Fast algorithms for optimization based remap and transport with feature preservation

SAND2013-5446C

Denis Ridzal Pavel Bochev Kara Peterson

Sandia National Laboratories

in collaboration with

Mikhail Shashkov (LANL)

Large-Scale Scientific Computations, Sozopol
June 4, 2013

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.

The remap problem

Optimization-based remap (OBR)

Optimization algorithms

Remap results

Adaptable targets

Transport results

The remap problem

Optimization-based remap (OBR)

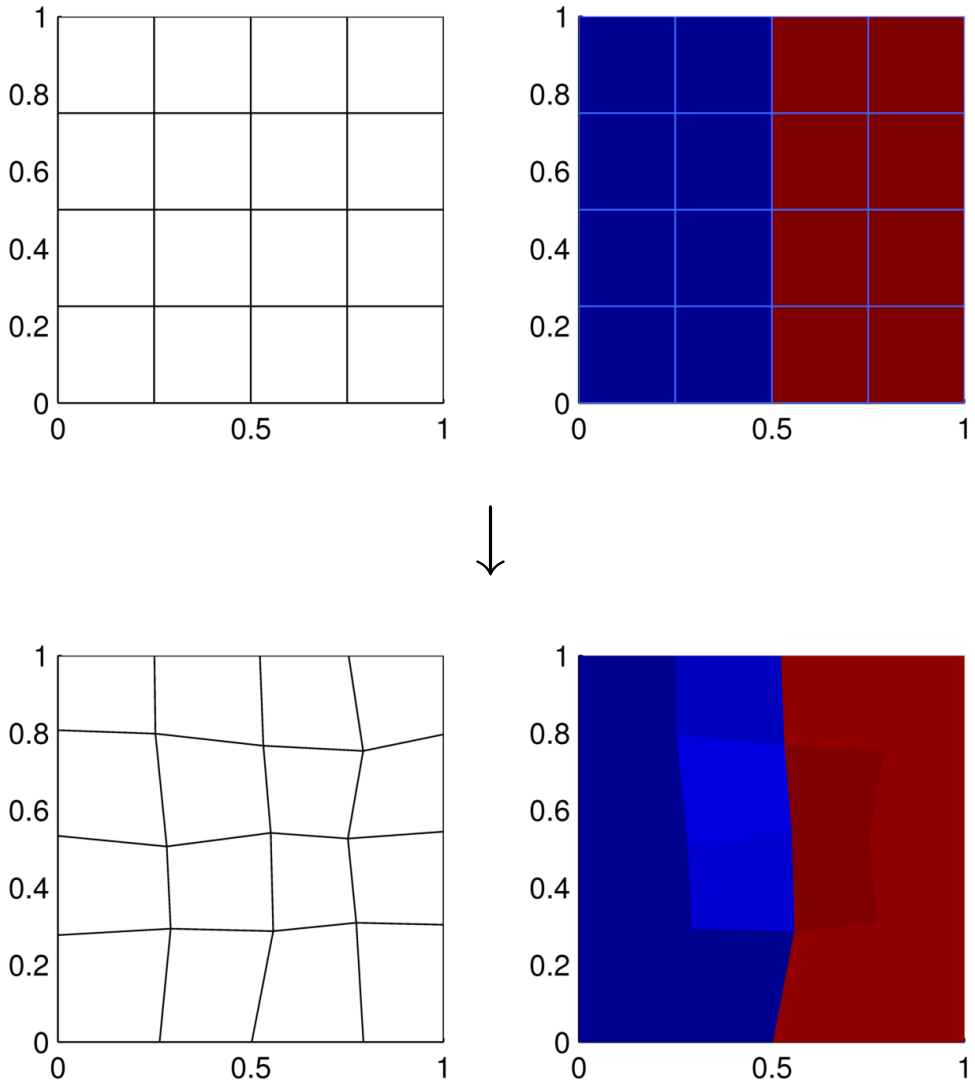
Optimization algorithms

Remap results

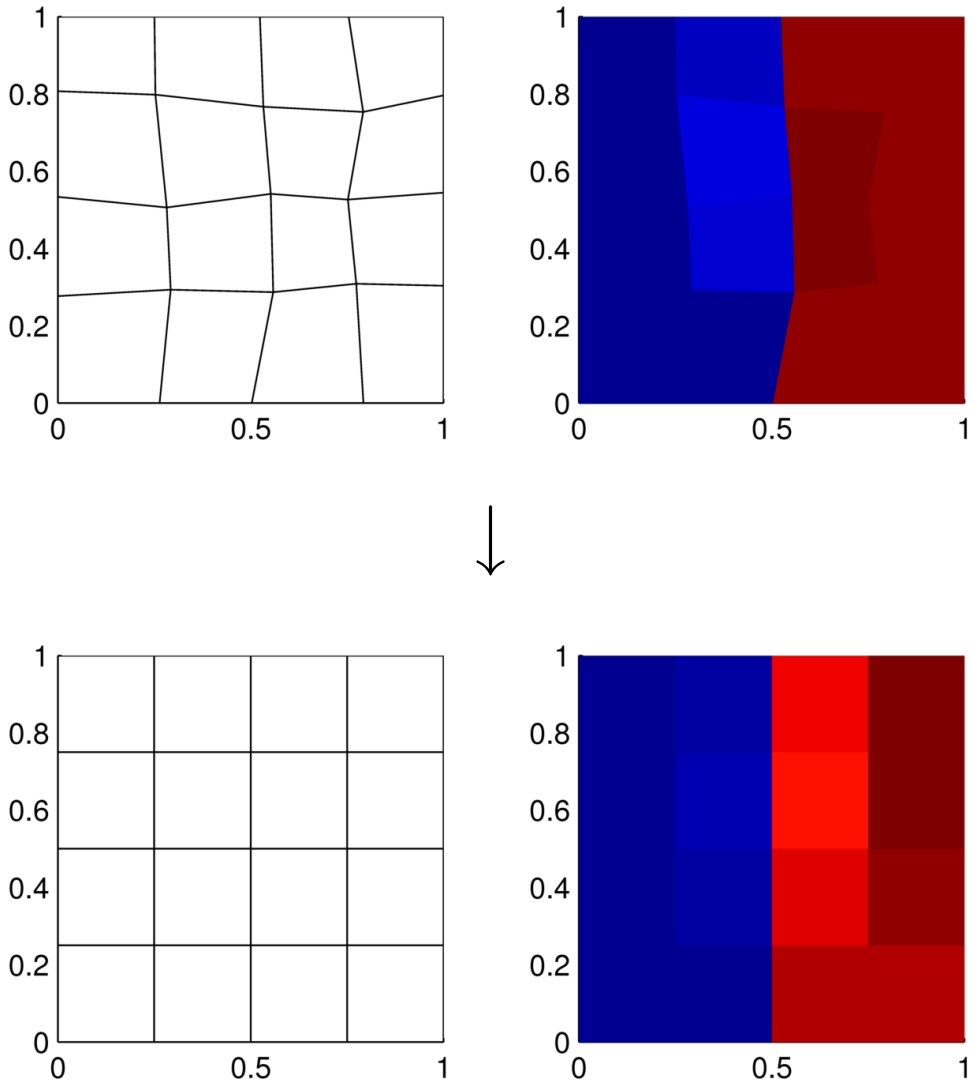
Adaptable targets

Transport results

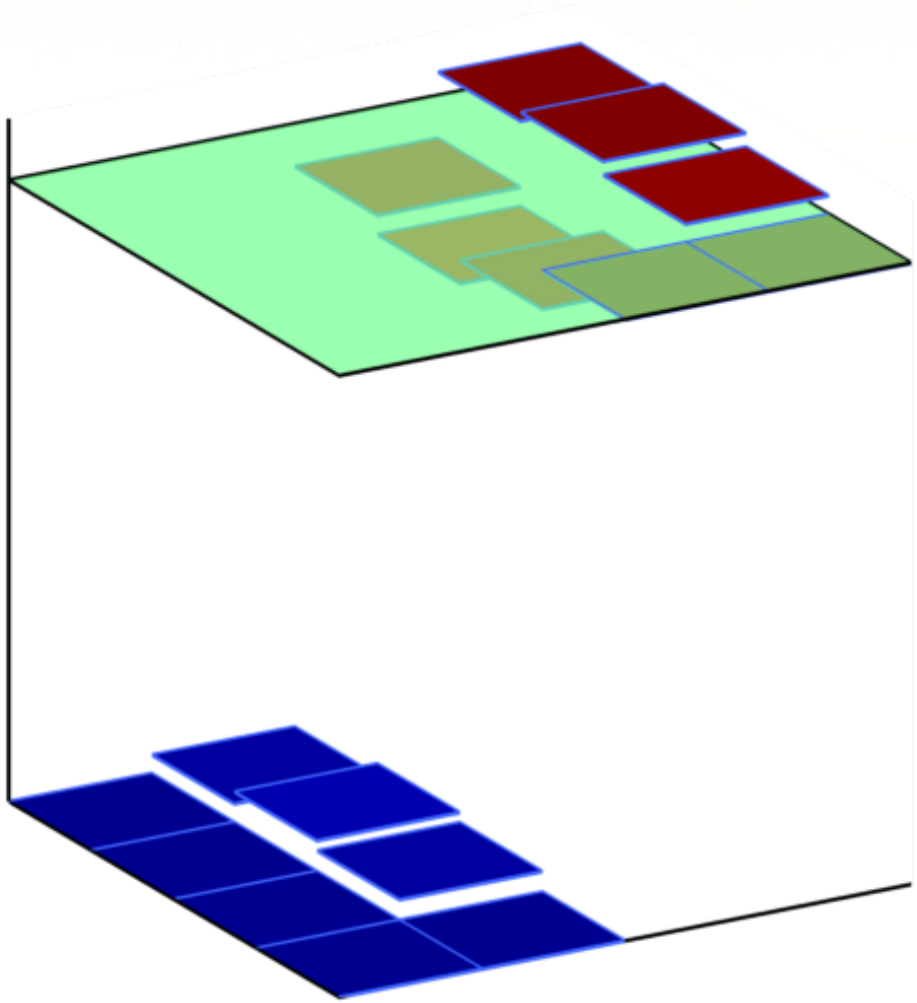
