SAND2011-2046C

Optimization-based modeling with applications to transport. Part 3. Computational studies

Denis Ridzal[2], Joseph Young[1], Pavel Bochev[1], and Kara Peterson[1]

[1] Numerical Analysis and Applications,
[2] Optimization and Uncertainty Quantification,
Sandia National Laboratories[3], Albuquerque, NM 87185-1320, USA,
{dridzal,josyoun,pbboche,kjpeter}@sandia.gov

Abstract. This paper is the final of three related articles that develop and demonstrate a new optimization-based framework for computational modeling. The framework uses optimization and control ideas to assemble and decompose multiphysics operators and to preserve their fundamental physical properties in the discretization process. One application of the framework is in the formulation of robust algorithms for optimization-based transport (OBT). Based on the theoretical foundations established in Part 1 and the optimization algorithm for the solution of the remap subproblem, derived in Part 2, this paper focuses on the application of OBT to a set of benchmark transport problems. Numerical comparisons with two other transport schemes based on incremental remapping, featuring flux-corrected remap and the linear reconstruction with van Leer limiting, respectively, demonstrate that OBT is a competitive transport algorithm.

1 Introduction

In this and two companion papers [1, 2] we formulate and study a new optimization-based framework for computational modeling. One application of the framework, introduced in Part 1 [1], is in the formulation of a new class of optimization-based transport (OBT) schemes, which combine *incremental remap* [3] with the reformulation of the *remap subproblem* as an inequality-constrained quadratic program (QP) [4]. An efficient algorithm for the solution of the remap subproblem is presented in Part 2 [2]. In this paper we apply the OBT framework to a series of benchmark transport problems cited in [5].

Numerical comparisons with two other transport schemes based on incremental remapping are presented. The first scheme solves the remap subproblem using flux-corrected remap (FCR); for an FCR reference see [6]. We denote this scheme by FCRT (FCR based Transport). The second scheme solves the remap

[3] 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.

subproblem via a linear flux reconstruction with van Leer limiting, see [3] and references therein. We denote the latter transport scheme by LVLT (Linear Van Leer based Transport). In comparisons with FCRT and LVLT, we demonstrate that OBT, while computationally more expensive, can be more accurate and significantly more robust.

2 Implementation

The OBT framework is developed in Part 1 [1]. To summarize, OBT for mass density relies on an incremental remap procedure with the following steps: (1) move an original computational grid in the direction of the advection and obtain a new grid; (2) compute mass density updates on the new grid; and (3) remap mass density onto the original grid. The remap subproblem in step (3) is formulated as an inequality-constrained quadratic program (QP) and solved using Newton's method for piecewise differentiable systems, derived in Part 2 [2].

The QP describing the remap subproblem has the form

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

where $\vec{F}^H \in \mathbb{R}^M$ are the given discrete high-order fluxes, $\vec{b}_{\min} \in \mathbb{R}^K$ and $\vec{b}_{\max} \in \mathbb{R}^K$ are lower and upper bounds obtained from local mass density bounds on the new grid, and $\mathbf{A} \in \mathbb{R}^{K\times M}$ is an inequality-constraint matrix. Below we define the dimensions K and M for a concrete implementation of OBT. We also elaborate on the computation of $\vec{F}^H$ and $\mathbf{A}$.

For the implementation of OBT, FCRT and LVLT algorithms we use structured quadrilateral grids. If N_x and N_y are the numbers of intervals in x and y directions, respectively, then $K = N_x N_y$. The high-order flux vector $\vec{F}^H$ can be computed via integration over exact cell intersections, following the theory in Part 1 [1]. We avoid this potentially costly computation by using the concept of *swept regions*, see [7, 4], where mass exchanges are allowed only between cells that share a side. This simplifies the computation of high-order fluxes used in OBT, as well as the computation of low and high-order fluxes used in FCRT and LVLT. Following the swept-region approximation, the dimension M is given by $M = (N_x+1)N_y + (N_y+1)N_x$. Assuming a dimensional partitioning of flux variables, Figure 1 gives the inequality-constraint matrix $\mathbf{A}$ for a structured grid with $N_x = 3$ and $N_y = 4$.

