M-OBR (FCR) does not

Motivated by the one-dimensional examples we devise a simple 9-cell test that examines the fundamental properties of OBR and M-OBR (FCR) in two dimensions. The test is a tensor-product version of the one-dimensional torture test. The computational domain is given by the product of unit intervals,  $\Omega = [0, 1] \times [0, 1]$ . The old mesh  $K_h(\Omega)$  is defined by a uniform partition of the unit intervals in  $x$  and  $y$  direction into 3 cells using the vertices  $x_1 = 0$ ,  $x_2 = 1/3$ ,  $x_3 = 2/3$  and  $x_4 = 1$  and  $y_1 = 0$ ,  $y_2 = 1/3$ ,  $y_3 = 2/3$  and  $y_4 = 1$ , respectively. The nodes of the new mesh  $\tilde{K}_h(\Omega)$  are set to

$$\tilde{x}_1 = x_1, \tilde{x}_2 = x_2 + \Delta_1^x, \tilde{x}_3 = x_3 - \Delta_2^x, \tilde{x}_4 = x_4,$$

$$\tilde{y}_1 = y_1, \tilde{y}_2 = y_2 + \Delta_1^y, \tilde{y}_3 = y_3 - \Delta_2^y, \tilde{y}_4 = y_4,$$

where  $\Delta_1^{x,y} > 0$  and  $\Delta_2^{x,y} > 0$  are such that  $\Delta_1^x + \Delta_2^x < 1/3$  and  $\Delta_1^y + \Delta_2^y < 1/3$ . In other words, as in one spatial dimension, the new mesh is defined by compressing



**Fig. 8** Euclidean norm of the difference (vertical axes) between the computed iFCR and OBR fluxes for problems (53) and (54) for increasing numbers of iFCR iterations (horizontal axes). The compression of the middle cell is given by  $\Delta_1 = \Delta_2 \in \{0.1660$  (black),  $0.1661$  (red),  $0.1662$  (blue),  $0.1663$  (green),  $0.1664$  (gray),  $0.1665$  (cyan) $\}$ . The density profile is  $\rho_1 = 80$ ,  $\rho_2 = 100$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ . We use logarithmic (top pane) and linear scales (bottom pane). In this example, OBR recovers the high-order target flux.

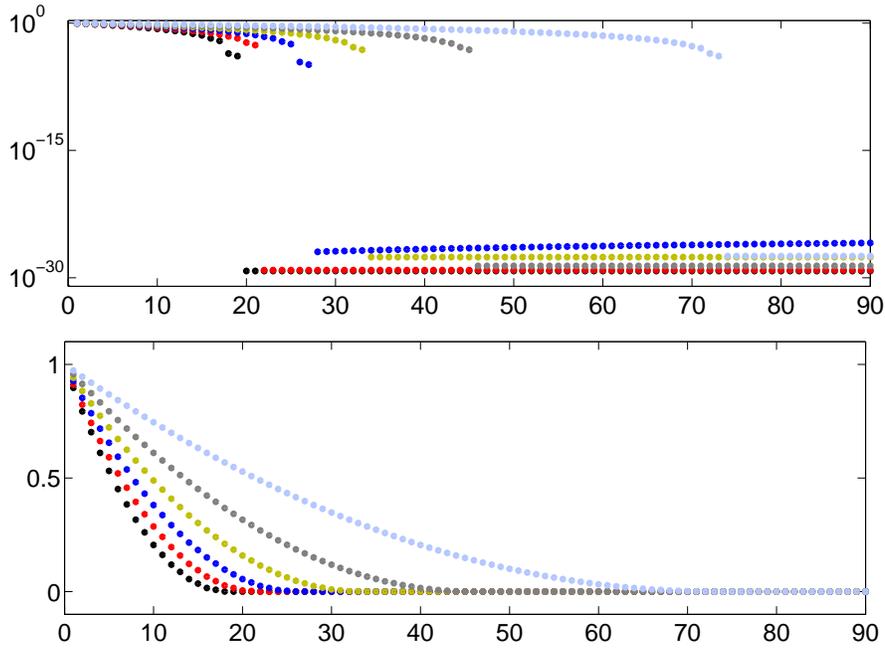
the middle cell of the old mesh. Note that the new mesh satisfies conditions (25)-(26), i.e. is admissible by OBR.

We examine both monotonicity (for OBR, M-OBR (FCR) and the donor-cell method based on swept regions) as well as the preservation of linear densities (for OBR and M-OBR (FCR)). For monotonicity studies, we employ a single remap from the original to the compressed mesh, while for the study of linearity preservation the density is additionally remapped back onto the original mesh. We point out that M-OBR (FCR) and the swept-region donor-cell method use the same computation of low-order fluxes. Monotonicity violations are detected based on the violations of inequality constraints in (20).

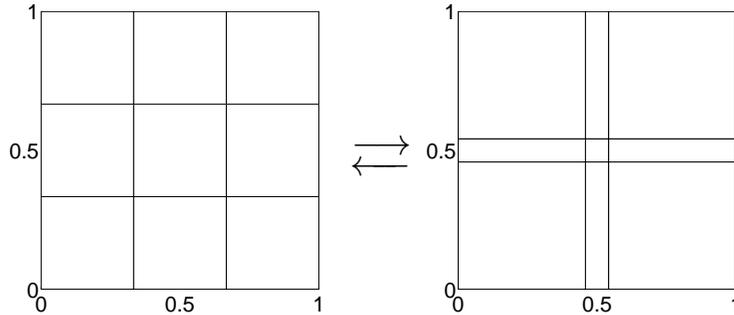
The density function is given by

$$\rho(x, y) = x \quad 0 \leq x, y \leq 1.$$

We study a series of compression increments  $\Delta_1^{x,y}, \Delta_2^{x,y}$  given by



**Fig. 9** Euclidean norm of the difference (vertical axes) between the computed iFCR and OBR fluxes for problems (53) and (54) for increasing numbers of iFCR iterations (horizontal axes). The compression of the middle cell is given by  $\Delta_1 = \Delta_2 \in \{0.1660$  (black),  $0.1661$  (red),  $0.1662$  (blue),  $0.1663$  (green),  $0.1664$  (gray),  $0.1665$  (cyan) $\}$ . The density profile is  $\rho_1 = 80$ ,  $\rho_2 = 82$ ,  $\rho_3 = 0$ , and  $\rho_1^b = \rho_3^b = 0$ . We use logarithmic (top pane) and linear scales (bottom pane). In this example, OBR does *not* recover the high-order target flux, yet the iFCR flux converges to the OBR flux, suggesting that iFCR may be a solution procedure for OBR.