Remap



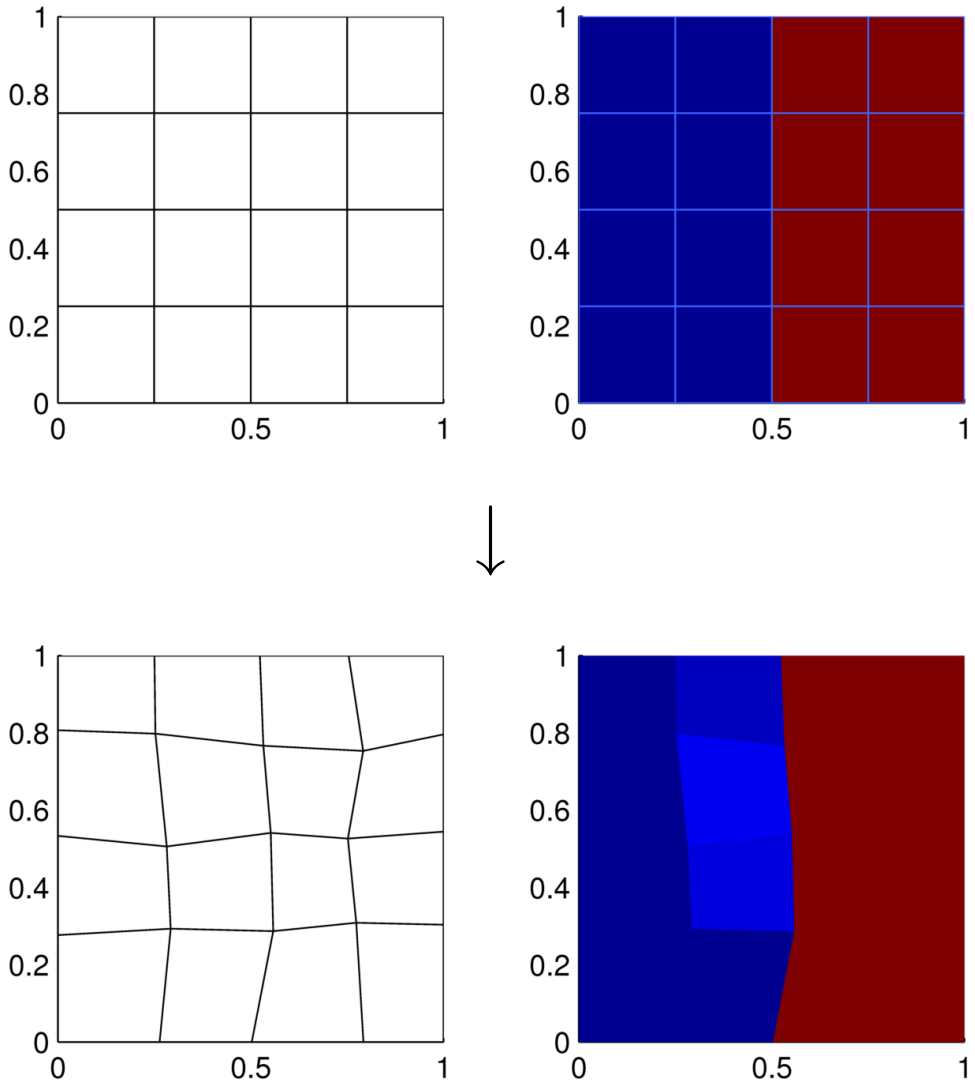
Remap



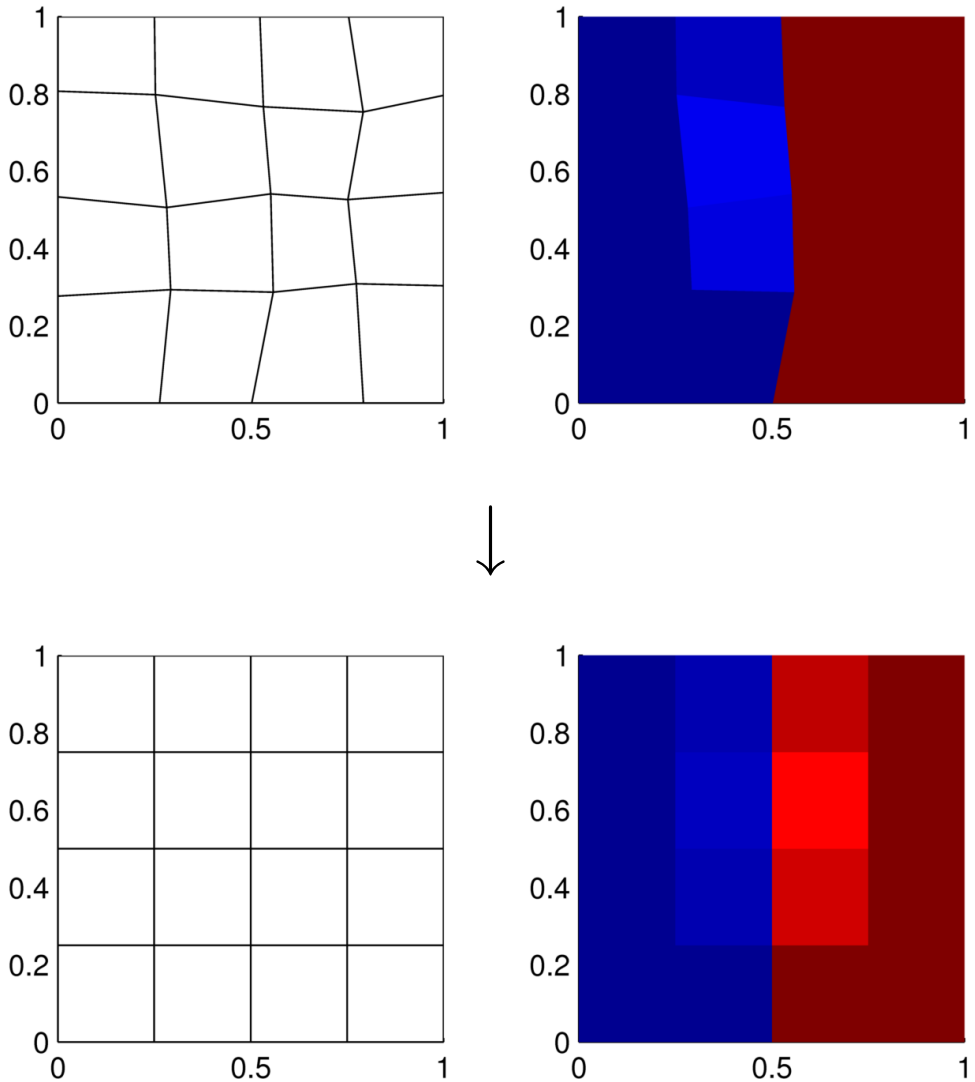
Remap



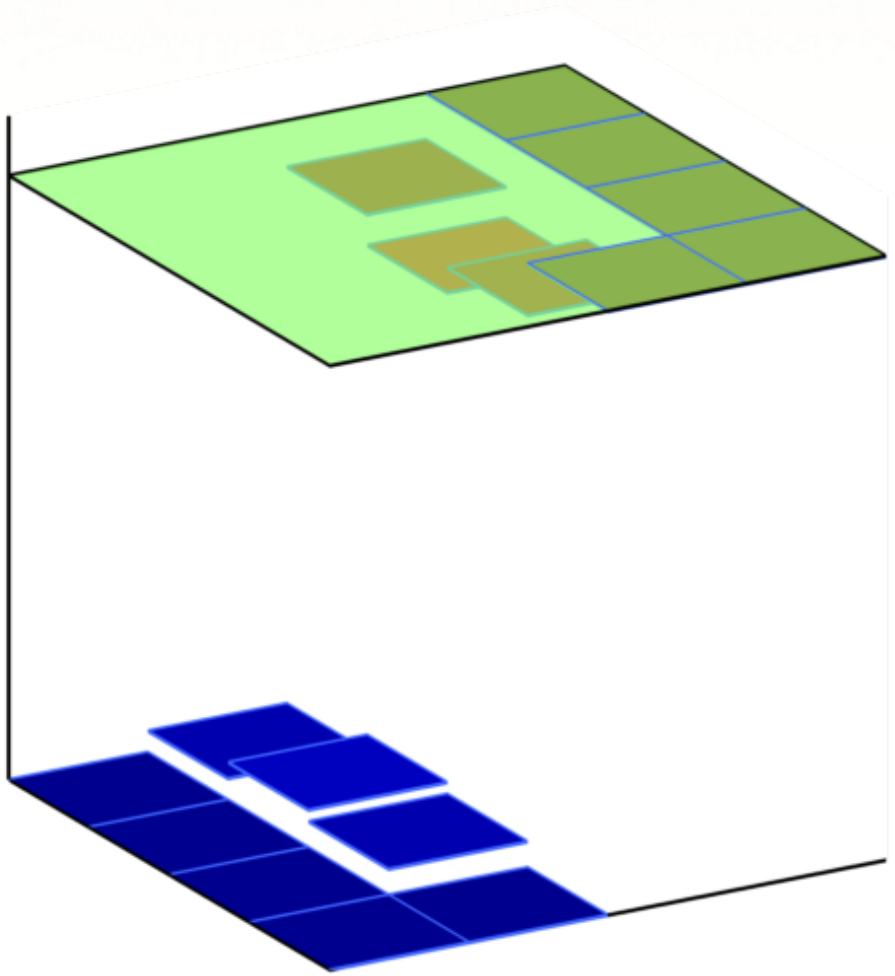
Remap



Remap



Remap



Mass-density remap

Given an *old* mesh $C(\Omega)$ and the mean density values ρ_i on the old mesh cells c_i, find accurate approximations $\widetilde{m}_i$ for the masses on a *new* mesh $\widetilde{C}(\Omega)$, associated with the new cells $\widetilde{c}_i$:

$$\widetilde{m}_i \approx \widetilde{m}_i^{\text{ex}} = \int_{\widetilde{c}_i} \rho(\mathbf{x}) dV\,, \quad i = 1, \dots, C\,; \quad \textbf{such that}$$

C1. The total mass is conserved: $\sum_{i=1}^{C} \widetilde{m}_i = \sum_{i=1}^{C} m_i = M$.

