LA-UR-02-6395

# Efficient Second-Order Bound-Preserving Remapping (Conservative Interpolation)

Milan Kucharik

Faculty of Nuclear Sciences and Physical Engineering,

Czech Technical University in Prague,

Břehová 7, 115 19 Prague 1, Czech Republic

kucharik@karkulka.fjfi.cvut.cz

Mikhail Shashkov

Theoretical Division, T-7, Los Alamos National Laboratory

MS-B284, Los Alamos, NM 87545

shashkov@lanl.gov

Burton Wendroff

Theoretical Division, T-7, Los Alamos National Laboratory

MS-B284, Los Alamos, NM 87545

bbw@lanl.gov

Subject Classification Numbers: 65N05, 65N10, 65N15, 80A20

Keywords: Conservative Interpolation, Remapping, ALE Methods

# Abstract

In this paper we describe an efficient, second-order accurate, bound-preserving remapping algorithm. The algorithm is based on reconstruction, approximate integration and mass redistribution. We demonstrate our new algorithm on a series of numerical examples.


# 1 Introduction - Statement of the Problem

Consider that we have a mesh that is a tessellation of some region $\Omega$ by a collection of cells $C_i$ $i = 1, ..., L$, that is, the cells have disjoint connected interiors and their union is $\Omega$. This we will call the old mesh.

Next, suppose there is a new tessellation of the region with a different collection of cells $\tilde{C}_i$ $i = 1, ..., L$, such that each new cell $\tilde{C}_i$ is obtained by a small displacement of the vertices of the old cell $C_i$ by some displacement vector field $\delta$.

For example, in the context of an Arbitrary Lagrangian-Eulerian methods the old would be the result of the Lagrangian step and the new mesh would be the result of mesh modification (rezoning) (see, for example, [5]).

The situation is that the only data that the computation has at hand are, for each cell $C_i$, the mass for some density function $\rho(\mathbf{x})$, said mass being

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

with corresponding mean value

$$\bar{\rho}_i = \frac{m_i}{V(C_i)},$$

where $V(C_i) > 0$ is the volume of $C_i$.

The conservative interpolation (remapping) problem is to compute the masses $\tilde{m}_i$ and mean values of the new grid, but since the density distribution is unknown we can only approximate them. We will denote these approximate values by $\tilde{m}_i^*$.

This approximation will have to satisfy at least the following two constraints:

I). Total mass should be conserved.

II). Exactness for a global linear function. If $\rho(\mathbf{x})$ happens to be a linear function on $\Omega$ then the new masses $\tilde{m}_i^*$ should exactly agree with the masses of that function for the new cells.

A possible procedure for accomplishing this has two stages. First, one constructs a new distribution $\hat{\rho}(x)$ such that

$$\int_{C_i} \hat{\rho}(\mathbf{x})\, dV = m_i$$

and if $\rho(\mathbf{x}))$ is a global linear function then $\hat{\rho}(\mathbf{x}) \equiv \rho(\mathbf{x})$. Usually $\hat{\rho}(\mathbf{x})$ is a discontinuous piece-wise linear (linear on each cell $C_i$) function, whose local gradients have been limited in some way to satisfy additional conditions such as weak monotonicity.

For the second stage the reconstructed function $\hat{\rho}(\mathbf{x})$ is exactly integrated over the new cells. This requires finding the intersection of each new cell with the old ones. Finding the intersections is easy in one dimension, doable but computationally very expensive in two dimensions, and not practical in three dimensions. We will call any such two stage method “exact”.

The goal of this paper is to present a new efficient, conservative, local-bound-preserving, second-order accurate algorithm for remapping which does not require finding intersections between the cells of the old and new mesh.

In the first stage of our new remapping method we also use a discontinuous piecewise linear $\hat{\rho}(\mathbf{x})$, but without gradient limiters.

In the second stage, the new mass is written in flux form, that is, the mass of each new cell will be set to the mass of the corresponding old cell plus terms which define exchange of mass with nearest neighbors (fluxes). Using the flux form trivially guarantees conservation of total mass. In contrast to Dukowicz and Baumgardner [3], who use exact integration to compute these fluxes, we use a quadrature, which does not require finding the intersections of new and old cells.

It may happen that the mean densities in the new cells will be out of local bounds, for example, they can be negative even if the original means were all positive. The cause of this can be: either use of unlimited gradients in the reconstruction, approximate integration, or both.