**Fig. 10** A  $3 \times 3$  uniform initial grid (left pane) and the “compressed” grid (right pane) with a  $4 \times 4$ -fold compression of the middle cell.

$$\Delta_1^{x,y} = \Delta_2^{x,y} = \frac{\ell - 1}{6\ell},$$

where  $\ell = \{5, 6, 7, 14, 15, 16, 100\}$  for the monotonicity study and for the linearity preservation study,  $\ell = \{3, 4, 5, 15, 16, 100\}$ , amounting to  $\ell \times \ell$ -fold compressions of the middle cell. An illustration for  $\ell = 4$  is shown in Figure 10.

	$\ell = 5$	$\ell = 6$	$\ell = 7$	$\ell = 14$	$\ell = 15$	$\ell = 16$	$\ell = 100$
OBR	✓	✓	✓	✓	✓	✓	✓
FCR	✓	✓	✓	✓	–	–	–
Donor-cell	✓	–	–	–	–	–	–
iFCR(2)	✓	✓	✓	✓	✓	–	–
iFCR(4)	✓	✓	✓	✓	✓	✓	–
iFCR(721)	✓	✓	✓	✓	✓	✓	✓

**Table 6** Monotonicity of OBR, M-OBR (FCR), the donor-cell method and iFCR, implemented using swept regions, with respect to the remap of a linear density function in two dimensions, for different compression ratios  $\ell \times \ell : 1$  of the middle cell. OBR is bound-preserving throughout, while M-OBR (FCR) and the donor-cell method are not. iFCR is monotone given a sufficient number of iterations. For iFCR( $n$ ) we select the smallest number of iterations  $n$  resulting in a bound-preserving remap, for compression ratios  $\ell \times \ell : 1$ , with  $\ell \in \{15, 16, 100\}$ , respectively.

	$\ell = 3$	$\ell = 4$	$\ell = 5$	$\ell = 15$	$\ell = 16$	$\ell = 100$
OBR	<b>1.36e-16</b>	<b>3.90e-16</b>	<b>2.91e-16</b>	<b>7.99e-16</b>	<b>3.33e-15</b>	<b>2.08e-13</b>
FCR	<b>1.32e-16</b>	4.34e-03	1.06e-02	4.33e-02	4.60e-02	1.97e-01
iFCR(2)	<b>1.36e-16</b>	<b>3.90e-16</b>	6.65e-04	4.07e-02	4.36e-02	1.30e-01
iFCR(20)	<b>1.36e-16</b>	<b>3.90e-16</b>	<b>2.91e-16</b>	<b>7.99e-16</b>	3.00e-03	1.20e-01
iFCR(834)	<b>1.36e-16</b>	<b>3.90e-16</b>	<b>2.91e-16</b>	<b>7.99e-16</b>	<b>3.33e-15</b>	<b>2.08e-13</b>

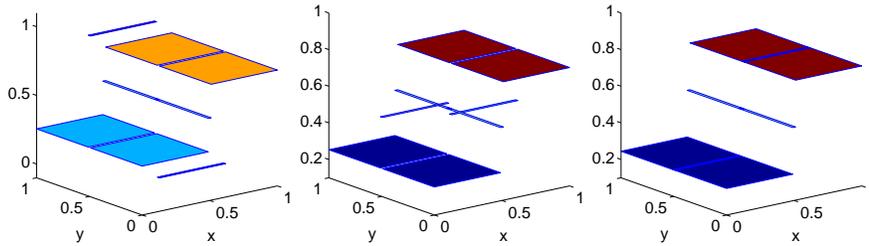
**Table 7**  $L_2$  errors in the OBR, M-OBR (FCR) and iFCR remap of a linear density function in two dimensions, for different compression ratios  $\ell \times \ell : 1$  of the middle cell. Errors close to machine precision are highlighted. OBR preserves linear densities for arbitrarily compressed middle cells, while M-OBR (FCR) does not. For iFCR( $n$ ) we select the smallest number of iterations  $n$  resulting in a linearity-preserving remap, for the compression ratios  $\ell \times \ell : 1$ , with  $\ell \in \{4, 15, 100\}$ , respectively.

Our first observation is that neither the donor-cell method nor M-OBR (FCR) preserve monotonicity for certain mesh motions admissible by OBR, see Table 6. This result is not surprising in view of the condition on mesh motion for the swept-region donor-cell method given in Margolin and Shashkov (2003, p.279), which is violated

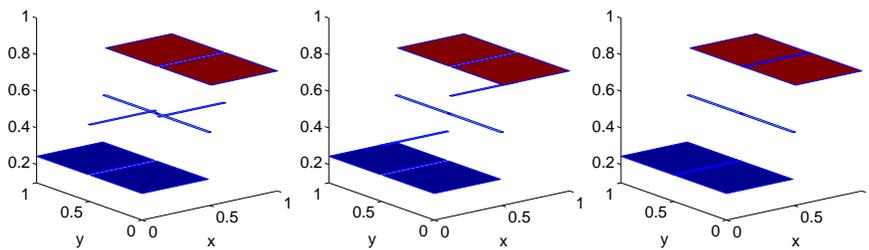
for meshes associated with  $\ell \geq 6$ . Table 6 also reveals that M-OBR (FCR) succeeds in “repairing” the loss of monotonicity inherited from the donor-cell method for  $6 \leq \ell \leq 14$ , but eventually loses monotonicity for  $\ell \geq 15$ .

Table 7 indicates that the loss of linearity preservation in M-OBR (FCR) sets in at fairly low compressions of the middle cell ( $\ell \geq 4$ ) and is therefore not directly related to the loss of monotonicity in the swept-region low-order fluxes, which occurs at  $\ell \geq 6$ . This observation is in agreement with one-dimensional results, where the low-order fluxes are computed based on exact cell intersections and are therefore provably bound-preserving as long as the locality assumption (3) is satisfied, and where M-OBR (FCR) nevertheless fails to preserve linear densities for compressive mesh motions.

In contrast to M-OBR (FCR) and the donor-cell method, OBR is monotonicity and linearity preserving in all our tests. Note also that the iFCR method requires a large number of iterations to recover the OBR solution. The differences between the methods are illustrated for the compression parameter  $\ell = 16$  in Figures 11 and 12.



**Fig. 11** Linear density  $\rho(x,y) = x$  remapped from the uniform  $3 \times 3$  grid to the compressed “torture” grid with  $\ell = 16$ . Left to right: the donor-cell method, M-OBR (FCR), OBR. It is clear that OBR gives the best density approximation.



**Fig. 12** Linear density  $\rho(x,y) = x$  remapped from the uniform  $3 \times 3$  grid to the compressed “torture” grid with  $\ell = 16$ . Left to right: iFCR(2), iFCR(10), iFCR(200). Given a sufficient number of iterations, iFCR recovers the OBR result from Figure 11.