C2. If $\rho(\mathbf{x})$ is a global linear function on Ω, then the remapped masses are exact:

$$\widetilde{m}_i = \widetilde{m}_i^{\text{ex}} = \int_{\widetilde{c}_i} \rho(\mathbf{x}) dV\,, \quad i = 1, \dots, C\,.$$

C3. The approximation of the mean density on the new cells, $\widetilde{\rho}_i = \frac{\widetilde{m}_i}{\widetilde{\mu}_i}$, satisfies the local bounds

$$\rho_i^{\min} \le \widetilde{\rho}_i \le \rho_i^{\max}\,, \quad i = 1, \dots, C\,,$$

in other words:

$$\widetilde{m}_i^{\min} := \rho_i^{\min} \widetilde{\mu}_i \le \widetilde{m}_i \le \rho_i^{\max} \widetilde{\mu}_i =: \widetilde{m}_i^{\max}\,.$$

A few comments

19xx–2009:

- Remap is a well-studied problem.
- Important applications in transport (ALE); many others.
- The constraints (C1)–(C3) are typically handled *by construction*:
  — a careful choice of variables in the remap scheme;
  — a special reconstruction procedure; and
  — a particular choice of 'limiter' (WIKIPEDIA: 15 slope limiters).
- Example: Flux-corrected remap (FCR).

2009–2012:

- We use global optimization ideas to reconcile (C1)–(C3).
- A mathematically rigorous way to handle constraints.
- Elegant theory, and connections to methods like FCR.
- Improved accuracy and robustness.

2012–2013:

- Optimization-based remap at the cost of conventional remap.

The remap problem

Optimization-based remap (OBR)

Optimization algorithms

Remap results

Adaptable targets

Transport results

Optimization-based formulations of remap

Generic formulation

- Introduce *exact mass update* operator $\widetilde{m}^{\text{ex}} = L(\rho(\mathbf{x}))$.
- Approximate $L(\rho(\mathbf{x}))$ via $L_h(m, u(\rho))$, where

$$\widetilde{m}^{\text{ex}} = L_h(m, u(\rho)) \qquad \forall \rho(\mathbf{x}) \in \mathcal{C}\,,$$

  i.e., it is exact for a given class $\mathcal{C}$ of functions, e.g., linear functions.
- Define a *target* u^{T} such that $L_h(m, u^{\mathsf{T}}) = \widetilde{m}^{\text{ex}},\ \forall \rho(\mathbf{x}) \in \mathcal{C}$.
- Solve optimization problem:

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{subject to} \\ L_h(m, u^{\mathsf{T}}) = \widetilde{m}^{\text{ex}} \quad \forall \rho(\mathbf{x}) \in \mathcal{C}\,; \quad \widehat{u} \in \mathcal{U}^h \\ \displaystyle\sum_{i=1}^{C} m_i = \sum_{i=1}^{C} (L_h(m, \widehat{u}))_i \quad \text{and} \quad \widetilde{m}^{\min} \le L_h(m, \widehat{u}) \le \widetilde{m}^{\max}\,. \end{cases}$$

Optimization-based formulations of remap

Flux-variable flux-target (FVFT) formulation

Given the mesh side-to-cell incidence matrix $\mathbf{D}$, define the approximate mass update operator of the form

$$\widetilde{m} = L_h(m, u(\rho)) := m + \mathbf{D}u(\rho)\,,$$

where $u(\rho)$ approximates the exact *mass fluxes* on the swept regions:

$$u_j^{\text{ex}} = \int_{r_j} \rho(\mathbf{x})dV \quad j \in S(\Omega)\,.$$

Compute target $u_j^{\mathsf{T}} := \int_{r_j} \rho^h(\mathbf{x})dV;\ j = 1, \ldots, S$; solve

$$\begin{cases} \text{minimize} & \frac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{subject to} \\ \widehat{u} \in S_0^h & \text{and} \quad \widetilde{m}^{\min} \leq m + \mathbf{D}\widehat{u} \leq \widetilde{m}^{\max}\,. \end{cases}$$

Optimization-based formulations of remap

Mass-variable mass-target (MVMT) formulation

Define the approximate mass update operator of the form

$$\widetilde{m} = L_h(m, u(\rho)) := m + u(\rho)\,,$$

where $u(\rho)$ approximates the exact *mass increments* between the new cells and old cells:

$$u_i^{\text{ex}} = \int_{\widetilde{c}_i} \rho(\mathbf{x})dV - \int_{c_i} \rho(\mathbf{x})dV \quad i \in C(\Omega)\,.$$

Compute target $u_i^{\mathsf{T}} := \int_{\widetilde{c}_i} \rho^h(\mathbf{x})dV - \int_{c_i} \rho^h(\mathbf{x})dV;\ i = 1, \ldots, C$; solve

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{subject to} \\ \widehat{u} \in C^h\,; \quad \displaystyle\sum_{i=1}^{C} \widehat{u}_i = 0 \quad \text{and} \quad \widetilde{m}^{\min} \leq m + \widehat{u} \leq \widetilde{m}^{\max}\,. \end{cases}$$

Optimization-based formulations of remap

Theoretical results

Uniqueness: Under very mild assumptions on mesh connectivity, FVFT and MVMT are strictly convex quadratic programs with *unique solutions*.

High-order: Suppose that the barycenter of each new cell is contained within the *convex hull* of the barycenters of its old neighbors. Then, FVFT and MVMT *preserve globally linear functions*.

Monotone: The feasible set is nonempty; hence, MVMT and FVFT *preserve local bounds*.

Big brother: FCR is a "local" version of FVFT, in the sense that its feasible set is separable, per cell. However, FCR's feasible set is *fully contained in and generally smaller than* FVFT's feasible set. Hence, FVFT is provably more accurate.

Optimization-based formulations of remap

FVFT ($\widehat{u} \in S_0^h \equiv \mathbb{R}^S$)

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{s.t.} \\ \widetilde{m}^{\min} \leq m + \mathbf{D}\widehat{u} \leq \widetilde{m}^{\max} \end{cases}$$

MVMT ($\widehat{u} \in C^h \equiv \mathbb{R}^C$)

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{s.t.} \\ \displaystyle\sum_{i=1}^{C} \widehat{u}_i = 0\,; \quad \widetilde{m}^{\min} \leq m + \widehat{u} \leq \widetilde{m}^{\max} \end{cases}$$

Preliminary comparison

- The objective functions are structurally identical and quite simple.
- The FVFT constraint is a **globally coupled** inequality constraint; its stencil is **sparse** and equivalent to that of the discrete divergence operator. In Bochev, Ridzal, Scovazzi, Shashkov (2011) the FVFT problem is solved using a semismooth Newton iteration; requires global linear system solves!
- The MVMT inequality constraint is a **simple bound constraint**; however, there is a single **dense** equality constraint that couples all variables!

Optimization-based formulations of remap

FVFT ($\widehat{u} \in S_0^h \equiv \mathbb{R}^S$)

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{s.t.} \\ \widetilde{m}^{\min} \leq m + \mathbf{D}\widehat{u} \leq \widetilde{m}^{\max} \end{cases}$$

MVMT ($\widehat{u} \in C^h \equiv \mathbb{R}^C$)

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{s.t.} \\ \displaystyle\sum_{i=1}^{C} \widehat{u}_i = 0\,; \quad \widetilde{m}^{\min} \leq m + \widehat{u} \leq \widetilde{m}^{\max} \end{cases}$$

Preliminary comparison

- The objective functions are structurally identical and quite simple.
- The FVFT constraint is a **globally coupled** inequality constraint; its stencil is **sparse** and equivalent to that of the discrete divergence operator. In Bochev, Ridzal, Scovazzi, Shashkov (2011) the FVFT problem is solved using a semismooth Newton iteration; requires global linear system solves!
- The MVMT inequality constraint is a **simple bound constraint**; however, there is a single **dense** equality constraint that couples all variables!
  $\rightarrow$ Any advantages?

MVMT Algorithm

We solve

$$\begin{cases} \text{minimize} \quad \dfrac{1}{2}\|\widehat{u} - u^{\mathsf{T}}\|_{\ell_2}^2 \qquad \text{subject to} \\ \displaystyle\sum_{i=1}^{C} \widehat{u}_i = 0 \quad \text{and} \quad \widetilde{m}^{\min} \leq m + \widehat{u} \leq \widetilde{m}^{\max}. \end{cases}$$

Known as the **singly linearly constrained QP with simple bounds**, see Dai, Fletcher (2006, Math. Program.).

Key observation: The related optimization problem without the mass conservation constraint, $\sum_{i=1}^{C} \widehat{u}_i = 0$, is **fully separable**!

The related problem can be solved by independently (and concurrently) solving C **one-dimensional** quadratic programs with simple bounds.

Goal: Satisfy the second constraint, $\sum_{i=1}^{C} \widehat{u}_i = 0$, "in a few iterations".

MVMT Algorithm

Define Lagrangian functional $\mathcal{L} : \mathbb{R}^C \times \mathbb{R} \times \mathbb{R}^C \times \mathbb{R}^C \to \mathbb{R}$,

$$\mathcal{L}(\widehat{u}, \lambda, \mu_1, \mu_2) = \frac{1}{2}\sum_{i=1}^{C}(\widehat{u}_i - u_i^{\mathsf{T}})^2 - \lambda \sum_{i=1}^{C} \widehat{u}_i -$$

$$\sum_{i=1}^{C} \mu_{1,i}\left(\widehat{u}_i - \widetilde{m}_i^{\min} + m_i\right) - \sum_{i=1}^{C} \mu_{2,i}\left(\widetilde{m}_i^{\max} - m_i - \widehat{u}_i\right) ,$$

where $\widehat{u} \in \mathbb{R}^C$ are the *primal optimization variables*, and $\lambda \in \mathbb{R}$, $\mu_1 \in \mathbb{R}^C$, and $\mu_2 \in \mathbb{R}^C$ are the *Lagrange multipliers*.