We implement OBT, FCRT and LVLT in Matlab™ and rely on vectorized arithmetic and efficient data structures for the storage of mesh data. We remark that such implementation can rival the computational performance of mathematically equivalent Fortran code, see [4, Sec. 6.4]. The global linear systems involving the matrices $\mathbf{A}\mathbf{A}^{\mathsf{T}}$, see Part 2 [2], are solved using sparse Cholesky and/or LU factorizations.

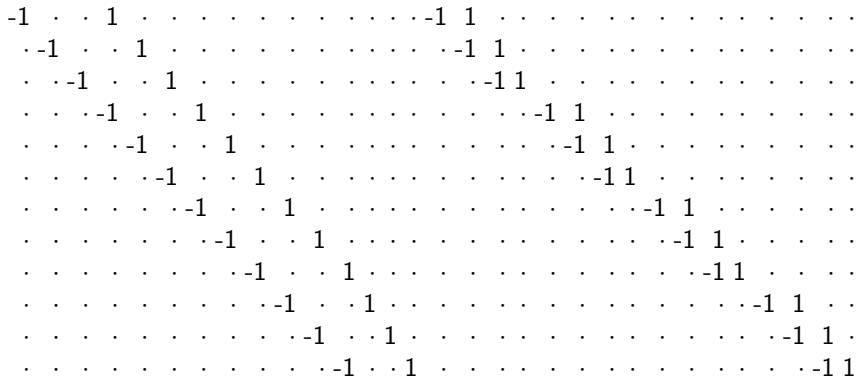

Figure 1. The inequality-constraint matrix $\mathbf{A}$ for a structured grid with $N_x = 3$ and $N_y = 4$. For compactness, zeros have been replaced by dots.

3 Results

In this section, we study OBT, FCRT and LVLT on a number of numerical examples similar to those given by LeVeque [5]. We do not give a direct comparison to all methods implemented by LeVeque, such as the transport with superbee limiting, however, we provide analogous figures so that an interested reader can make a qualitative comparison.

In Example 1, we compute a solid body rotation of a smooth hump in a circular flow. Specifically, we define the initial hump to be

$$q(x, y, 0) = \frac{1}{4}(1 + \cos(\pi r(x, y))),$$

where

$$r(x, y) = \min\{\sqrt{(x - x_0)^2 + (y - y_0)^2}, r_0\}/r_0,$$

and define the rotating flow as

$$u = -(y - 1/2), \qquad v = (x - 1/2).$$

In order to test the accuracy of the methods, we set $r_0 = 0.15$, $x_0 = 25$, and $y_0 = 0.5$ and compute the solution over the domain $[0, 1] \times [0, 1]$. In order to insure that we rotate the hump one complete revolution, we use $\lfloor 8(2\pi n) \rfloor$ time steps, where n denotes the size of the computational grid in both the x and y directions, resulting in a maximum CFL number of about 1/8. We summarize the amount of error and the convergence rates for each method in Table 1.

LVLT							
#cells	#remaps	L_2 err	L_1 err	L_∞ err	L_2 rate	L_1 rate	L_∞ rate
80×80	4021	5.85e-03	1.29e-03	7.36e-02	—	—	—
100×100	5026	4.00e-03	8.88e-04	5.08e-02	1.70	1.67	1.66
120×120	6031	2.94e-03	6.59e-04	3.78e-02	1.69	1.65	1.64
140×140	7037	2.35e-03	5.30e-04	2.97e-02	1.64	1.60	1.62
FCRT							
#cells	#remaps	L_2 err	L_1 err	L_∞ err	L_2 rate	L_1 rate	L_∞ rate
80×80	4021	5.66e-03	1.24e-03	5.42e-02	—	—	—
100×100	5026	3.89e-03	8.63e-04	3.66e-02	1.68	1.62	1.76
120×120	6031	2.85e-03	6.45e-04	2.57e-02	1.69	1.61	1.84
140×140	7037	2.29e-03	5.21e-04	1.98e-02	1.63	1.56	1.82
OBT							
#cells	#remaps	L_2 err	L_1 err	L_∞ err	L_2 rate	L_1 rate	L_∞ rate
80×80	4021	6.15e-03	1.40e-03	5.71e-02	—	—	—
100×100	5026	4.11e-03	9.38e-04	3.76e-02	1.81	1.81	1.88
120×120	6031	2.95e-03	6.83e-04	2.60e-02	1.82	1.78	1.94
140×140	7037	2.33e-03	5.46e-04	1.98e-02	1.75	1.70	1.91

Table 1. Errors and convergence rate estimates after applying a variety of methods to Example 1. In this example, we rotate a smooth hump one revolution for a number of time steps specified in the table above.