## 8 Computational studies

The purpose of this section is an in-depth comparison of accuracy, robustness and computational cost of OBR and M-OBR (FCR). These attributes are assessed on a series of convergence studies in two dimensions involving (i) smooth cyclic remap on grids with moderate displacements and (ii) cyclic remap on grids with large displacements.

### 8.1 Methodology for the estimation of convergence rates of remap algorithms

The convergence studies in this section are designed to assess the asymptotic accuracy of the OBR and M-OBR (FCR) algorithms in the context of a continuous rezone strategy. In this case, the appropriate notion of remap error and convergence rates can be defined with the help of a *cyclic remap* test as in Margolin and Shashkov (2003). The precise methodology used in the chapter is described below.

A cyclic remap test simulates continuous rezone by performing remap over a parametrized sequence of grids  $K_h^r(\Omega)$ ,  $r = 0, \dots, R$ , such that the following three conditions are satisfied:

- Every  $K_h^r(\Omega)$ ,  $r = 1, \dots, R$ , is topologically equivalent to the initial grid  $K_h^0(\Omega)$ , i.e. all grids in the sequence have the same number of cells and the same connectivity as  $K_h^0(\Omega)$ .
- Any two consecutive grids  $K_h^{r-1}(\Omega)$ ,  $K_h^r(\Omega)$  satisfy the locality assumption (2).
- The first and the last grids coincide, i.e.,  $K_h^0(\Omega) = K_h^R(\Omega)$ .

The integer  $R$  is the number of remap steps. Its reciprocal  $1/R$  can be thought of as a “pseudo-time” step which defines the temporal resolution of the cyclic remap test. The total resolution of the test is specified by the pair  $(K, R)$ , where  $K$  is the number of cells in  $K_h^0(\Omega)$ .

Given a cyclic mesh sequence  $\{K_h^r(\Omega)\}_{r=0}^R$ , called a *cyclic grid*, with total resolution  $(K, R)$ , let  $\vec{\rho}^r \in \mathbb{R}^K$  denote the approximate density solution on  $K_h^r(\Omega)$ , and  $\|\cdot\|$  be a given norm on  $\mathbb{R}^K$ . The remap error on  $\{K_h^r(\Omega)\}_{r=0}^R$  is defined by the norm of the density difference on the first and the last grids in the sequence, i.e.

$$\mathcal{E}(\rho; \|\cdot\|, K, R) = \|\vec{\rho}^0 - \vec{\rho}^R\|. \quad (55)$$

This definition is justified by the fact that  $K_h^0(\Omega) = K_h^R(\Omega)$ , and so the difference between the first and last solutions provides a measure of the total error accrued by the remap algorithm.

To compute the remap error  $\mathcal{E}(\rho; \|\cdot\|, K, R)$  in (55) we use three norms suggested in Margolin and Shashkov (2003). Note that in the case of cyclic remap one does not need to know the exact density distribution to compute the numerical error, which can be instead calculated by comparing the initial and final cell densities. Given an

arbitrary vector  $\vec{\phi} \in \mathbb{R}^K$  these norms are defined as follows:

$$\|\vec{\phi}\|_2 = \left( \sum_{i=1}^K \phi_i^2 V(\kappa_i) \right)^{1/2}, \quad \|\vec{\phi}\|_1 = \sum_{i=1}^K |\phi_i| V(\kappa_i), \quad \|\vec{\phi}\|_\infty = \max_{0 \leq i \leq K} |\phi_i|. \quad (56)$$

If  $\vec{\phi}$  is a piecewise constant approximation of a given scalar function  $\phi(x)$ , then these norms are discrete approximations of the  $L_2$ ,  $L_1$  and  $L_\infty$  norms on  $\Omega$ , respectively.

Once the appropriate notion of remap error is defined, the estimate of convergence rates proceeds in the usual fashion: we compute remap errors using a sequence of cyclic grids with increasing resolution and then estimate the slope of the curve representing the log-log plot of the remap error versus the spatial resolution of the cyclic grid. To this end we use least-squares regression fit. Specifically, for a sequence of cyclic grids with resolutions  $(K^q, R^q)$ ,  $q = 1, \dots, Q$  and the corresponding remap errors  $\mathcal{E}^q = \mathcal{E}(\rho; \|\cdot\|, K^q, R^q)$ , the rate of convergence  $\nu^q$  is estimated by least-squares regression, i.e. by solving the minimization problem

$$\{\nu^q, \omega^q\} = \arg \min \sum_{i=1}^q (\log \mathcal{E}^q + \nu \log R^q - \omega)^2, \quad 1 < q \leq Q. \quad (57)$$

## 8.2 Smooth cyclic remap on grids with moderate displacements

The cyclic grids and the density functions for this study are adopted from Margolin and Shashkov (2003); Liska et al (2010). Specifically, for a given number  $R$  of remap steps and  $r = 0, \dots, R$  the mesh node positions in  $K_h^r(\Omega)$  are given by

$$x_{ij}^r = x(\xi_i, \eta_j, t_r), \quad y_{ij}^r = y(\xi_i, \eta_j, t_r), \quad 0 \leq i \leq N_x, \quad 0 \leq j \leq N_y, \quad (58)$$

where  $N_x$  and  $N_y$  are the numbers of cells in  $x$  and  $y$  direction, respectively,  $x(\xi, \eta, t)$  and  $y(\xi, \eta, t)$  are coordinate maps and

$$\xi_i = \frac{i}{N_x}, \quad i = 0, \dots, N_x; \quad \eta_j = \frac{j}{N_y}, \quad j = 0, \dots, N_y; \quad \text{and} \quad t_r = \frac{r}{R}, \quad r = 0, \dots, R,$$

are the initial (uniform) grid coordinates and the sequence of pseudo-time steps, respectively. We define two sets of coordinate maps. The first set is given by

$$x(\xi, \eta, t) = (1 - \alpha(t))\xi + \alpha(t)\xi^3; \quad (59a)$$

$$y(\xi, \eta, t) = (1 - \alpha(t))\eta + \alpha(t)\eta^2; \quad (59b)$$

$$\alpha(t) = \frac{\sin(4\pi t)}{2}. \quad (59c)$$

It generates a sequence of rectangular, tensor-product (logically Cartesian) grids. The second set is

$$x(\xi, \eta, t) = \xi + \alpha(t) \sin(2\pi\xi) \sin(2\pi\eta); \quad (60a)$$

$$y(\xi, \eta, t) = \eta + \alpha(t) \sin(2\pi\xi) \sin(2\pi\eta); \quad (60b)$$

with

$$\alpha(t) = \begin{cases} t/5 & \text{if } t \leq 5 \\ (1-t)/5 & \text{if } t > 5 \end{cases}.$$

The grids defined by (60) are logically Cartesian but not rectangular. One can show that for any  $0 \leq t \leq 1$  the grids generated by (59) and (60) are valid (see Margolin and Shashkov (2003)).