Karush-Kuhn-Tucker (KKT) conditions:

$$\widehat{u}_i = u_i^{\mathsf{T}} + \lambda + \mu_{1,i} - \mu_{2,i}; \quad i = 1, \ldots, C$$

$$\widetilde{m}_i^{\min} - m_i \leq \widehat{u}_i \leq \widetilde{m}_i^{\max} - m_i; \quad i = 1, \ldots, C$$

$$\mu_{1,i} \geq 0\,, \quad \mu_{2,i} \geq 0; \quad i = 1, \ldots, C$$

$$\mu_{1,i}\left(\widehat{u}_i - \widetilde{m}_i^{\min} + m_i\right) = 0\,, \quad \mu_{2,i}\left(-\widehat{u}_i + \widetilde{m}_i^{\max} - m_i\right) = 0; \quad i = 1, \ldots, C$$

$$\sum_{i=1}^{C} \widehat{u}_i = 0$$

MVMT Algorithm

Idea: We solve the KKT conditions directly.

First, we focus on the conditions in black, separable in the index i. For any *fixed* value of λ a solution to the "black" conditions is given by

$$\begin{cases} \widehat{u}_i = u_i^{\mathsf{T}} + \lambda; & \mu_{1,i} = \mu_{2,i} = 0 & \text{if} \quad \widetilde{m}_i^{\min} - m_i \le u_i^{\mathsf{T}} + \lambda \le \widetilde{m}_i^{\max} - m_i \\ \widehat{u}_i = \widetilde{m}_i^{\min} - m_i; & \mu_{2,i} = 0,\ \mu_{1,i} = \widehat{u}_i - u_i^{\mathsf{T}} - \lambda & \text{if} \quad u_i^{\mathsf{T}} + \lambda < \widetilde{m}_i^{\min} - m_i \\ \widehat{u}_i = \widetilde{m}_i^{\max} - m_i; & \mu_{1,i} = 0,\ \mu_{2,i} = u_i^{\mathsf{T}} - \widehat{u}_i + \lambda & \text{if} \quad u_i^{\mathsf{T}} + \lambda > \widetilde{m}_i^{\max} - m_i\,, \end{cases}$$

for all $i = 1, \ldots, C$.

Ignoring μ_1 and μ_2 and treating $\widehat{u}_i$ as a function of λ yields

$$\widehat{u}_i(\lambda) = \text{median}(\widetilde{m}_i^{\min} - m_i,\ u_i^{\mathsf{T}} + \lambda,\ \widetilde{m}_i^{\max} - m_i)\,, \qquad i = 1, \ldots, C\,.$$

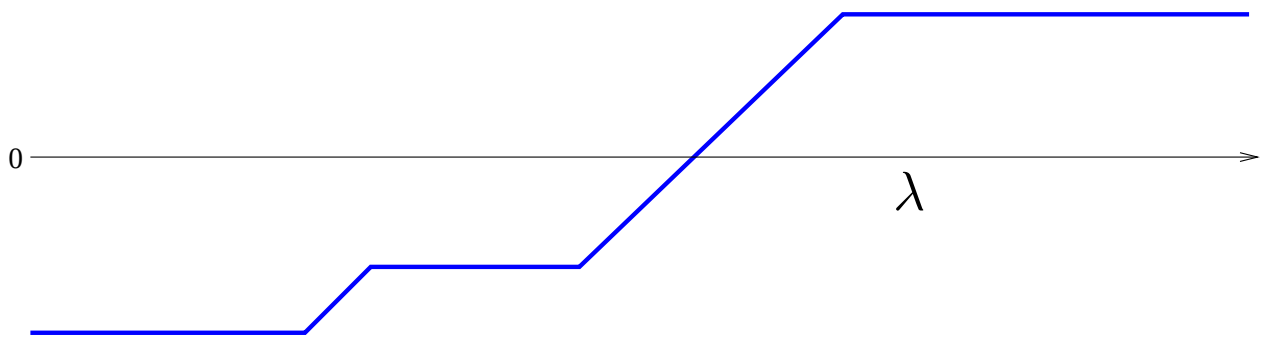
This is a trivial $\mathcal{O}(C)$ computation.

MVMT Algorithm

Second, we adjust λ in an outer iteration in order to satisfy the "red" constraint $\sum_{i=1}^{C} \widehat{u}_i(\lambda) = 0$.

When we find the λ^* such that $\sum_{i=1}^{C} \widehat{u}_i(\lambda^*) = 0$ holds, we will have solved the full KKT conditions.

The function $\sum_{i=1}^{C} \widehat{u}_i(\lambda)$ is a piecewise linear, monotonically increasing function of a single scalar variable λ. Therefore, a **secant method** can be efficiently employed as the outer iteration.



In all our examples, the algorithm requires $\leq$ 5 secant iterations!

MVMT Algorithm

1. **Initialization:** Set $\lambda_0 \leftarrow 0$, $\eta \leftarrow 10^{-12}$ and $\Delta\lambda^{FD} \leftarrow 10^{-8}$.
2. **Finite difference step:**
   1. Compute $\widehat{u}_i(\lambda_0) \leftarrow \text{median}(\widetilde{m}_i^{\min} - m_i,\ u_i^{\mathsf{T}} + \lambda_0,\ \widetilde{m}_i^{\max} - m_i)\ \forall i$.

      Compute residual $r_p \leftarrow \sum_{i=1}^{C} \widehat{u}_i(\lambda_0)$.

      If $|r_p| < \eta$, then return $\widehat{u}_i(\lambda_0)\ \forall i$ and stop.
   2. Set $\widehat{u}_i(\lambda_0 + \Delta\lambda^{FD}) \leftarrow \text{median}(\widetilde{m}_i^{\min} - m_i,\ u_i^{\mathsf{T}} + \lambda_0 + \Delta\lambda^{FD},\ \widetilde{m}_i^{\max} - m_i)\ \forall i$.

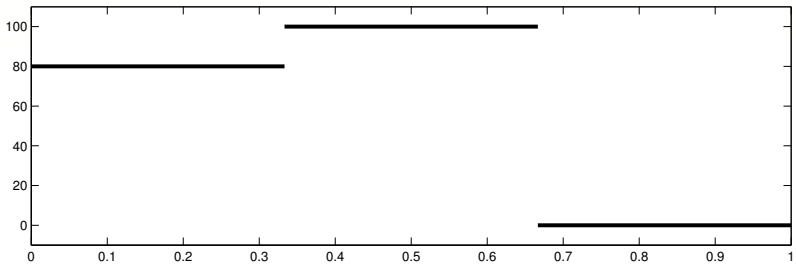
      Compute residual $r_c \leftarrow \sum_{i=1}^{C} \widehat{u}_i(\lambda_0 + \Delta\lambda^{FD})$.
   3. Set $\alpha \leftarrow \Delta\lambda^{FD}/(r_c - r_p)$.
   4. Set $\lambda_p \leftarrow \lambda_0$. Set $\lambda_c \leftarrow \lambda_p - \alpha r_p$.
3. **While** $|r_c| > \eta$ **(Secant Iteration)**
   1. Compute $\widehat{u}_i(\lambda_c) \leftarrow \text{median}(\widetilde{m}_i^{\min} - m_i,\ u_i^{\mathsf{T}} + \lambda_c,\ \widetilde{m}_i^{\max} - m_i)\ \forall i$.

      Compute residual $r_c \leftarrow \sum_{i=1}^{C} \widehat{u}_i(\lambda_c)$.
   2. Set $\alpha \leftarrow (\lambda_p - \lambda_c)/(r_p - r_c)$. Set $r_p \leftarrow r_c$.
   3. Set $\lambda_p \leftarrow \lambda_c$. Set $\lambda_c \leftarrow \lambda_c - \alpha r_c$.

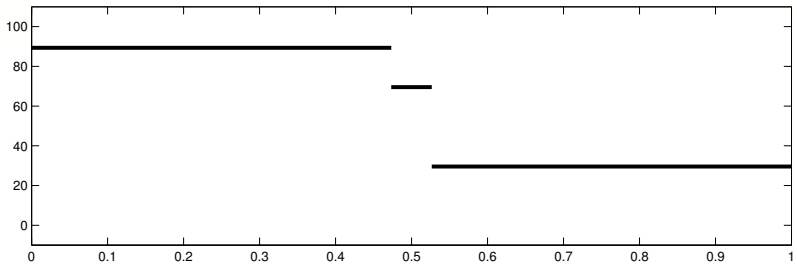
   End While
4. Return $\widehat{u}_i(\lambda_c)\ \forall i$ and stop.

Remap: Shape preservation

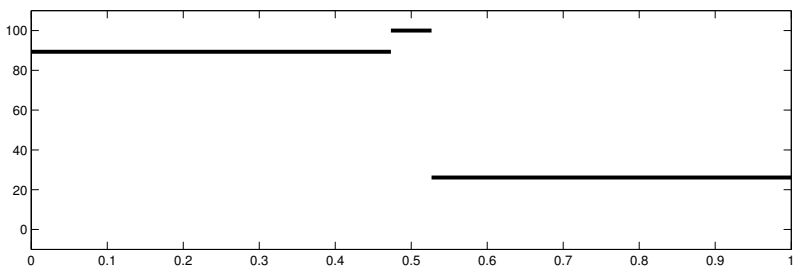
Original Density (Old Mesh)



FCR Result (New Mesh)



FVFT Result (New Mesh)



MVMT Result (New Mesh)

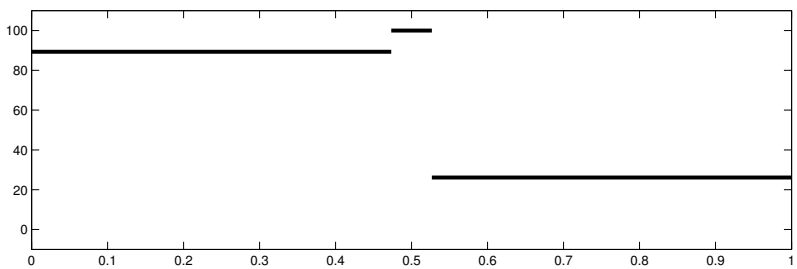


Figure: FVFT and MVMT preserve the shape of the peak, giving identical results, while FCR loses accuracy and transforms the peak into a step function.

