Divergence preserving reconstruction of the nodal components of a vector field from its normal components to edges

Shashkov, Mikhail Jurievich
Liska, Richard

Provided by the author(s) and the Los Alamos National Laboratory (2019-01-11).

To be published in: International Journal for Numerical Methods in Fluids

DOI to publisher's version: 10.1002/fld.4289

Permalink to record: http://permalink.lanl.gov/object/view?what=info:lanl-repo/lareport/LA-UR-15-23646

Disclaimer:
Approved for public release. Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the Los Alamos National Security, LLC for the National Nuclear Security Administration of the U.S. Department of Energy under contract DE-AC52-06NA25396. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 0000; **00**:1–13
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld

Divergence preserving reconstruction of the nodal components of a vector field from its normal components to edges

Richard Liska[1]* and Mikhail Shashkov[2]

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

SUMMARY

We have developed a new divergence preserving method for the reconstruction of the Cartesian components of a vector field from the orthogonal projection of a vector field to the normals to edges in 2D. In this method, discrete divergences computed from the nodal components and from the normal ones are *exactly* the same. Our new method consists of two stages. At the first stage, we use an extended version of the local procedure described in [*J. Comput. Phys.*, **139**:406–409, 1998] to obtain a “reference” nodal vector. This local procedure is exact for linear vector fields, however, the discrete divergence is not preserved. Then we formulate a constrained optimization problem, in which this reference vector plays the role of a target and the divergence constraints are enforced by using Lagrange multipliers. It leads to the solution of “elliptic” like discrete equations for the cell-centered Lagrange multipliers. The new global divergence preserving method is exact for linear vector fields. We describe all details of our new method and present numerical results which confirm our theory. Copyright © 0000 John Wiley & Sons, Ltd.

Received . . .

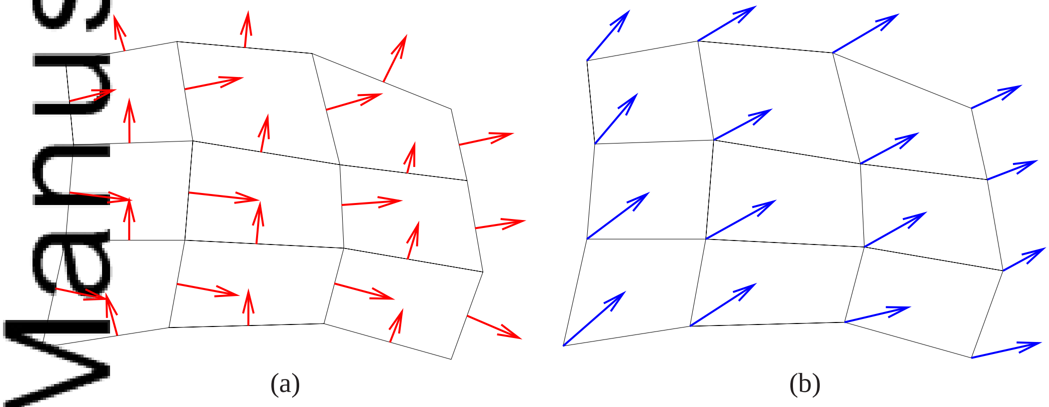
KEY WORDS: Lagrangian; hydrodynamics; vector interpolation; divergence preserving; vector representation; finite difference

1. INTRODUCTION

We continue to develop the foundation of the discrete vector and tensor calculus (DVTC) [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. From theoretical and practical points of view it is very important to investigate how different discrete representations of a vector field on the same mesh are related to each other. It is well known, that in continuum case the vector field $\mathbf{w}$ can be uniquely recovered from its divergence - $\mathbf{div}\,\mathbf{w}$ and curl - $\mathbf{curl}\,\mathbf{w}$ and appropriately specified boundary conditions. This observation suggests that one needs to compare the discrete divergence and the discrete curl for different representations of the vector field on the mesh. The quality of recovering one discrete component of the vector from another can be judged by how close are the discrete divergence

*Correspondence to: Faculty of Nuclear Sciences and Physical Engineering Czech Technical University in Prague Brehova 7 115 19 Prague 1 Czech Republic. E-mail: liska@siduri.fjfi.cvut.cz and curls to each other for the different representations. Reconstructed vector fields are often used in different parts of the algorithm to compute discrete analogs of differential operators such as divergence and curl. Therefore, to estimate the quality of the reconstruction, we need to evaluate not only the accuracy of the vector itself but also the accuracy of the approximate discrete **DIV** and **CURL** applied to the reconstructed vector field.