Convergence rates are estimated as follows. First, we use (58) to define a sequence of  $Q$  cyclic grids where  $Q = 4$ ,  $q = 1, \dots, Q$ , with total resolutions ( $K^q \equiv N_x^q \times N_y^q, R^q$ ) given by  $(64 \times 64, 320)$ ,  $(128 \times 128, 640)$ ,  $(256 \times 256, 1280)$ , and  $(512 \times 512, 2560)$ , respectively. Thus, the total resolution is increased by a factor of  $(2 \times 2, 2)$  in every subsequent set. Then, for every norm in (56) we compute the errors

$$\mathcal{E}^q = \mathcal{E}(\rho; \|\cdot\|, K^q, R^q), \quad q = 1, 2, 3, 4,$$

and solve (57) with  $\{\mathcal{E}^1, \mathcal{E}^2\}$ ,  $\{\mathcal{E}^1, \mathcal{E}^2, \mathcal{E}^3\}$ , and  $\{\mathcal{E}^1, \mathcal{E}^2, \mathcal{E}^3, \mathcal{E}^4\}$ . This approach yields three increasingly accurate estimates of the convergence rates in each norm.

This estimation procedure is applied to three different density functions suggested in Margolin and Shashkov (2003): the ‘‘sine’’

$$\rho(x, y) = 1 + \sin(2\pi x) \sin(2\pi y), \quad (61)$$

the ‘‘peak’’

$$\rho(x, y) = \begin{cases} 1, & r > 0.25 \\ \max\{1.001, 4(r - 0.25) + 1\}, & r \leq 0.25 \end{cases}, \quad (62a)$$

$$r = \sqrt{(x - 0.5)^2 + (y - 0.5)^2}, \quad (62b)$$

and the ‘‘shock’’

$$\rho(x, y) = \begin{cases} 2, & y \geq (x - 0.4)/0.3 \\ 1, & y \leq (x - 0.4)/0.3 \end{cases}. \quad (63)$$

Errors of the OBR and M-OBR (FCR) algorithms and the corresponding convergence rates are presented in Tables 8–10. We observe that for the peak and shock densities the OBR and M-OBR (FCR) convergence rates are virtually identical, whereas for the sine density the  $L_2$  and  $L_\infty$  rates of OBR are better by 0.2. Intuitively this can be explained by noting that the peak and shock examples are comprised of piecewise linear functions for which the global optimization problem likely decouples into local optimization problems around the discontinuities. This diminishes the distinction between global (OBR) and local (M-OBR) optimization formulations of remap. In contrast, for the sine density, which is a globally smooth function, the

OBR							
#cells	#remaps	$L_2$ err	$L_1$ err	$L_\infty$ err	$L_2$ rate	$L_1$ rate	$L_\infty$ rate
$64 \times 64$	320	6.58e-04	4.91e-04	5.78e-03	—	—	—
$128 \times 128$	640	8.88e-05	6.16e-05	1.64e-03	2.89	3.00	1.82
$256 \times 256$	1280	1.21e-05	7.82e-06	4.65e-04	2.88	2.99	1.82
$512 \times 512$	2560	1.70e-06	9.89e-07	1.39e-04	2.87	2.98	1.80
FCR							
#cells	#remaps	$L_2$ err	$L_1$ err	$L_\infty$ err	$L_2$ rate	$L_1$ rate	$L_\infty$ rate
$64 \times 64$	320	7.78e-04	4.95e-04	8.75e-03	—	—	—
$128 \times 128$	640	1.22e-04	6.49e-05	2.81e-03	2.67	2.93	1.64
$256 \times 256$	1280	2.00e-05	8.49e-06	8.89e-04	2.64	2.93	1.65
$512 \times 512$	2560	3.43e-06	1.08e-06	2.84e-04	2.61	2.95	1.65

**Table 8** OBR and M-OBR (FCR) errors and convergence rate estimates for the “sine” density (61) using 4 tensor-product cyclic grids defined by (59). The  $L_2$  and  $L_\infty$  rates for OBR are slightly better than those for M-OBR (FCR). Additionally, we observe superconvergence for both methods in  $L_2$  and  $L_1$  norms.

OBR							
#cells	#remaps	$L_2$ err	$L_1$ err	$L_\infty$ err	$L_2$ rate	$L_1$ rate	$L_\infty$ rate
$64 \times 64$	320	6.97e-03	2.55e-03	8.00e-02	—	—	—
$128 \times 128$	640	3.09e-03	8.90e-04	5.06e-02	1.17	1.52	0.66
$256 \times 256$	1280	1.40e-03	3.10e-04	3.16e-02	1.16	1.52	0.67
$512 \times 512$	2560	6.40e-04	1.09e-04	1.96e-02	1.15	1.52	0.68
FCR							
#cells	#remaps	$L_2$ err	$L_1$ err	$L_\infty$ err	$L_2$ rate	$L_1$ rate	$L_\infty$ rate
$64 \times 64$	320	5.98e-03	2.14e-03	8.33e-02	—	—	—
$128 \times 128$	640	2.54e-03	7.30e-04	5.29e-02	1.24	1.55	0.66
$256 \times 256$	1280	1.11e-03	2.50e-04	3.33e-02	1.22	1.55	0.66
$512 \times 512$	2560	4.98e-04	8.71e-05	2.07e-02	1.20	1.54	0.67

**Table 9** OBR and M-OBR (FCR) errors and convergence rate estimates for the “peak” density (62a) using 4 tensor-product cyclic grids defined by (59). For this classical example, the convergence rates of OBR and M-OBR (FCR) are virtually identical.

feasible set of the global optimization problem remains fully coupled. In addition, we note that the  $L_2$  and  $L_1$  results in Table 8 are subject to superconvergence due to the choice of the grids.

Overall, these results indicate that OBR and M-OBR (FCR) have approximately the same accuracy on classical test problems. Consequently, one may wonder if the effects of the toy examples of Section 7 are never encountered in practice. In the next section we confirm, however, that important differences exist not only on toy problems; in particular, we demonstrate that OBR is more accurate and more robust than M-OBR (FCR) on grids that are of significant practical merit.