As we can see from Table 1, the three methods give similar numerical results. OBT exhibits slightly better asymptotic convergence than FCRT or LVLT. The lack of perfect second order convergence may be attributed to the time discretization.

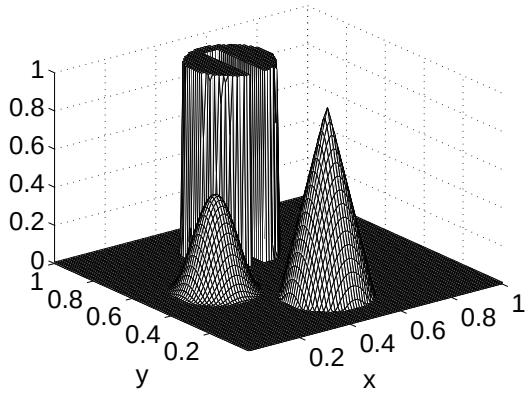

Figure 2. Initial data for the solid body rotation tests.

In Example 2, we use the same rotating flow from above, but rotate a combination of a smooth hump, cone, and slotted disk shown in Figure 2. In each test,

we use the same smooth hump as above and use a cone and disk with radius 0.15 centered at $(0.5, 0.25)$ and $(0.5, 0.75)$, respectively. As before, we rotate the objects one full revolution, but use $\lfloor 2\pi n \rfloor$ time steps, where n denotes the size of the computational grid. In our first test, we summarize a comparison of the runtime required for each method in Table 2. In addition, we give a qualitative comparison between OBT and FCRT in Figures 3 and 4, respectively.

Grid Size	40×40	80×80	160×160	320×320
OBT	4.00	34.21	422.85	4108.27
FCRT	0.83	5.48	45.27	375.90
LVLT	0.89	5.84	45.38	362.65

Table 2. Comparison of the computational cost for Example 2 using OBT, FCRT, and LVLT, in seconds (total wall-clock time). In each test, we compute one revolution of the smooth hump, cone, and slotted disk for increasing grid resolutions. For the number of time steps, we use $\lfloor 2\pi n \rfloor$ where n denotes the grid size. The computational cost of OBT is on average 10 times higher than the cost of FCRT or LVLT.

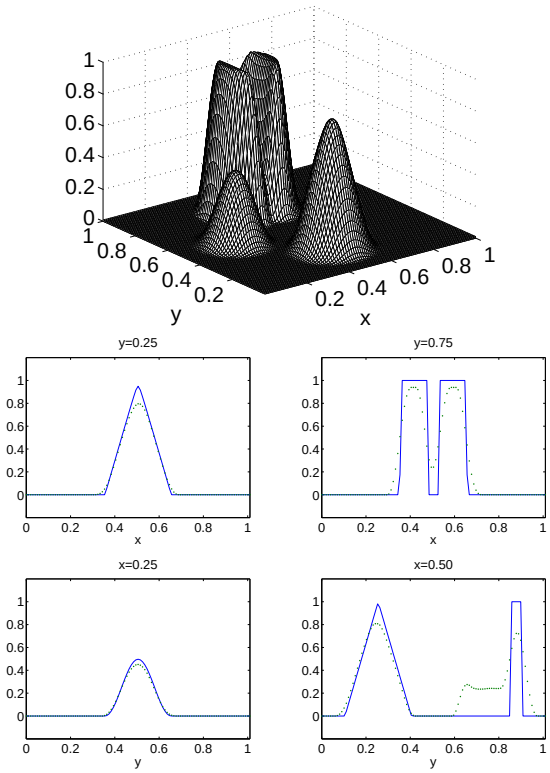

Figure 3. The result of applying OBT to Example 2 for one revolution (628 time steps) on a 100×100 grid. We show four different cross sections of the solution along with the surface plot. The solid lines denote the true solution.