In [8], authors have introduced a new procedure for the reconstruction of the Cartesian components of a vector field at the nodes (see Fig. 1(b)) of a logically rectangular grid, when this vector field is given by its components normal to the edges of the grid cells (see Fig. 1(a)). This reconstruction procedure is exact for linear vector functions, which is critical for accurate computation of the discrete operators. However, in [8] authors have considered an algorithm only for the internal points of the mesh. Moreover, the method introduced in [8] is not divergence preserving, that is, the discrete divergence computed from the normal components is not the same as the discrete divergence computed from the normal components to the edges.


(a)

(b)

Figure 1. Two different representations of a vector field on the mesh – normal components to the edges (a) and vectors at the nodes (b).

There are many areas in which a mimetic reconstruction of vectors is very important [11]. The reconstruction of the Cartesian components of the vector field from its normal components appears naturally in the Lagrangian gas dynamics discretizations based on the Godunov method, where the normal component of the velocity vector on an edge between two cells is computed from the solution of a 1D Riemann problem, but the Cartesian components are needed at all nodes in order to compute the nodal motion [12, 13].

This class of methods has to satisfy the Geometric Conservation Law (GCL) [14, 15], which in the simplest case means that the volume change of the cell computed from the normal components of the velocity field using the volume evolution equation has to be the same as from the geometry of the cell, whose vertexes are moving with the nodal velocities. This is the same as requiring that the discrete divergences computed using two different representations of the vector field are the same.

Modern Godunov-like methods are trying to satisfy GCL by using some special procedure which involves nodal Godunov solver, [16, 17, 18, 19].

We have developed a new divergence preserving method for the reconstruction of the Cartesian components of the vector from the orthogonal projections of the vector field to the normals of the

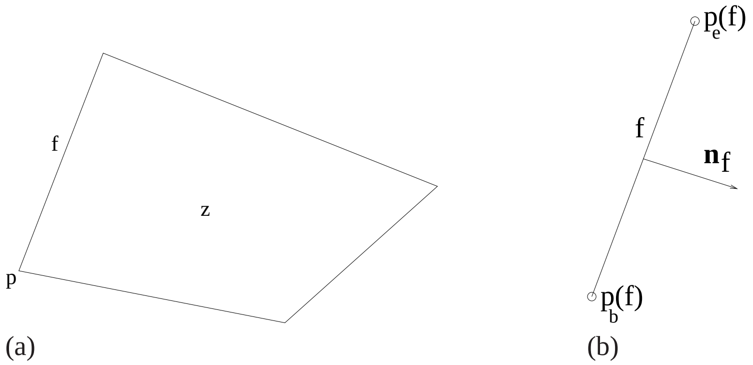


Figure 2. (a) Zone (cell), z; point (vertex, node), p; edge f. (b) Edge f - $\mathbf{p}_b(f)$ and $\mathbf{p}_e(f)$ are the beginning and end points of the edge. The oriented unit normal to the edge is $\mathbf{n}_f$.

edges. In this method, the discrete divergences computed from the nodal components and from the normal ones are *exactly* the same.

Our new method has two stages. At the first stage, we extend the local procedure described in [8] so, that it can also deal with the boundary nodes. The obtained nodal vector is called the “reference” vector. Then we formulate a constrained optimization problem, in which this reference vector plays the role of a target and the divergence constraints are enforced by using Lagrange multipliers. It leads to the solution of “elliptic” like discrete equations for the cell-centered Lagrange multipliers. The presented method is similar to Chorin’s projection method [20] in the context of the solution of incompressible Navier-Stokes equations. While in Chorin’s method one solves similar equations for the pressure to satisfy the divergence free conditions, in our case the divergence is arbitrary and its value comes from the given normal components of the vector field.

We describe all details of our new method and present numerical results which confirm our theory. The remainder of the paper is organized as follows. In Section 2, we introduce notations for the mesh elements and for the discrete operators. The local method of reconstruction of the nodal components of the vector field from its normal components to the edges is described in Section 3. The statement of the divergence preserving reconstruction is given in Section 4. The discrete equations, which one needs to solve for the divergence preserving reconstruction, are derived in Section 5. The normal boundary conditions for our methods are introduced in Section 6. Numerical tests are presented in Section 7.