OBR							
#cells	#remaps	$L_2$ err	$L_1$ err	$L_\infty$ err	$L_2$ rate	$L_1$ rate	$L_\infty$ rate
$64 \times 64$	320	9.12e-02	2.88e-02	4.72e-01	—	—	—
$128 \times 128$	640	7.12e-02	1.75e-02	4.86e-01	0.36	0.72	-0.04
$256 \times 256$	1280	5.57e-02	1.06e-02	4.87e-01	0.36	0.72	-0.02
$512 \times 512$	2560	4.33e-02	6.35e-03	4.98e-01	0.36	0.73	-0.02
FCR							
#cells	#remaps	$L_2$ err	$L_1$ err	$L_\infty$ err	$L_2$ rate	$L_1$ rate	$L_\infty$ rate
$64 \times 64$	320	8.43e-02	2.45e-02	4.67e-01	—	—	—
$128 \times 128$	640	6.57e-02	1.47e-02	4.77e-01	0.36	0.73	-0.03
$256 \times 256$	1280	5.12e-02	8.87e-03	4.77e-01	0.36	0.73	-0.02
$512 \times 512$	2560	3.99e-02	5.34e-03	4.88e-01	0.36	0.73	-0.02

**Table 10** OBR and M-OBR (FCR) errors and convergence rate estimates for the “shock” density (63) using 4 tensor-product cyclic grids defined by (59). For this classical example, the convergence rates of OBR and M-OBR (FCR) are virtually identical.

### 8.3 Cyclic remap on grids with large displacements

Theorem 3 asserts that the feasible set of M-OBR (FCR) is always a subset of the feasible set of the OBR formulation. This suggests that (37) may be more accurate than (43). Examples in this section show that this is indeed the case and that the smaller feasible set of (43) can impact adversely the accuracy and, more importantly, robustness of M-OBR (FCR).

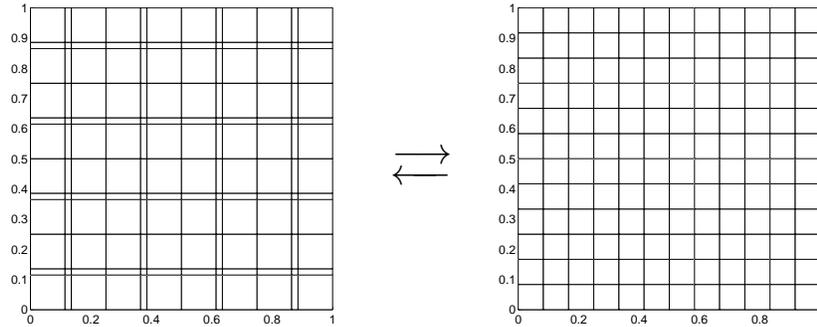
We begin with a study of accuracy. To this end, we compare convergence rates of the OBR and M-OBR (FCR) algorithms for the sine density (61) on a sequence of cyclic grids resulting from compressing every third cell equally in  $x$  and  $y$  direction, followed by a relaxation into a fully uniform grid. This mesh motion is motivated by the examples of Section 7 and is intended to mimic the effects of a repeated mesh repair procedure, see Figure 13.

The ‘repeated-repair’ cyclic grid is given by

$$x_{ij}^r = \begin{cases} x_{ij}^0 & \text{if } r \text{ is even, for all } i, j, \text{ otherwise (when } r \text{ is odd):} \\ x_{ij}^0 & \text{if } i \equiv 0 \pmod{3}, \text{ or if } i = N_x, \\ x_{ij}^0 + \Delta_x & \text{if } i \equiv 1 \pmod{3}, \text{ for } i < N_x, \\ x_{ij}^0 - \Delta_x & \text{if } i \equiv 2 \pmod{3}, \text{ for } i < N_x \end{cases} \quad (64)$$

and

$$y_{ij}^r = \begin{cases} y_{ij}^0 & \text{if } r \text{ is even, for all } i, j, \text{ otherwise (when } r \text{ is odd):} \\ y_{ij}^0 & \text{if } j \equiv 0 \pmod{3}, \text{ or if } j = N_y, \\ y_{ij}^0 + \Delta_y & \text{if } j \equiv 1 \pmod{3}, \text{ for } j < N_y, \\ y_{ij}^0 - \Delta_y & \text{if } j \equiv 2 \pmod{3}, \text{ for } j < N_y. \end{cases} \quad (65)$$



**Fig. 13** Grid deformation due to local compression (left pane) and the ‘repaired’ uniform grid (right pane), see (64)-(65).

The initial grid  $K_h^0$  is a uniform grid on the unit square  $[0, 1] \times [0, 1]$ . We set

$$\Delta_x = \Delta_y = \frac{4}{5} (x_{10}^0 - x_{00}^0),$$

resulting in a constant compression ratio of  $4 \times 4 : 1$  for every third grid cell in  $x$  and  $y$  direction, whenever  $r$  is odd. For even  $r$  the grid is relaxed to its original position. See Figure 13.

Estimates of the convergence rates of OBR and M-OBR (FCR) are presented in Table 11. The first observation is that the accuracy of the OBR algorithm on the repeated-repair cyclic grid is immune to the underlying mesh motion. In particular, the convergence rates of OBR in all three norms equals the best possible theoretical rates for a linearity-preserving scheme.

In contrast, it is clear that the convergence rates of M-OBR (FCR) suffer on the repeated-repair cyclic grid. The estimates in all three norms show a consistent trend toward a first-order scheme. We note that this is not due to a potential loss of monotonicity in low-order (donor-cell) fluxes; the compression parameters have been chosen such that the monotonicity of low-order fluxes is preserved. In other words, the loss of accuracy is purely due to a smaller feasible set employed by M-OBR (FCR). On the other hand, we observe that iFCR recovers the result of OBR at the expense of only 2 flux iterations per remap.

Our second study examines the robustness of OBR and M-OBR (FCR). To this end, we investigate the behavior of the methods on  $64 \times 64$  meshes when the pseudo-time step  $1/R$  is decreased significantly beyond the previously used test value of  $1/320$ . Table 12 displays the  $L_1$  error in remapping the linear density  $\rho(x, y) = x$  on the tensor product cyclic grid (59) for varying pseudo-time steps. In the test, we choose to declare loss of linearity preservation when the  $L_1$  error exceeds  $1e-8$ . We note that it is expected that both OBR and M-OBR (FCR) will eventually fail to preserve linear densities due to the restrictions on admissible mesh motions,