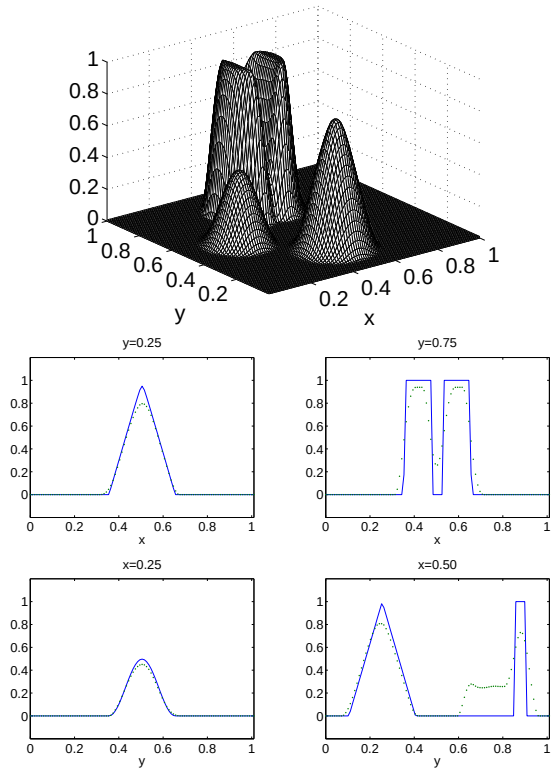

Figure 4. The result of applying FCRT to Example 2 for one revolution (628 time steps) on a 100×100 grid. We show four different cross sections of the solution along with the surface plot. The solid lines denote the true solution.

In terms of computational cost, we note that OBT is on average 10 times more expensive than FCRT or LVLT. However, as with FCRT and LVLT, the computational cost of OBT scales linearly with mesh refinement. Improvements in the optimization algorithm, in particular the solution of global linear systems, are possible and will lead to a reduction in the computational cost. In terms of qualitative results, the methods are comparable, see Figure 3 for OBT and Figure 4 for FCRT (LVLT not shown).

The purpose of Example 3 is to examine the robustness of the methods. In particular, we rotate the slotted disk from Example 2 about its axis for one complete revolution. In order to accomplish this, we center the slotted disk at $(0.5, 0.5)$ and use the same rotating flow from above. We compute the result of rotation for $2\pi(1/\Delta t)$ time steps where we use an initial $\Delta t = 1/100$. Then, we slowly increase the size of the time step until the L_1 error doubles from its initial value computed at $\Delta t = 1/100$. We give the result of this test on a 100×100 grid in Table 3 and a qualitative depiction of OBT in Figure 5.

	$1/\Delta t$=100 CFL=1.00	$1/\Delta t$=62 CFL=1.60	$1/\Delta t$=61 CFL=1.62	$1/\Delta t$=45 CFL=2.20	$1/\Delta t$=44 CFL=2.25	$1/\Delta t$=19 CFL=5.50	$1/\Delta t$=18 CFL=5.21
OBT	**2.14e-02**	**2.37e-02**	**2.38e-02**	**2.60e-02**	**2.62e-02**	**4.02e-02**	4.36e-02
FCRT	**1.97e-02**	**2.19e-02**	**2.21e-02**	**3.00e-02**	6.00e+06	9.45e+38	1.83e+40
LVLT	**2.14e-02**	**2.36e-02**	8.15e-01	3.47e+54	2.85e+56	2.83e+79	6.23e+77

Table 3. L_1 errors in OBT, FCRT and LVLT for Example 3. In this example, we rotate a slotted disk centered at the point (0.5,0.5) one revolution ($\lfloor 2\pi(1/\Delta t)\rfloor$ time steps). Our goal is to determine the largest time step for which the error measured in the L_1 norm doubles given a baseline with $\Delta t = 1/100$. Results that are better than this error bound are given in bold. FCRT and LVLT clearly exhibit numerical breakdown at CFL numbers larger than 2.25. In contrast, OBT remains stable.