Remap: Monotonicity preservation

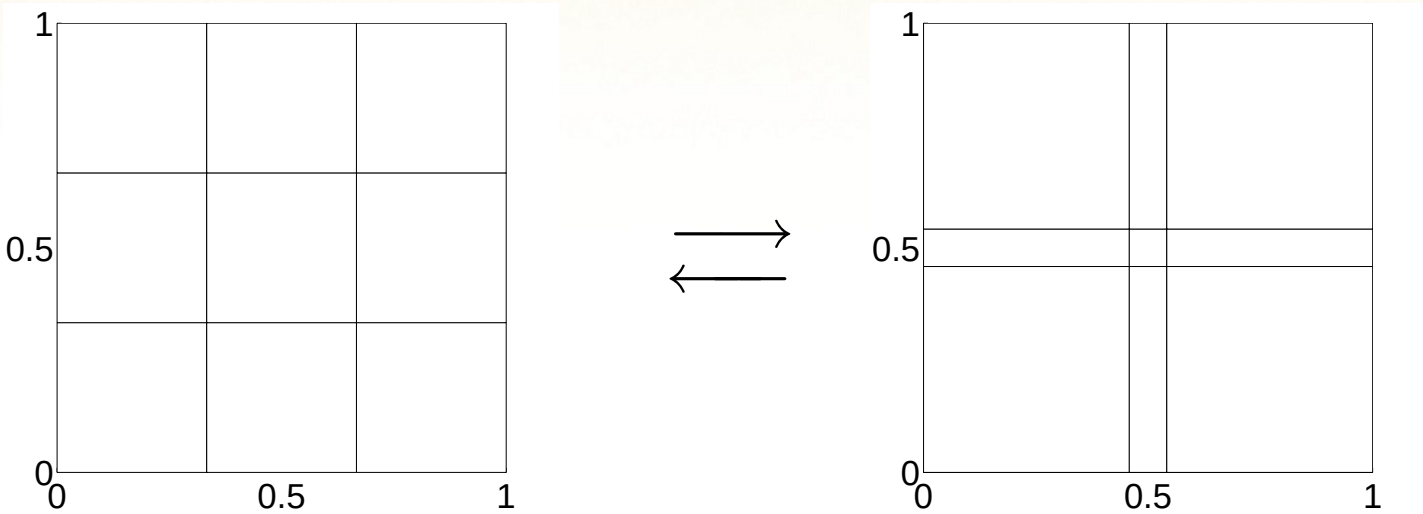


Figure: A 3×3 uniform mesh (left pane) and a "compressed" mesh (right pane) with a $\ell \times \ell$-fold compression, $\ell = 4$, of the middle cell.

	$\ell = 5$	$\ell = 6$	$\ell = 7$	$\ell = 14$	$\ell = 15$	$\ell = 16$	$\ell = 100$
FVFT	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**
MVMT	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**
FCR	**yes**	**yes**	**yes**	**yes**	no	no	no

Table: Monotonicity of FVFT, MVMT and FCR, applied to a linear density.

Remap: Asymptotic accuracy

		FVFT		MVMT		FCR	
mesh	remaps	L_1 err	rate	L_1 err	rate	L_1 err	rate
64×64	320	1.09e-03	—	1.15e-03	—	1.03e-03	—
128×128	640	2.69e-04	2.03	2.77e-04	2.06	2.81e-04	1.87
256×256	1280	6.71e-05	2.01	6.82e-05	2.04	9.23e-05	1.74
512×512	2560	1.68e-05	2.01	1.69e-05	2.03	3.65e-05	1.60

Table: FVFT, MVMT and FCR errors and convergence rate estimates for the sine density using a sequence of repeated-repair ("torture") cyclic meshes. FVFT and MVMT are second-order accurate; FCR is not.

In summary, optimization-based remappers are more robust and more accurate than FCR.

The results are similar if FCR is replaced with another non-global (explicit) remapper.

Remap: Speed

mesh	remaps	FCR time(sec)	MVMT time(sec)	ratio	FVFT time(sec)	ratio
Sine						
64×64	320	4.1	4.1	1.0	6.8	1.7
128×128	640	25.1	24.6	1.0	48.7	1.9
256×256	1280	177.2	173.1	1.0	384.6	2.2
512×512	2560	2049.1	1918.0	0.9	3677.5	1.8
Peak						
64×64	320	4.9	4.9	1.0	7.7	1.6
128×128	640	28.0	28.6	1.0	53.9	1.9
256×256	1280	194.5	192.8	1.0	400.8	2.1
512×512	2560	2060.1	2096.5	1.0	4410.5	2.1
Shock						
64×64	320	4.6	4.9	1.1	9.0	2.0
128×128	640	27.4	29.1	1.0	85.1	3.1
256×256	1280	192.5	195.6	1.0	414.4	2.2
512×512	2560	2064.9	2146.7	1.0	3117.1	1.5

Table: Comparison of the computational cost on the tensor-product cyclic grid. Ratios of run times of MVMT and FVFT with respect to FCR are included.

The remap problem

Optimization-based remap (OBR)

Optimization algorithms

Remap results

Adaptable targets

Transport results

Reconstruction residual

- Targets are built from the reconstruction:

$$\rho^h(\mathbf{x})|_{c_i} := \rho_i^h(\mathbf{x}) = \rho_i + \mathbf{g}_i \cdot (\mathbf{x} - \mathbf{b}_i) \quad \forall c_i \in C(\Omega)\,,$$

  where ρ_i are density values on the old cells c_i, $\mathbf{g}_i$ is a least-squares approximation of the gradient $\nabla\rho$ based on ρ_i from the cells in the neighborhood $N(c_i)$, and $\mathbf{b}_i$ is the barycenter of c_i.

- Use a *reconstruction residual* to modify the gradient of $\rho^h(\mathbf{x})$:
  If the exact density is linear there holds

$$\rho(\mathbf{b}_i) = \frac{\int_{c_i} \rho(\mathbf{x}) dV}{\mu(c_i)} = \rho_i \quad \forall i = 1, \ldots, C\,.$$

  This property prompts the following definition,

$$q_i = \sum_{j \in N(c_i)} |\rho_j - \rho_i^h(\mathbf{b}_j)|\,.$$

Adaptable target definition

- Consider a family of real valued functions $\alpha_i(\xi)$ such that

$$\alpha_i(\xi) \geq 1 \quad \text{and} \quad \alpha_i(0) = 1\,.$$

- Define the adaptable density reconstruction as

$$\rho^A(\mathbf{x})|_{c_i} := \rho_i^A(\mathbf{x}) = \rho_i + \alpha_i(q_i)\mathbf{g}_i \cdot (\mathbf{x} - \mathbf{b}_i)\,.$$

- Substitute ρ^A for ρ^h to define adaptable OBR target extensions.

- One example is **MVMT-a**, where for the given constants $\gamma_1, \gamma_2 > 0$,

$$\alpha_i(q_i) = \begin{cases} 1 & \text{if } q_i / \max\limits_{i=1,\ldots,C}\{q_i\} \leq \gamma_1 \\ 1 + \gamma_2\, q_i / \max\limits_{i=1,\ldots,C}\{q_i\} & \text{otherwise}\,. \end{cases}$$

Note: Compare with the monotone reconstruction

$$\rho^L(\mathbf{x})|_{c_i} := \rho_i^L(\mathbf{x}) = \rho_i + \alpha_i(\rho, \rho^h)\mathbf{g}_i \cdot (\mathbf{x} - \mathbf{b}_i)\,,$$

where $\alpha_i(\rho, \rho^h)$, $0 \leq \alpha_i(\rho, \rho^h) \leq 1$, is a *limiting coefficient*.