To fix out of bound mean densities we introduce a third stage - repair. In this stage masses of new cells are redistributed in such a way that modified new means are in the range of local bounds, and they are changed as little as possible compared with those obtained as a result of the second stage.

In next two sections we will describe the algorithm in some detail and will give some numerical examples. In those examples for the purpose of comparison we also present some results using a particular exact method, that is, the method of limited reconstruction of Barth and Jespersen [1].

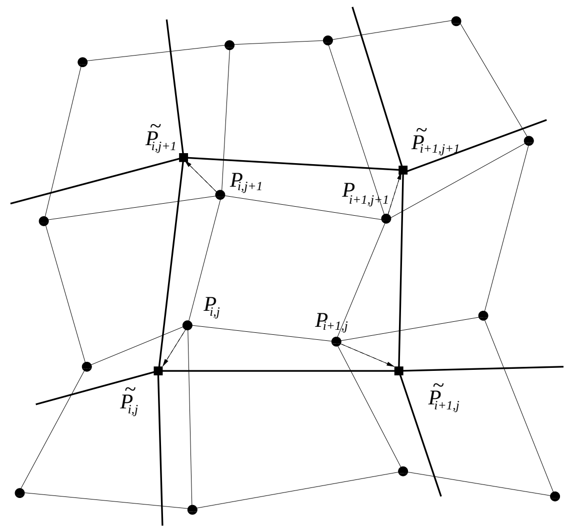



Figure 1: Old (thin lines and solid circles) and New Mesh (bold lines and solid squares); Swept Regions.

## 2 Remapping Algorithm

The method we are describing is by no means restricted to a logically rectangular grid or to 2D, but we explain it in those terms.

The grid points or vertices of a logically rectangular grid have the vertices $P_{i,j} = (x_{i,j}, y_{i,j}),\ i = 1, \cdots, m,\ j = 1, \cdots, n$. The cells $C_{i+\frac{1}{2},j+\frac{1}{2}}$ are the quadrilaterals formed by the four vertices $P_{i,j}$, $P_{i+1,j}$, $P_{i+1,j+1}$, $P_{i,j+1}$ (see Fig. 1).

## 2.1 Reconstruction - First Stage

The first task is to construct the function $\hat{\rho}(x,y)$. We define a piecewise linear function $\hat{\rho}(x,y)$ as

$$\hat{\rho}(x,y) =$$

$$\hat{\rho}_{i+\frac{1}{2},j+\frac{1}{2}} = \bar{\rho}_{i+\frac{1}{2},j+\frac{1}{2}} + g^x_{i+\frac{1}{2},j+\frac{1}{2}}\,(x - \bar{x}_{i+\frac{1}{2},j+\frac{1}{2}}) + g^y_{i+\frac{1}{2},j+\frac{1}{2}}\,(y - \bar{y}_{i+\frac{1}{2},j+\frac{1}{2}}),$$

$$(x,y) \in C_{i+\frac{1}{2},j+\frac{1}{2}}$$

where the centroid $(\bar{x}_{i+\frac{1}{2},j+\frac{1}{2}}, \bar{y}_{i+\frac{1}{2},j+\frac{1}{2}})$ is

$$\bar{x}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{V(C_{i+\frac{1}{2},j+\frac{1}{2}})} \int_{C_{i+\frac{1}{2},j+\frac{1}{2}}} x\,dV\ ,$$

$$\bar{y}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{1}{V(C_{i+\frac{1}{2},j+\frac{1}{2}})} \int_{C_{i+\frac{1}{2},j+\frac{1}{2}}} y\,dV,$$

and these can be computed exactly for polygons, as can the integral of any linear function (see, for example, [4]). The gradient $(g^x_{i+\frac{1}{2},j+\frac{1}{2}}, g^y_{i+\frac{1}{2},j+\frac{1}{2}})$ can be obtained from any difference approximation to the gradient of $\rho(x,y)$ using neighboring mean values as long as it is exact if the means are the mean values of a global linear function (for examples of such approximation see [4]).