OBR							
#cells	#remaps	$L_1$ err	$L_2$ err	$L_\infty$ err	$L_1$ rate	$L_2$ rate	$L_\infty$ rate
128×128	640	2.69e-04	3.65e-04	2.03e-03	—	—	—
256×256	1280	6.71e-05	9.08e-05	5.07e-04	<b>2.00</b>	<b>2.01</b>	<b>2.00</b>
512×512	2560	1.68e-05	2.27e-05	1.20e-04	<b>2.00</b>	<b>2.00</b>	<b>2.04</b>
1024×1024	5120	4.19e-06	5.66e-06	2.69e-05	<b>2.00</b>	<b>2.00</b>	<b>2.08</b>
FCR							
128×128	640	2.81e-04	3.47e-04	1.23e-03	—	—	—
256×256	1280	9.23e-05	1.19e-04	5.14e-04	1.61	1.54	1.26
512×512	2560	3.65e-05	5.05e-05	2.50e-04	1.47	1.39	1.15
1024×1024	5120	1.69e-05	2.39e-05	1.24e-04	1.35	1.28	1.10
iFCR(2)							
128×128	640	2.69e-04	3.64e-04	1.57e-03	—	—	—
256×256	1280	6.71e-05	9.07e-05	3.95e-04	<b>2.00</b>	<b>2.01</b>	<b>1.99</b>
512×512	2560	1.68e-05	2.27e-05	9.88e-05	<b>2.00</b>	<b>2.00</b>	<b>2.00</b>
1024×1024	5120	4.19e-06	5.66e-06	2.47e-05	<b>2.00</b>	<b>2.00</b>	<b>2.00</b>

**Table 11** OBR, M-OBR (FCR) and iFCR(2) errors and convergence rate estimates for the sine density (61) using 4 cyclic repeated-repair grids defined by (64)-(65). Rates expected of a second-order scheme are highlighted. It is evident that OBR delivers second-order accuracy, while M-OBR (FCR) exhibits a trend toward a first-order scheme. iFCR(2) gives  $L_1$  and  $L_2$  errors and convergence rates that are nearly identical to those given by OBR.

introduced earlier. OBR fails to preserve linear densities at  $R = 154$ , while M-OBR (FCR) fails at  $R = 212$ . Therefore, for this particular grid, the admissible pseudo-time step for OBR is approximately 1.4 times larger than that for M-OBR (FCR). Additionally, we observe that while OBR exhibits a graceful loss of accuracy once the OBR mesh motion conditions (25)-(26) are violated, M-OBR (FCR) becomes numerically unstable. This is most likely due to the loss of monotonicity in M-OBR (FCR) discussed and demonstrated in Sections 5 and 7.3, respectively. We also note that iFCR is more robust than FCR, however it does not duplicate the robustness of OBR.

Similarly, Table 13 displays the  $L_1$  error in remapping the linear density  $\rho(x, y) = x$  on the smooth nonorthogonal cyclic grid (60). In this case OBR fails to preserve linear densities at  $R = 15$ , while M-OBR (FCR) fails at  $R = 24$ . Therefore, for this particular grid, the admissible pseudo-time step for OBR is approximately 1.6 times larger than that for M-OBR (FCR). Additionally, we observe that iFCR is more robust and more accurate than FCR, however the  $L_1$  remap error does not converge to that of OBR as the number of iterations increases.

	$R = 213$	$R = 212$	$R = 211$	$R = 155$	$R = 154$	$R = 153$	$R = 100$	$R = 50$
OBR	<b>1.32e-13</b>	<b>1.42e-13</b>	<b>1.60e-13</b>	<b>4.60e-09</b>	4.06e-06	1.53e-05	1.97e-03	6.48e-03
FCR	<b>1.32e-13</b>	5.32e-08	1.10e-06	2.26e-03	2.35e-03	2.44e-03	5.73e+04	8.50e+11
iFCR <sup>1</sup>	<b>1.32e-13</b>	<b>1.42e-13</b>	<b>1.60e-13</b>	1.36e-03	1.64e-03	1.39e-03	1.72e+02	1.29e+09
iFCR <sup>2</sup>	<b>1.32e-13</b>	<b>1.42e-13</b>	<b>1.60e-13</b>	2.29e-03	1.14e-03	1.17e-03	4.61e+01	3.92e+08
iFCR <sup>3</sup>	<b>1.32e-13</b>	<b>1.42e-13</b>	<b>1.60e-13</b>	<b>4.60e-09</b>	4.01e-05	1.32e-04	2.90e+01	1.32e+10
iFCR <sup>4</sup>	<b>1.32e-13</b>	<b>1.42e-13</b>	<b>1.60e-13</b>	<b>4.60e-09</b>	4.01e-05	1.32e-04	2.64e+01	2.29e+08

**Table 12**  $L_1$  errors in the OBR, M-OBR (FCR) and iFCR remap of a linear density function on the  $64 \times 64$  tensor-product grid, for different values of the pseudo-time step  $1/R$ . Here  $\text{iFCR}^1 = \text{iFCR}(2)$ ,  $\text{iFCR}^2 = \text{iFCR}(20)$ ,  $\text{iFCR}^3 = \text{iFCR}(200)$  and  $\text{iFCR}^4 = \text{iFCR}(1000)$ . Errors smaller than  $1e-8$  are highlighted. OBR fails to preserve linear densities at  $R = 154$ , while M-OBR (FCR) fails at  $R = 212$ , resulting in a pseudo-time step advantage for OBR of  $212/154 \approx 1.4$ . Beyond this point, OBR exhibits a graceful loss of accuracy; M-OBR (FCR) becomes numerically unstable. iFCR is more robust than FCR, however it does not duplicate the robustness of OBR.

	$R = 25$	$R = 24$	$R = 23$	$R = 16$	$R = 15$	$R = 14$	$R = 10$	$R = 5$
OBR	<b>2.32e-14</b>	<b>4.49e-14</b>	<b>2.15e-13</b>	<b>4.52e-10</b>	4.14e-05	5.13e-04	1.16e-03	2.45e-03
FCR	<b>2.32e-14</b>	3.63e-07	1.67e-06	8.60e-04	1.16e-03	1.69e-03	5.74e-03	1.09e-02
iFCR <sup>1</sup>	<b>2.32e-14</b>	<b>4.49e-14</b>	<b>2.15e-13</b>	1.52e-03	3.13e-03	4.95e-03	8.08e-03	1.38e-02
iFCR <sup>2</sup>	<b>2.32e-14</b>	<b>4.49e-14</b>	<b>2.15e-13</b>	<b>4.52e-10</b>	7.48e-05	6.89e-04	3.03e-02	7.89e-02
iFCR <sup>3</sup>	<b>2.32e-14</b>	<b>4.49e-14</b>	<b>2.15e-13</b>	<b>4.52e-10</b>	7.48e-05	6.89e-04	1.93e-02	3.44e-02
iFCR <sup>4</sup>	<b>2.32e-14</b>	<b>4.49e-14</b>	<b>2.15e-13</b>	<b>4.52e-10</b>	7.48e-05	6.89e-04	1.93e-02	3.39e-02

**Table 13**  $L_1$  errors in the OBR and M-OBR (FCR) remap of a linear density function on the  $64 \times 64$  smooth nonorthogonal grid, for different values of the pseudo-time step  $1/R$ . Here  $\text{iFCR}^1 = \text{iFCR}(2)$ ,  $\text{iFCR}^2 = \text{iFCR}(20)$ ,  $\text{iFCR}^3 = \text{iFCR}(200)$  and  $\text{iFCR}^4 = \text{iFCR}(1000)$ . Errors smaller than  $1e-8$  are highlighted. OBR fails to preserve linear densities at  $R = 15$ , while M-OBR (FCR) fails at  $R = 24$ , resulting in a pseudo-time step advantage for OBR of  $24/15 \approx 1.6$ . iFCR is more robust than FCR, however the  $L_1$  error does not converge to that of OBR as the number of iterations increases.