Planar transport: Combo rotation

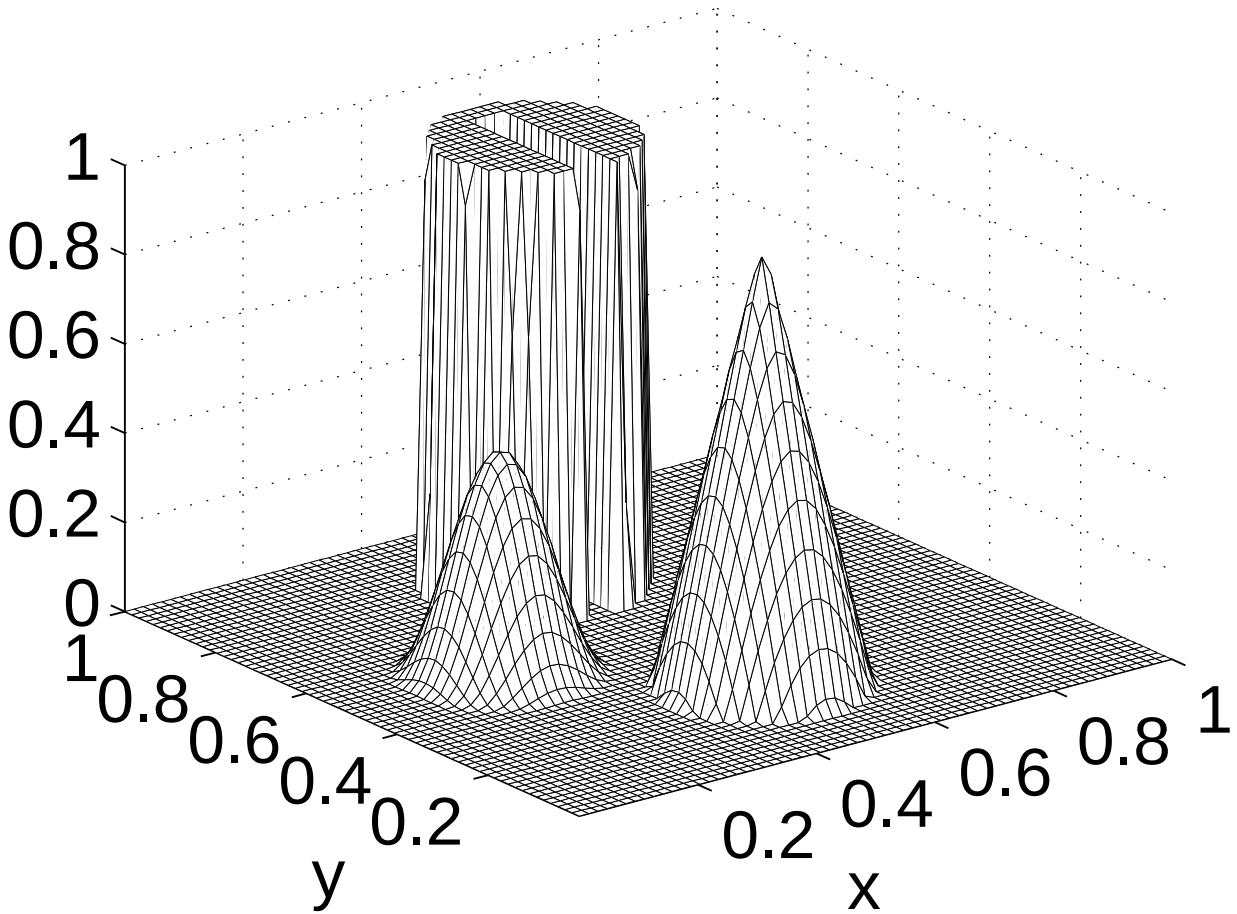
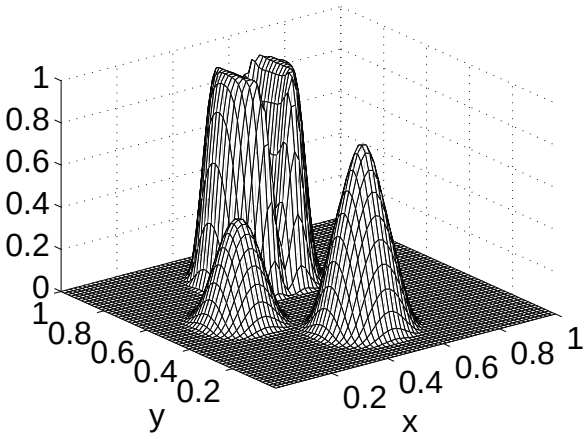


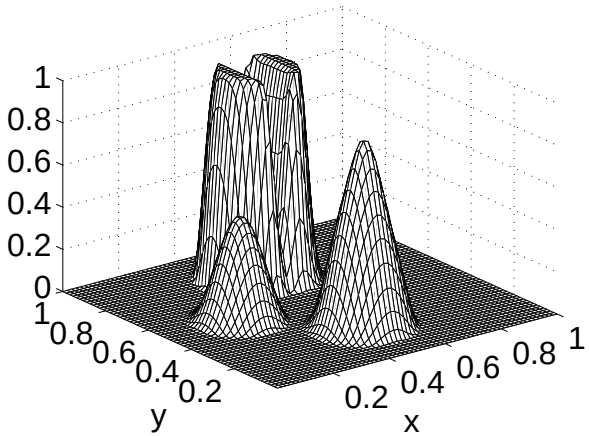
Figure: Initial data for the solid-body rotation test.

Planar transport: Combo rotation

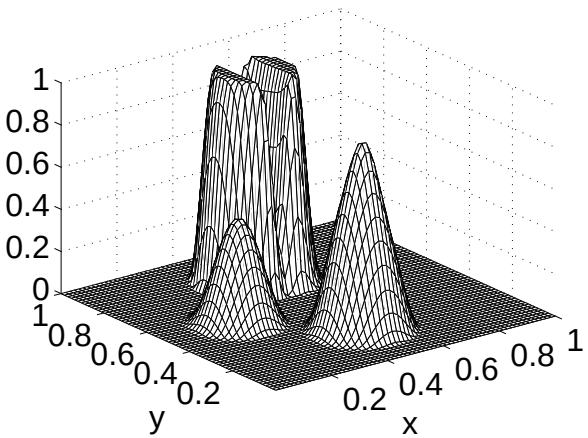
MVMT



FCR



FVFT



MVMT-a

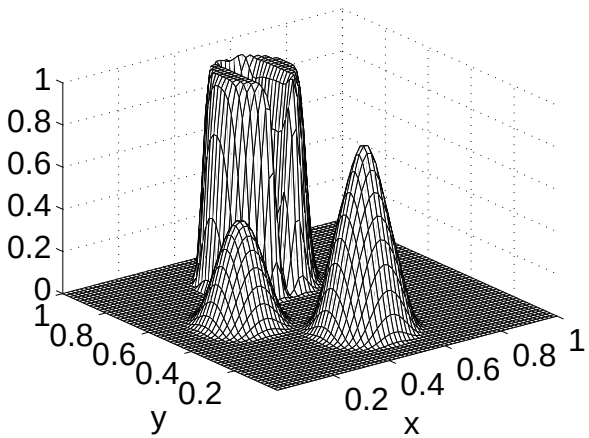


Figure: After one full revolution (810 time steps) on a 128×128 mesh.

Planar transport: Combo rotation

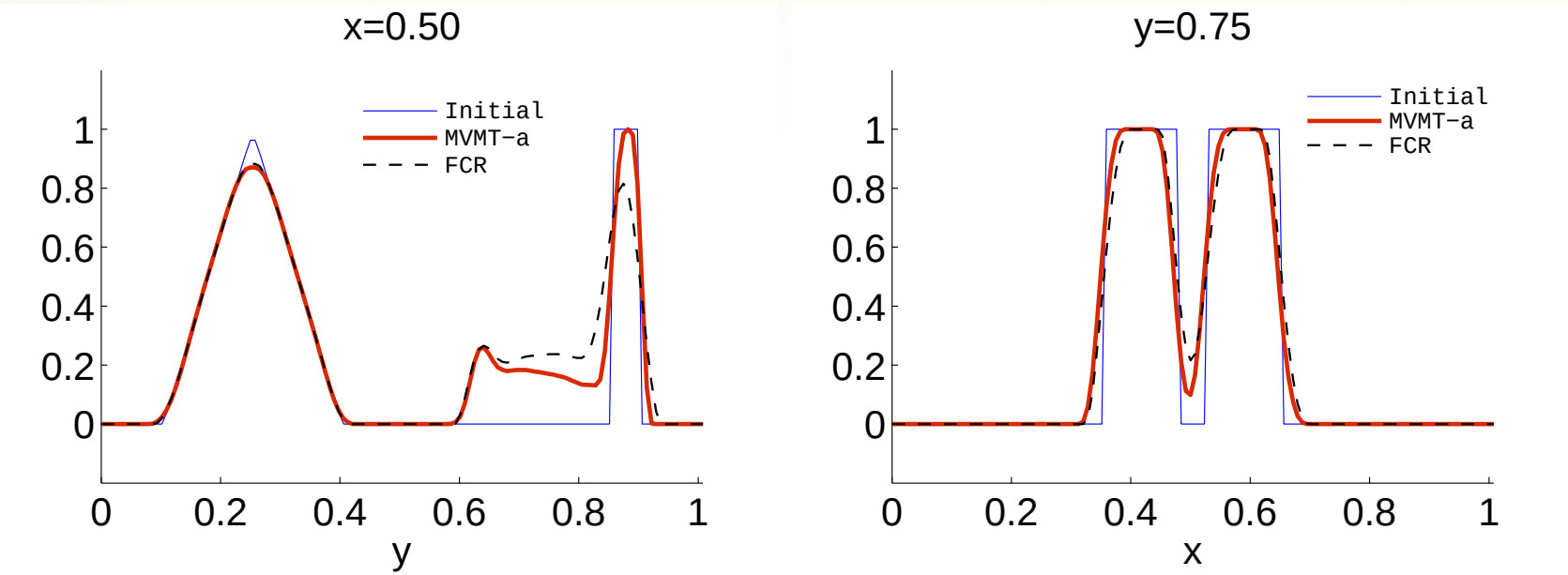


Figure: MVMT-a resolves the back side and the cylinder slot better than FCR.

		FCR		**FVFT**		**MVMT**		**MVMT-a**	
mesh	steps	L_1 error	rate	L_1 error	rate	L_1 error	rate	L_1 error	rate
64×64	408	3.42e-2	—	3.59e-2	—	3.93e-2	—	3.13e-2	—
128×128	810	1.94e-2	0.82	2.05e-2	0.80	2.34e-2	0.81	1.73e-2	0.86
256×256	1614	1.12e-2	0.81	1.19e-2	0.80	1.39e-2	0.80	1.12e-2	0.75

Table: Comparison of the L_1 errors of FCR, FVFT, MVMT and MVMT-a.

Planar transport: Combo rotation

mesh	steps	FCR time(sec)	FVFT time(sec)	ratio	MVMT time(sec)	ratio	MVMT-a time(sec)	ratio
64×64	408	3.3	63.7	19.3	3.4	1.0	3.8	1.1
128×128	810	26.4	496.4	18.8	26.2	1.0	28.8	1.1
256×256	1614	229.1	3464.2	15.1	222.7	1.0	230.9	1.0

Table: Computational cost.

FVFT is not cost-competitive in transport applications.

FCR, MVMT and MVMT-a show nearly identical speeds.