## 2.2 Flux Form and Swept Region Quadrature - Stage Two

### 2.2.1 Flux Form

We now look for a finite difference equation defining the new masses $\tilde{m}^*_{i+\frac{1}{2},j+\frac{1}{2}}$ in terms of the old masses $m_{i+\frac{1}{2},j+\frac{1}{2}}$, and mass exchange with neighboring cells,

$$\tilde{m}^*_{i+\frac{1}{2},j+\frac{1}{2}} = m_{i+\frac{1}{2},j+\frac{1}{2}} + \mathcal{F}_{i+1,j+\frac{1}{2}} - \mathcal{F}_{i,j+\frac{1}{2}} + \mathcal{F}_{i+\frac{1}{2},j+1} - \mathcal{F}_{i+\frac{1}{2},j}\,. \tag{1}$$

Here $\mathcal{F}$ are edge fluxes, which are defined following ideas from [4] and will be described in next subsection.

#### 2.2.2 Swept Regions

Consider an oriented edge $F_{i,j+\frac{1}{2}} = \{P_{i,j+1}, P_{i,j}\}$. This edge is common to cells $C_{i+\frac{1}{2},j+\frac{1}{2}}$ and $C_{i-\frac{1}{2},j+\frac{1}{2}}$. The displacement vector field $\delta$ moves $P_{i,j}$ to its new position $\tilde{P}_{i,j}$ and $P_{i,j+1}$ to $\tilde{P}_{i,j+1}$, thereby forming the oriented quadrilateral $\delta F_{i,j+\frac{1}{2}} = (P_{i,j}, \tilde{P}_{i,j}, \tilde{P}_{i,j+1}, P_{i,j+1})$ (see Fig. 1). It is important to note that this quadrilateral can have self-intersections. This quadrilateral was called a fluxing area in [3], swept region in [4], but the notion seems to have been introduced earlier by Collela [2]. Similarly, for an oriented edge $F_{i+\frac{1}{2},j} = \{P_{i,j}, P_{i+1,j}\}$ we define the swept region as $\delta F_{i+\frac{1}{2},j} = (P_{i,j}, P_{i+1,j}, \tilde{P}_{i+1,j}, \tilde{P}_{i,j})$.

We next summarize the development in [4]. First of all, the critical function of the swept region is its *signed* area $V(\delta F_{i,j+\frac{1}{2}})$, which in turn depends on the ordering of its vertices (orientation of the boundary). It is introduced by expressing the area in terms of a line integral as follows:

$$V(\delta F_{i,j+\frac{1}{2}}) = \int_{\delta F_{i,j+\frac{1}{2}}} dV = \oint_{\partial(\delta F_{i,j+\frac{1}{2}})} x\, dy\,. \tag{2}$$

Because of the chosen orientation, for situation shown in Fig. 1, $V(\delta F_{i,j+\frac{1}{2}}) < 0$.

As it has been shown in [4] we can introduce *signed* integration of any polynomial function over a polygon by reducing it to a line integral similar to

(2). The sign of course will depend on the chosen orientation. In the rest of the paper each time we integrate a linear function over a polygon we always will assume that it is signed integration.

We define flux $\mathcal{F}_{i,j+1/2}$ in (1) as follows

$$\mathcal{F}_{i,j+\frac{1}{2}} = \int_{\delta(F_{i,j+\frac{1}{2}})} \hat{\rho}_{i,j+\frac{1}{2}}(x,y)\, dV\,,$$

where

$$\hat{\rho}_{i,j+\frac{1}{2}}(x,y) = \begin{cases} \hat{\rho}_{i+\frac{1}{2},j+\frac{1}{2}}(x,y)\,, & V(\delta F_{i,j+\frac{1}{2}}) \geq 0\,, \\ \hat{\rho}_{i-\frac{1}{2},j+\frac{1}{2}}(x,y)\,, & V(\delta F_{i,j+\frac{1}{2}}) < 0\,, \end{cases} \quad .$$

In words, in the swept region the integrand is taken entirely from the cell on one side or the other of the edge, depending on the sign of the swept area. In the case shown in Fig. 1 $V(\delta F_{i,j+\frac{1}{2}}) < 0$ and we use reconstructed linear function belonging to the left side of the edge ($\hat{\rho}_{i-\frac{1}{2},j+\frac{1}{2}}(x,y)$) because most of the swept region lies inside cell $C_{i-\frac{1}{2},j+\frac{1}{2}}$.

Similar formulas are used for the other family of edges.

Formula (1) can be considered as a conservative quadrature for computing the new mass $\tilde{m}_{i+\frac{1}{2},j+\frac{1}{2}}$. In fact, in the situation shown in Fig. 1 the new cell $\tilde{C}_{i+\frac{1}{2},j+\frac{1}{2}}$ is the union of the old cell $C_{i+\frac{1}{2},j+\frac{1}{2}}$ and the four swept regions corresponding to its edges. Formula (1) means that the new mass is approximated by the integral over the old cell (which gives $m_{i+\frac{1}{2},j+\frac{1}{2}}$) plus integrals over the swept regions, where the choice of which piece of the reconstructed function to use is based on the sign of the volume of corresponding the swept area.