## 8.4 Computational cost

From Theorem 4 we know that (43) decouples into a set of independent single-variable inequality-constrained optimization problems whose solution is given by (45). In other words, the computational cost of M-OBR (FCR) is quite low. On the other hand, the OBR formulation is a globally coupled inequality-constrained optimization problem. It is therefore of considerable practical interest to assess the performance penalty incurred by the need to solve a global optimization problem.

The algorithms used to solve M-OBR (FCR) and OBR formulations are described in Section 5. Table 14 presents preliminary timing results using Matlab™ implementations of M-OBR (FCR), iFCR, and OBR. For accurate estimates of the computational cost we choose the examples of Section 8.2. We make two observations.

First, while a direct comparison of our implementation of M-OBR (FCR) and the closely related SFCR method implemented in Fortran, see Liska et al (2010, p.1490), is not possible, we note that the computational cost of our Matlab™ implementation is in the range of the computational cost of the Fortran implementation. We achieve this by employing only vectorized Matlab™ operations, which are delegated to fast computational kernels. The linear (ten-fold) scaling of the mesh-to-mesh computational cost reported in Liska et al (2010, p.1490) is evident in our case when meshes are sufficiently large, i.e. when the computational overhead associated with the Matlab™ environment becomes negligible.

Second, noting that additional studies with more efficient implementations of M-OBR (FCR) and, especially, OBR are needed, we can already see that the computational cost of OBR is proportional, up to a very modest constant, to the cost of M-OBR (FCR). On average, OBR is only 2.1 times slower than M-OBR (FCR). Considering the gains in accuracy and robustness as well as the less restrictive conditions on admissible mesh motions, OBR is a strong alternative to M-OBR (FCR). Finally, for iFCR we observe that approximately 20 flux iterations per remap can be employed at the cost of OBR.

Sine								
cells	remaps	OBR	FCR	ratio	iFCR(2)	ratio	iFCR(20)	ratio
64×64	320	<b>7.3</b>	4.2	<b>1.7</b>	4.4	<b>1.7</b>	7.4	<b>1.0</b>
128×128	640	<b>49.5</b>	25.4	<b>1.9</b>	27.6	<b>1.8</b>	50.5	<b>1.0</b>
256×256	1280	<b>390.6</b>	176.5	<b>2.2</b>	198.9	<b>2.0</b>	387.8	<b>1.0</b>
512×512	2560	<b>3662.8</b>	1812.5	<b>2.0</b>	2156.6	<b>1.7</b>	4955.4	<b>0.7</b>
Peak								
64×64	320	<b>8.4</b>	4.9	<b>1.7</b>	5.1	<b>1.6</b>	8.7	<b>1.0</b>
128×128	640	<b>57.8</b>	28.5	<b>2.0</b>	31.0	<b>1.9</b>	55.7	<b>1.0</b>
256×256	1280	<b>418.6</b>	183.8	<b>2.3</b>	203.2	<b>2.1</b>	448.7	<b>0.9</b>
512×512	2560	<b>4528.6</b>	1832.9	<b>2.5</b>	2264.0	<b>2.0</b>	5156.8	<b>0.9</b>
Shock								
64×64	320	<b>9.8</b>	4.9	<b>2.0</b>	4.9	<b>2.0</b>	8.2	<b>1.2</b>
128×128	640	<b>88.9</b>	28.1	<b>3.2</b>	31.1	<b>2.9</b>	54.1	<b>1.6</b>
256×256	1280	<b>438.6</b>	184.7	<b>2.4</b>	220.4	<b>2.0</b>	409.0	<b>1.1</b>
512×512	2560	<b>3214.6</b>	1794.1	<b>1.8</b>	2237.4	<b>1.4</b>	4806.3	<b>0.7</b>

**Table 14** Comparison of the computational costs (times are in seconds) of OBR, M-OBR (FCR), iFCR(2) and iFCR(20) as measured by Matlab™ wall-clock times on a single Intel Xeon X5680 3.33GHz processor, for density functions defined in (61), (62a) and (63) and the cyclic grid (59). Ratios of OBR times and FCR/iFCR(2)/iFCR(20) times are also reported. The cost of OBR is proportional, up to a modest constant, to the cost of M-OBR (FCR). The average cost ratio is only **2.1**. The OBR to iFCR(2) cost ratio is **1.9**. The OBR to iFCR(20) cost ratio is **1.0**.

## 9 Conclusions

In this chapter we formulate and study a new class of optimization-based, conservative, bound and linearity preserving remap algorithms (OBR). The use of an optimization setting allows us to separate accuracy considerations from the enforcement of physical bounds by making the former the objective of optimization, while the latter is used to define the constraints in the optimization problem. In so doing we obtain a scheme that is provably linearity preserving and bound-preserving on arbitrary unstructured grids, including grids with non-convex polygonal or polyhedral cells.

Rigorous characterization of the relationship between the OBR and the FCR algorithm of Liska et al (2010) is another key contribution of this chapter. Specifically, we prove that the FCR is equivalent to an inequality-constrained optimization problem, termed M-OBR, which is derived from OBR by replacing its constraints by a set of simpler sufficient conditions for the local bounds. These conditions are *decoupled* box constraints derived using a worst-case local analysis to simplify the original *coupled* inequality constraints. Using the relationship between the constraints in OBR and M-OBR (FCR) we prove that the feasible set of M-OBR (FCR) is always contained in the feasible set of OBR. It follows that asymptotically OBR is at least as accurate as M-OBR (FCR). Furthermore, numerical comparison between OBR and the *iterated* FCR (iFCR) strongly suggest that the latter provides an iterative solution algorithm for the global optimization problem in the OBR formulation.

Succinctly, our theoretical and computational results establish the following hierarchy among the OBR, FCR and iFCR methods:

- OBR defines a “*master*” optimization formulation for the remap problem, characterized by a global set of linear constraints derived from physical considerations;
- FCR simplifies the master optimization problem by *decoupling* the linear constraints, which reduces the size of the feasible set;
- iFCR is an *iterative procedure* that under some conditions may recover the OBR solution.