Planar transport: Cylinder translation

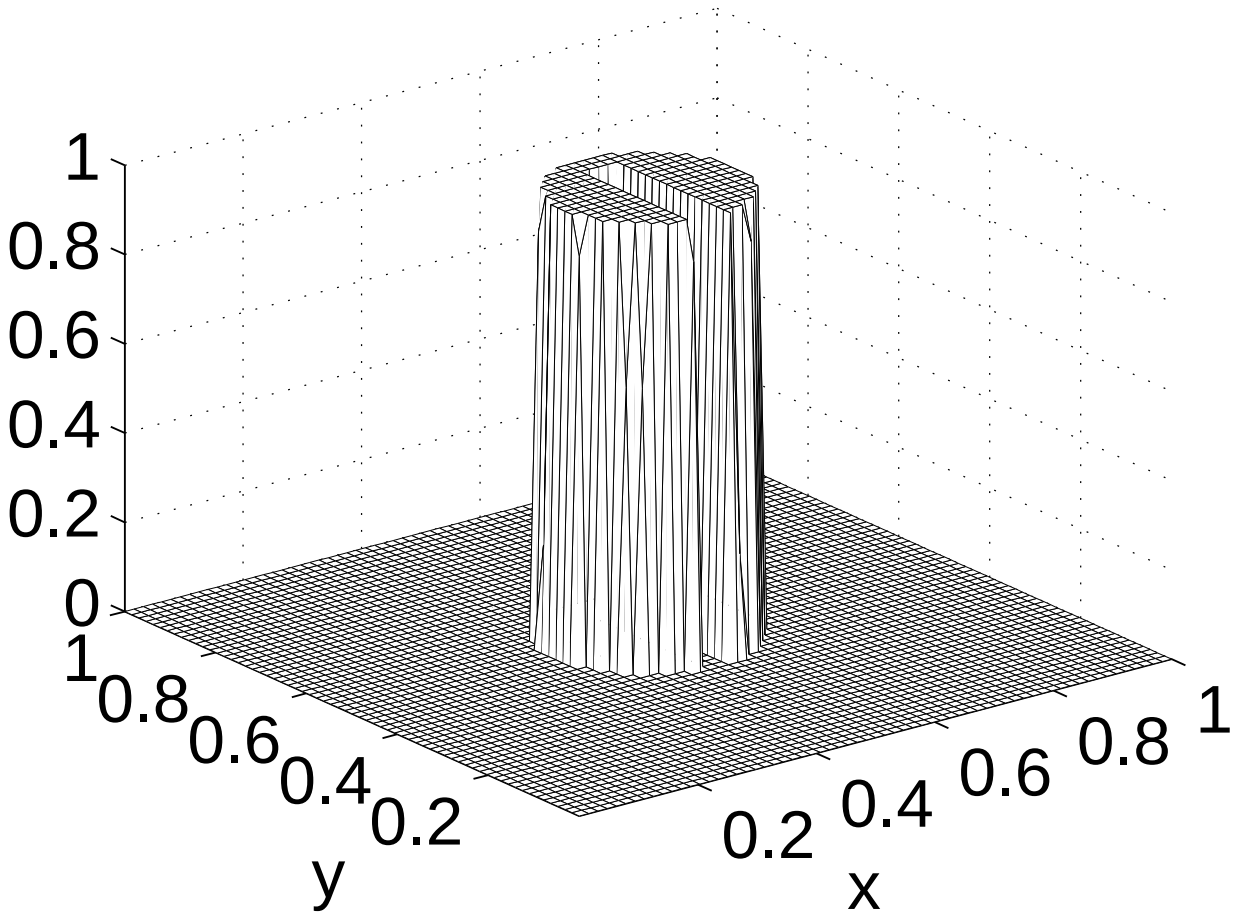
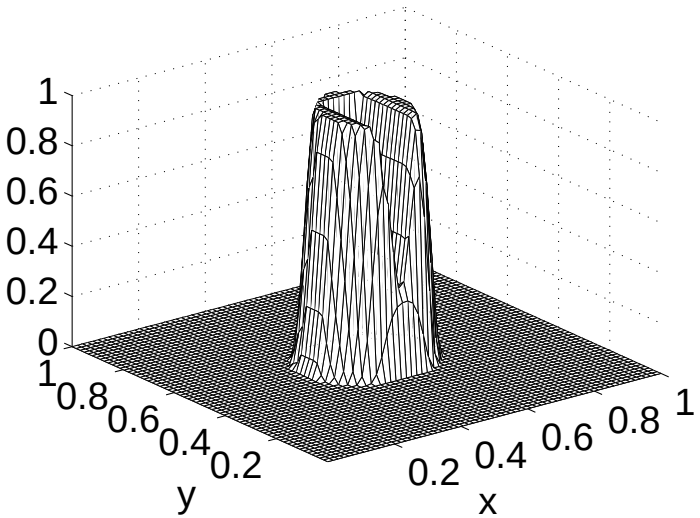


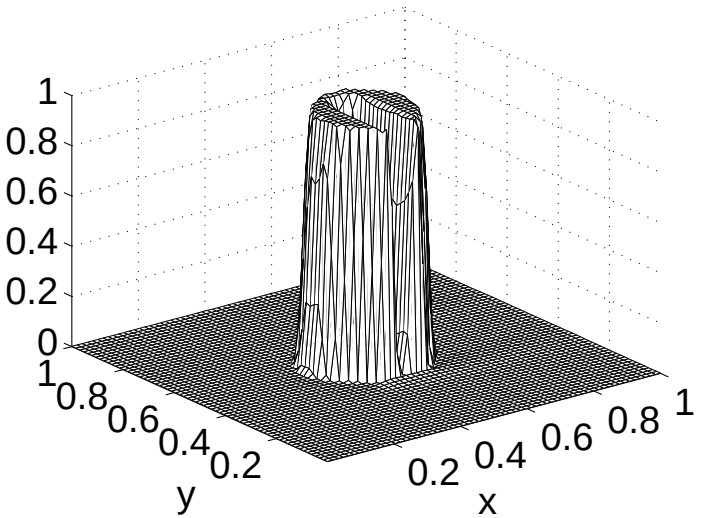
Figure: Initial data for the solid-body translation test. The cylinder is moved three widths of the domain to the right and three widths to the left.

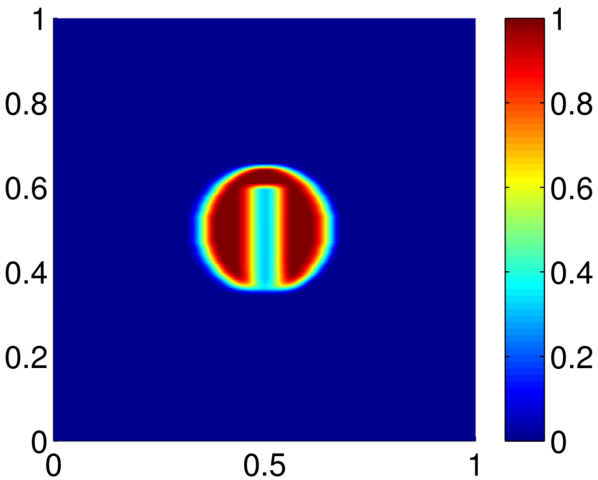
Planar transport: Cylinder translation

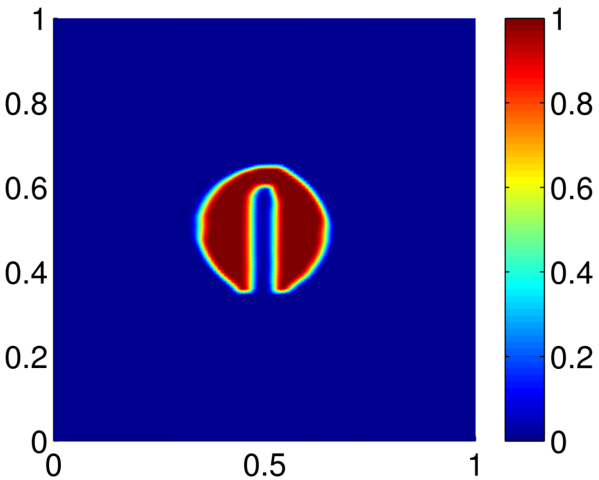
FCR



MVMT-a







Planar transport: Cylinder translation

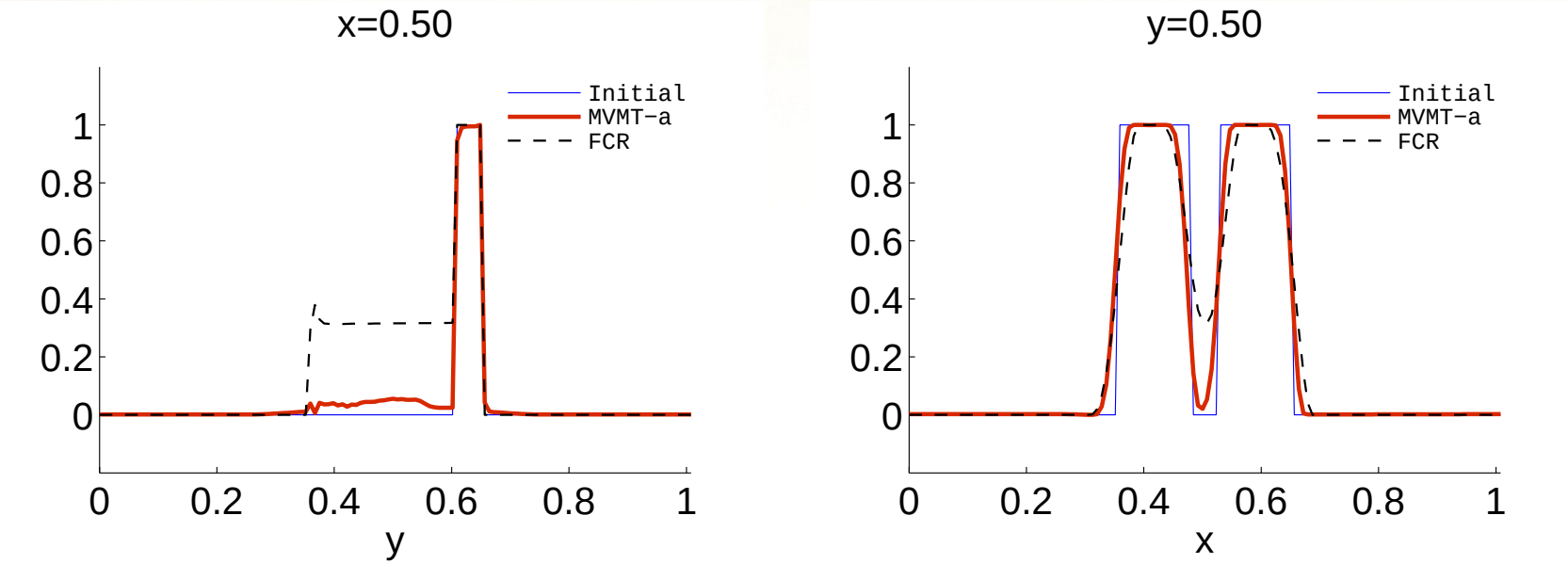


Figure: Comparison of the qualitative accuracy of MVMT-a and FCR for the solid-body translation test. MVMT-a resolves the back side and the slot of the cylinder much better than FCR.