## 2.3 Conservative Mass Re-Distribution - Stage Three

The quantity which one is remapping may have specific physical meaning. For example, the given means might be derived from a concentration lying between zero and one, and for the remapped means to be physically correct they have to be in the same range. At this point our algorithm may create values out of range, especially for non-smooth $\rho$. In this section we describe a procedure based on mass redistribution which locally adjusts the out of range values to be within the bounds.

By way of illustration, suppose all the initial means are nonnegative and that we would like the same to be true of the remapped means. Note that if the initial density distribution is linear and nonnegative over the whole domain $\Omega$ then by constraint II the new means will be nonnegative, so no correction will be needed. Now, suppose that the mass of some new cell $\tilde{C}_{i+\frac{1}{2},j+\frac{1}{2}}$, computed using ((1)), is negative. Let $m^- = \tilde{m}^*_{i+\frac{1}{2},j+\frac{1}{2}} < 0$. Look at the eight immediate neighbors of the cell and compute the total mass $M$ of those neighbors having positive masses. If $M \geq |m^-|$, we can reduce each of those positive masses $m^+$ by $m^+\,|m^-|/M$, and then set $\tilde{m}^*_{i+\frac{1}{2},j+\frac{1}{2}} = 0$, thereby conserving mass. If $M < |m^-|$ then we cannot correct the negative mass from the eight neighbors. In this case one can try larger neighborhoods. Eventually by extending neighborhoods (in the limiting case it will be the entire mesh) we will be able to correct negative value because total mass is conserved and is positive. In all examples which we have run we were able to fix the problem by using at most a $7 \times 7$ neighborhood. A similar procedure can be done for any global or local bound. For example, we could take as upper and lower bounds the maximum and minimum of neighboring old cell values. In fact, in the numerical examples presented in the next section such local bounds were imposed.

## 3 Tests

Our algorithm can be indicated by the key words “unlimited, swept, repair”, so we will call it the USR algorithm. For comparison purposes we also present results for the following similar algorithms: USN, meaning unlimited, swept, no repair; and as a reference we use BE, meaning Barth with exact integration.

In our numerical tests we will do a sequence of 10 remappings in order to test the cumulative error accumulation. The first grid is uniform in the unit square $[0,1] \times [0,1]$. Each successive problem consists of a remapping from the data on the previous grid to the next grid in the sequence, that being a small random perturbation of the uniform grid, Fig. 2.

First, we remapped a smooth quartic $(x - .5)^4 + (y - .5)^4$ , with grids $100 \times 100$, $200 \times 200$, $400 \times 400$ and we observed second-order accuracy for the new method (USR).

The next test remaps a cubical pyramid, Fig. (3), on a $75 \times 75$ grid, using the four methods indicated above. Note that without any limiters or mass re-distribution, i.e., the USN method, some values will go out of range and there will be oscillations. This is seen in Fig. (4b), where even negative values appear. The $L_1$ norm of the error in this case is $9.04 \times 10^{-3}$.

Now we can see in Fig. (5) that this problem is fixed by our method (USR), which differs from (USN) only by adding repair. The $L_1$ norm of the