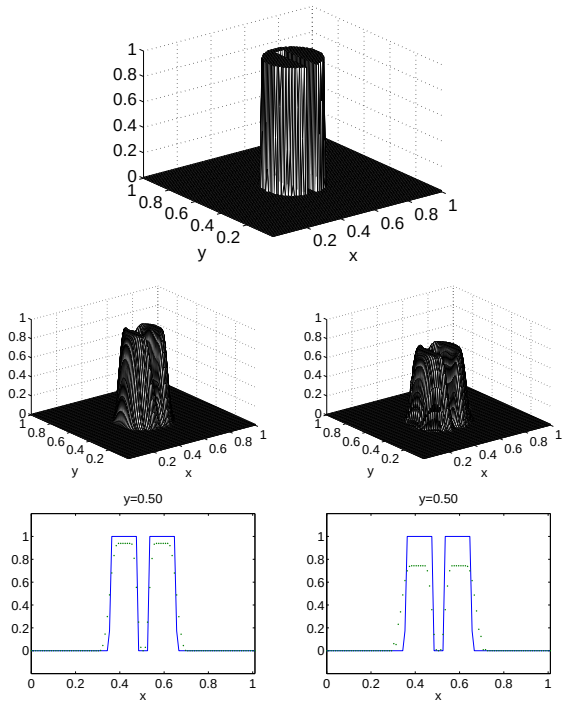

Figure 5. The result of applying OBT to Example 3. In this example we rotate a slotted disk centered at the point (0.5,0.5) one revolution. In the above plots, we show the original disk on top. On the bottom, we show surface and cross section plots of the disk after one revolution where $\Delta t = .01$ (CFL=1.00) on the left and $\Delta t = 0.047$ (CFL=4.7) on the right.

As we can see, OBT exhibits a level of robustness that far exceeds FCRT and LVLT. In fact, OBT produces qualitatively reasonable results for CFL numbers of about 5, while both FCRT and LVLT break down numerically at CFL numbers beyond 2.25.

4 Conclusion

In this paper we applied the framework for optimization-based transport, formalized in [1, 2], to benchmark transport problems presented in [5].

Numerical comparisons with transport schemes based on incremental flux-corrected remap and incremental linear flux reconstruction with van Leer limiting demonstrate that optimization-based transport is a competitive alternative. In particular, while computationally more expensive due to the solution of a globally constrained optimization problem, optimization-based transport is shown to be more accurate asymptotically and significantly more robust.

Future work includes performance optimizations and applications to the transport of systems.

References

1. Bochev, P., Ridzal, D., Young, D.: Optimization–based modeling with applications to transport. Part 1. Abstract formulation. In Lirkov, I., Margenov, S., Wasniewski, J., eds.: Proceedings of LSSC 2011. Springer Lecture Notes in Computer Science (Submitted 2011)
2. Young, J., Ridzal, D., Bochev, P.: Optimization–based modeling with applications to transport. Part 2. Optimization algorithm. In Lirkov, I., Margenov, S., Wasniewski, J., eds.: Proceedings of LSSC 2011. Springer Lecture Notes in Computer Science (Submitted 2011)
3. Dukowicz, J.K., Baumgardner, J.R.: Incremental remapping as a transport/advection algorithm. Journal of Computational Physics **160**(1) (2000) 318 – 335
4. Bochev, P., Ridzal, D., Scovazzi, G., Shashkov, M.: Formulation, analysis and numerical study of an optimization-based conservative interpolation (remap) of scalar fields for arbitrary lagrangian-eulerian methods. Journal of Computational Physics **In press** (2011)
5. LeVeque, R.J.: High-resolution conservative algorithms for advection in incompressible flow. SIAM Journal on Numerical Analysis **33**(2) (1996) 627–665
6. Liska, R., Shashkov, M., Váchal, P., Wendroff, B.: Optimization-based synchronized flux-corrected conservative interpolation (remapping) of mass and momentum for arbitrary Lagrangian-Eulerian methods. J. Comput. Phys. **229** (March 2010) 1467–1497
7. Margolin, L.G., Shashkov, M.: Second-order sign-preserving conservative interpolation (remapping) on general grids. J. Comput. Phys. **184**(1) (2003) 266–298