mesh	steps	**FCR** time(sec)	**MVMT-a** time(sec)	ratio	**FCR** L_1 error	rate	**MVMT-a** L_1 error	rate
64×64	546	4.2	4.9	1.2	2.77e-2	—	2.75e-2	—
128×128	1092	35.2	37.8	1.1	1.59e-2	0.80	1.34e-2	1.04
256×256	2178	282.2	295.5	1.0	9.45e-3	0.78	6.70e-3	1.02

Spherical transport: Two-cylinder deformation (Nair/Lauritzen)

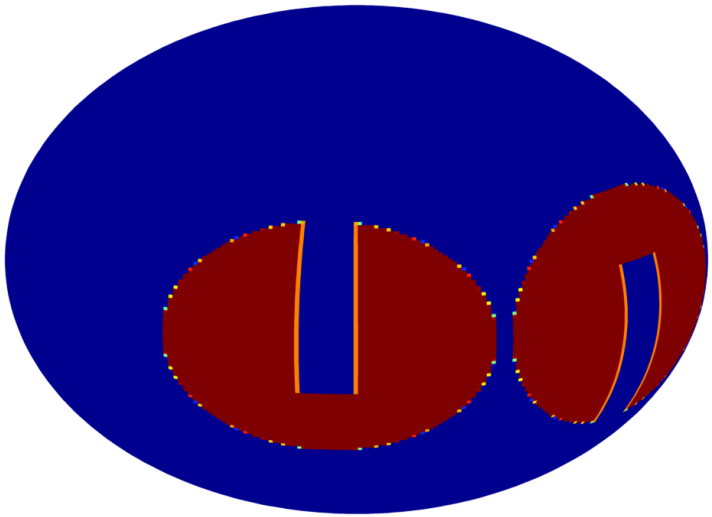


Figure: Initial data for the transport tests on the sphere.

Spherical transport: Two-cylinder deformation

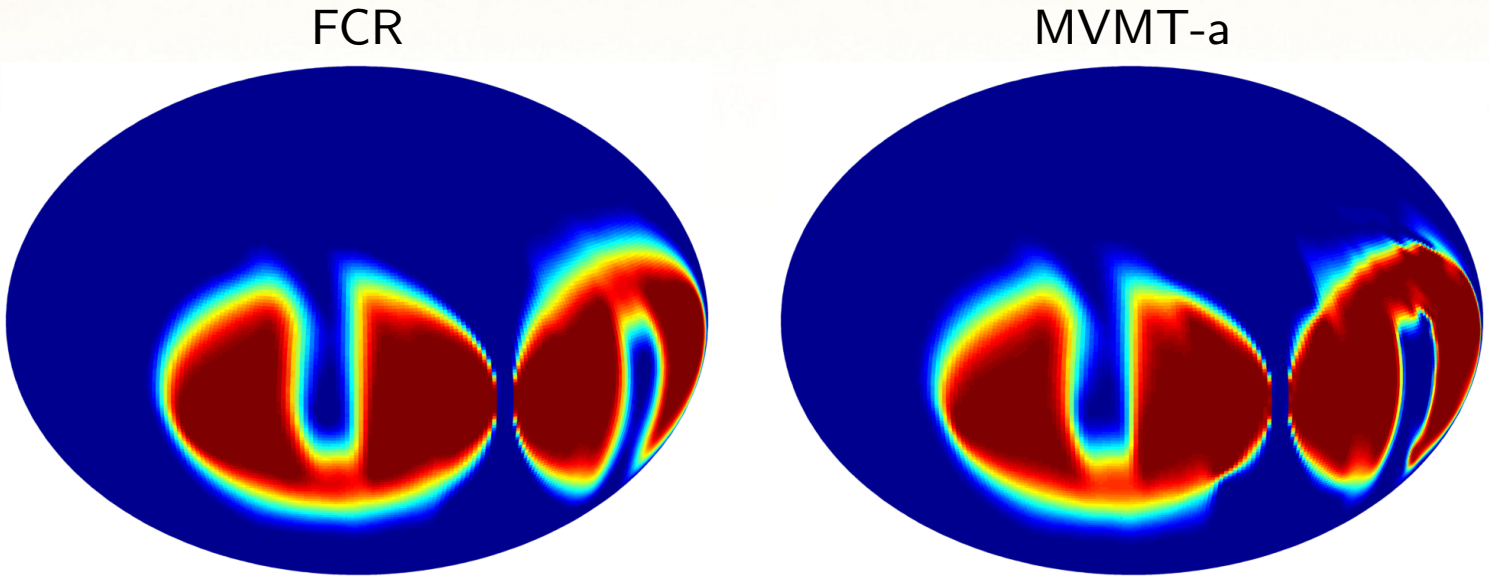


Figure: Transport results for the deformational flow test on the sphere at a final time $T = 5$ after 2400 time steps on a 0.75° mesh. The MVMT-a results appear sharper than the FCR results, particularly for the cylinder on the right.

		FCR	**MVMT-a**		**FCR**		**MVMT-a**	
mesh	steps	time(sec)	time(sec)	ratio	L_1 error	rate	L_1 error	rate
3°	600	23.0	24.2	1.1	4.34e-2	—	3.60e-2	—
1.5°	1200	187.7	190.0	1.0	2.85e-2	0.61	2.27e-2	0.66
0.75°	2400	1644.4	1717.7	1.0	1.67e-2	0.69	1.40e-2	0.68

Spherical transport: Two-cylinder rotation

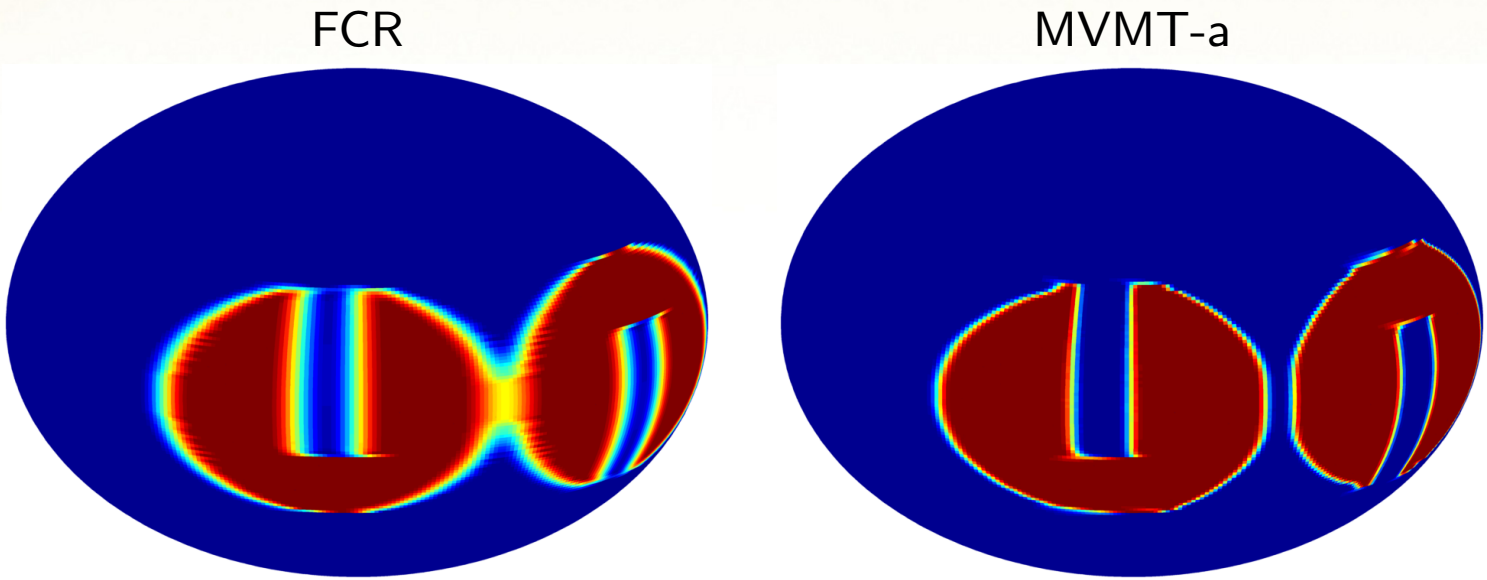


Figure: Transport results for the solid-body rotation test on the sphere, for two revolutions, left to right and back (1920 time steps) on a 0.75° mesh. The MVMT-a results provide a much better match to the initial data.

mesh	steps	**FCR** time(sec)	**MVMT-a** time(sec)	ratio	**FCR** L_1 error	rate	**MVMT-a** L_1 error	rate
3°	480	17.4	18.2	1.0	3.25e-2	—	2.79e-2	—
1.5°	960	132.5	151.6	1.1	1.99e-2	0.78	1.36e-3	1.04
0.75°	1920	1184.5	1379.0	1.2	1.10e-2	0.78	5.41e-3	1.18

Summary

- Optimization-based remappers (OBR) are more robust and more accurate than explicit remappers.
- The divide-and-conquer strategy separates accuracy concerns from the preservation of features such as monotonicity.
- The mass-based OBR algorithm, abbreviated MVMT, is as fast as flux-corrected remap.
- The optimization approach allows for specially tuned targets.

- Dual optimization variables may be used to tune targets.
- Multi-tracer transport can be done efficiently (in progress).
- Quadratic+ targets pose an interesting optimization challenge.

References:

Bochev, Ridzal, Scovazzi, Shashkov (2011,JCP) / Bochev, Ridzal, Shashkov (2013,JCP)
Bochev, Ridzal, Young, Peterson (2012,LNCS) / Bochev, Ridzal, Peterson (2013,JCP)