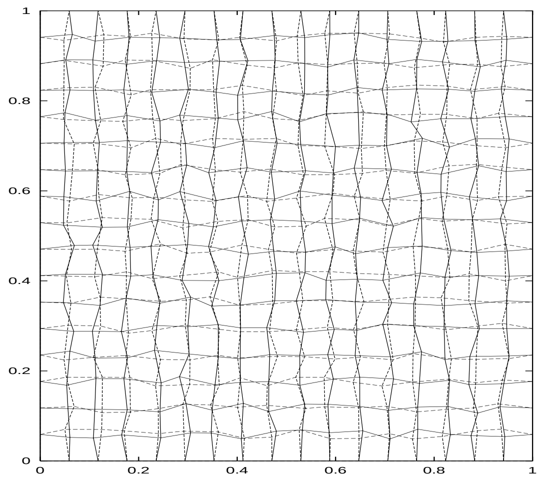
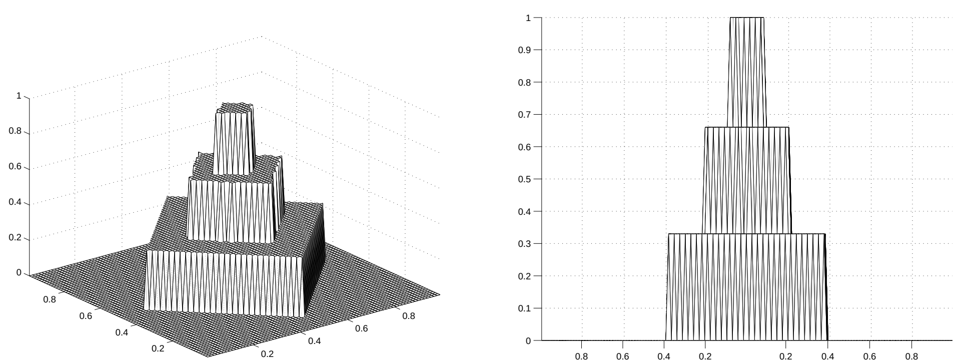
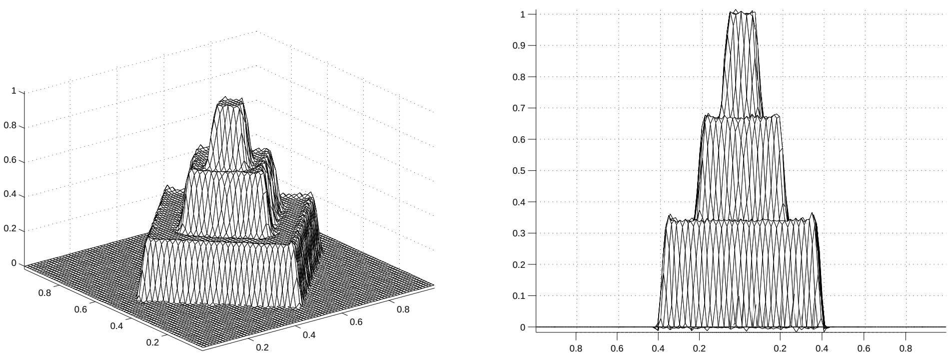



Figure 2: Two consecutive grids, $17 \times 17$.




a) b)

Figure 3: Initial Pyramid: a) Full view, b) Side view



a) b)

Figure 4: Unlimited, swept, no repair - (USN): a) Full view, b) Side view

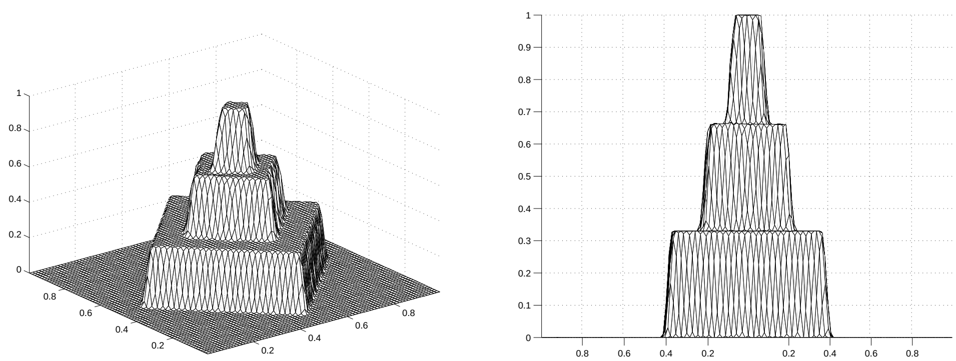
error in this case is $8.27 \times 10^{-3}$.

For comparison the result of the exact method (BE) is shown in Fig. (6). The $L_1$ norm of the error in this case is $9.52 \times 10^{-3}$. As expected, the values are within bounds, but also the shape appears to have been rounded off more than with USR.

We also used our USR algorithm for a discontinuous quadratic pagoda shape, Figs.(7a) and (7b). The $L_1$ norm of the error in this case is $9.19 \times 10^{-4}$, while BE for this problem had an error of $9.69 \times 10^{-4}$.