Because FCR is motivated by flux-corrected transport (FCT), this hierarchy opens up an interesting possibility that FCT and *iterative* FCT, see Kuzmin et al (2005), may also be connected to a “*master*” global optimization formulation for the selection of accurate and monotone fluxes in transport algorithms.

The computational examples in this chapter provide further illustration of the hierarchy among these methods. For smooth cyclic grids with moderate displacements there are no significant differences in the accuracy and the convergence rates of M-OBR (FCR) and OBR. However, on cyclic grids with large displacements the smaller feasible set of M-OBR (FCR) can adversely impact its accuracy and robustness. In particular, we demonstrate that on such grids M-OBR (FCR) defaults to a first-order accurate scheme, while OBR achieves the theoretically best possible accuracy (second order) for a linearity-preserving scheme. Furthermore, in a series of large-displacement examples we show that the OBR formulation admits a larger pseudo-time step (1.4 to 1.6 times) and that M-OBR (FCR) can suffer nu-

merical breakdown due to the loss of monotonicity in low-order fluxes based on swept-region computations. In contrast, OBR does not require the computation of low-order fluxes; at the same time, the employed computation of high-order fluxes using swept regions is safe because monotonicity is enforced separately, through inequality constraints. Finally, a “torture” test reveals that under certain conditions the smaller feasible set of M-OBR (FCR) can lead to the loss of qualitative information about the shape of the remapped density function.

Preliminary studies show that for a set of standard remap test problems the cost of OBR is proportional, up to a very modest constant, to the cost of M-OBR (FCR). On average, not counting potential gains from the time-step advantage of OBR, it is only about twice as expensive as FCR. This suggests that OBR can be competitive in practical applications where a (i) provably linearity-preserving (and otherwise *optimally* accurate) and (ii) bound-preserving method is desired.

The extension of the OBR approach to systems, and further theoretical and computational studies, including formal analysis of iFCR as an iterative solution algorithm for OBR will be the subject of a forthcoming paper.

## Acknowledgments

All authors acknowledge funding by the DOE Office of Science Advanced Scientific Computing Research (ASCR) Program. PB, DR and GS also acknowledge funding by the NNSA Climate Modeling and Carbon Measurement Project. DR and MS also acknowledge funding by the Advanced Simulation & Computing (ASC) Program.

Our colleagues Dmitri Kuzmin, Richard Liska, Kara Peterson, John Shadid, Pavel Váchal and Joseph Young provided many comments and valuable insights that helped improve this work.

## References

- Bell J, Berger M, Saltzman J, Welcome M (1994) Three-dimensional adaptive mesh refinement for hyperbolic conservation laws. *SIAM J Sci Comput* 15(1):127–138
- Berger M, Murman SM, Aftosmis MJ (2005) Analysis of slope limiters on irregular grids. In: Proceedings of the 43rd AIAA Aerospace Sciences Meeting, AIAA, Reno, NV, AIAA2005-0490
- Berger MJ, Colella P (1989) Local adaptive mesh refinement for shock hydrodynamics. *J Comput Phys* 82:64–84
- Bochev P, Day D (2008) Analysis and computation of a least-squares method for consistent mesh tying. *J Comp Appl Math* 218:21–33
- Bochev P, Ridzal D, Scovazzi G, Shashkov M (2011) Formulation, analysis and numerical study of an optimization-based conservative interpolation (remap) of

- scalar fields for arbitrary Lagrangian-Eulerian methods. *Journal of Computational Physics* 230(12):5199–5225
- Carey G, Bicken G, Carey V, Berger C, Sanchez J (2001) Locally constrained projections on grids. *Int J Num Meth Engrg* 50:549–577
- Coleman TF, Li Y (1996) A reflective newton method for minimizing a quadratic function subject to bounds on some of the variables. *SIAM Journal on Optimization* 6(4):1040–1058
- Dukowicz JK, Kodis JW (1987) Accurate conservative remapping (rezoning) for arbitrary Lagrangian-Eulerian computations. *SIAM J Sci Stat Comput* 8:305–321
- Hirt C, Amsden A, Cook J (1974) An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *Journal of Computational Physics* 14:227–253
- Jones PW (1999) First- and second-order conservative remapping schemes for grids in spherical coordinates. *Monthly Weather Review* 127(9):2204–2210
- Kucharik M, Shashkov M, Wendroff B (2003) An efficient linearity-and-bound-preserving remapping method. *Journal of Computational Physics* 188(2):462 – 471
- Kuzmin D, Löhner R, Turek S (eds) (2005) *Flux-Corrected Transport. Principles, Algorithms and Applications*. Springer Verlag, Berlin, Heidelberg
- Laursen T, Heinstejn M (2003) A three dimensional surface-to-surface projection algorithm for non-coincident domains. *Comm Numer Meth Eng* 19:421–432
- Liska R, Shashkov M, Váchal P, Wendroff B (2010) Optimization-based synchronized flux-corrected conservative interpolation (remapping) of mass and momentum for arbitrary lagrangian-eulerian methods. *Journal of Computational Physics* 229(5):1467–1497
- Loubere R, Shashkov MJ (2005) A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian-Eulerian methods. *Journal of Computational Physics* 209(1):105 – 138
- Loubere R, Staley M, Wendroff B (2006) The repair paradigm: New algorithms and applications to compressible flow. *Journal of Computational Physics* 211(2):385 – 404
- Margolin LG, Shashkov M (2003) Second-order sign-preserving conservative interpolation (remapping) on general grids. *J Comput Phys* 184(1):266–298
- Margolin LG, Shashkov M (2004) Remapping, recovery and repair on a staggered grid. *Computer Methods in Applied Mechanics and Engineering* 193(39-41):4139 – 4155
- Miller DS, Burton DE, Oliviera JS (1996) Efficient second order remapping on arbitrary two dimensional meshes. Technical Report UCRL-ID-123530, Lawrence Livermore National Laboratory
- Rider WJ, Kothe DB (1997) Constrained minimization for monotonic reconstruction. In: *Proceedings of the 13th AIAA Computational Fluid Dynamics Conference, AIAA, Snowmass, Colorado, 97-2036*, pp 955–964
- Rockafellar RT (1970) *Convex Analysis*. Princeton University Press, Princeton, N.J.
- Schär C, Smolarkiewicz PK (1996) A synchronous and iterative flux-correction formalism for coupled transport equations. *Journal of Computational Physics* 128(1):101 – 120

Constrained-Optimization Based Data Transfer: A New Perspective on Flux Correction. 55

Swartz B (1999) Good neighborhoods for multidimensional Van Leer limiting. *Journal of Computational Physics* 154(1):237 – 241