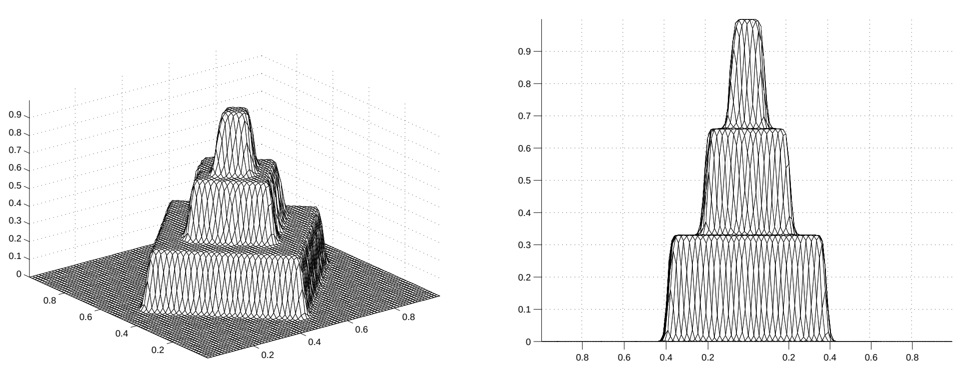
# 4 Conclusion

The conclusion from these runs is that our algorithm has the smallest $L_1$ error (an unexpected result) and is less dissipative. Many other tests were performed on these and other problems with more remappings and different movement of the grids. All confirm that the error with USR is most often



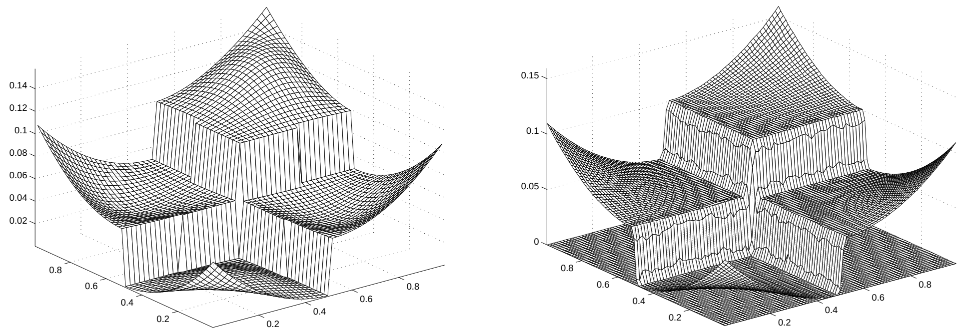

a) b)

Figure 5: Unlimited, swept, repair - USR: a) Full view, b) Side view




a) b)

Figure 6: BE exact method: a) Full view, b) Side view




a) b)

Figure 7: Piecewise quadratic: a) Initial, b) After 10 maps

smallest or else differs very little from the smallest; the ability to achieve this without the expense of computing intersections of the old and new cells is a clear advantage of the new method.

# Acknowledgment

This work was performed under the auspices of the US Department of Energy at Los Alamos National Laboratory, under contract W-7405-ENG-36. The authors acknowledge the partial support of the DOE/BES Program in the Applied Mathematical Sciences and the Laboratory Directed Research and Development program (LDRD). M. Shashkov also acknowledges the partial support of DOE's Accelerated Strategic Computing Initiative (ASCI). M. Kucharik also acknowledges the partial support of Czech Grant Agency grant 201/00/0586

The authors thank L. Margolin, B. Swartz, R. Liska and M. Berndt for

fruitful discussions and constructive comments.

## References

[1] T. J. Barth. Numerical methods for gasdynamic systems on unstructured meshes. In C. Rohde D. Kroner, M. Ohlberger, editor, *An introduction to Recent Developments in Theory and Numerics for Conservation Laws, Proceedings of the International School on Theory and Numerics for Conservation Laws*, Berlin, 1997. Lecture Notes in Computational Science and Engineering, Springer.

[2] P. Collela. Multidimensional upwind methods for hyperbolic conservation laws. *J. Comp. Phys.*, 87:171–200, 1990.

[3] J. Dukowicz and J. Baumgardner. Incremental remapping as a transport/ advection algorithm. *J. Comp. Phys.*, 160:318–335, 2000.

[4] L. Margolin and M. Shashkov, *Second-order sign-preserving remapping on general grids*, Technical Report LA-UR-02-525, Los Alamos National Laboratory, 2002, (http://cnls.lanl.gov/∼shashkov), To appear at Journal of Computational Physics.

[5] L. Margolin P. Knupp and M. Shashkov. Reference jacobian optimization-based rezone strategies for arbitrary lagrangian eulerian methods. *J. Comp. Phys.*, 176:93–128, 2002.