2. NOTATIONS

The point of a mesh is denoted by $\mathbf{p} = (x, y)$, a cell (zone) is denoted by z, and the edge of the cell denoted by f - Fig.2(a). The edge (face) f is given by two points $\mathbf{p}_b(f)$ and $\mathbf{p}_e(f)$. The midpoint of the edge is given by $\mathbf{f} = (\mathbf{p}_b(f) + \mathbf{p}_e(f))/2$ and the oriented unit normal to the edge f is $\mathbf{n}_f$ - Fig. 2(b).

The theory below is presented for a general mesh consisting of polygons for which the formulas are more compact. At a few places and in the numerical tests we assume that our mesh is logically rectangular, so that all cells (zones) are quadrilaterals. The nodes are then numbered by indexes i, j and the cells by half-indexes $i + 1/2, j + 1/2$.

A vector field is denoted by $\mathbf{w} = (u,\, v)$. The nodal components of the vector are $\mathbf{w}_p = (u_p,\, v_p)$; the edge component of the vector is $w_f = (\mathbf{w},\, \mathbf{n}_f)$, where $\mathbf{n}_f = (n_f^x,\, n_f^y)$ is the unit normal to the edge f. The edge component of the vector can be defined either as a point value at the center of the edge $w_f = \mathbf{w}(\mathbf{f}) \cdot \mathbf{n}_f$ or as an integral average over the edge $w_f = \int_{\mathbf{f}} \mathbf{w} \cdot \mathbf{n}_f \; d\, s$.

The divergence acting from edges to zones is defined by

$$\left(\mathbf{DIV}^f\,\mathbf{w}\right)_z = \frac{\left(\mathbf{D}^f\,\mathbf{w}\right)_z}{V_z}, \quad \left(\mathbf{D}^f\,\mathbf{w}\right)_z = \sum_{f\in\mathcal{F}(z)} w_f\,S_f\,\psi_{f,z}, \tag{1}$$

where V_z is the area (volume) of the zone, S_f is the length of the edge and $\psi_{f,z}$ is either 1 or -1 depending on the orientation of the normal $\mathbf{n}_f$ with respect to the zone z (it is 1 when $\mathbf{n}_f$ is the outer normal of the zone z and -1 otherwise). $\mathcal{F}(z)$ is the set of all edges of the zone z.

The divergence acting from nodes to zones is defined by

$$(\mathbf{DIV}^p\,\mathbf{w})_z = \frac{(\mathbf{D}^p\,\mathbf{w})_z}{V_z}, \quad (\mathbf{D}^p\,\mathbf{w})_z = \sum_{f\in\mathcal{F}(z)} \left[\left(\frac{\mathbf{w}_{p_b(f)} + \mathbf{w}_{p_e(f)}}{2}, \mathbf{S}_{f,z}\right)\right], \tag{2}$$

where $\mathbf{S}_{f,z} = S_f\,\mathbf{n}_f \psi_{f,z}$.

3. LOCAL METHOD OF RECONSTRUCTION OF THE NODAL COMPONENTS OF A VECTOR FIELD FROM ITS NORMAL COMPONENTS TO EDGES

The local methods have been introduced in [8], where the authors reconstruct the nodal Cartesian components of the vector field from its normal components on a set of edges adjacent in some sense to the given node. This set of edges defines the stencil of some local interpolation operator. In [8], the authors have considered only a reconstruction for the internal points of the mesh. In this paper we extend this algorithm to all points, i.e., we include also boundary nodes.

The algorithm recovers not only the Cartesian components of the vector at the node $\mathbf{w}_p$, but also its gradient $\nabla\mathbf{w}_p$. The method is based on the solution of a local minimization problem. The local functional has the form

$$\Phi_p^L = \sum_{f\in\mathcal{S}(p)} ((\mathbf{w}_p + (\mathbf{f} - \mathbf{p}) \cdot \nabla\mathbf{w}_p, \mathbf{n}_f) - w_f)^2, \tag{3}$$

with $\mathcal{S}(p)$ being the stencil of this local operator. For the interior nodes the stencil $\mathcal{S}(p)$ includes all edges of all cells, which have the node $\mathbf{p}$ as their vertex. So, for the logically rectangular mesh, this stencil includes 12 edges of 4 cells meeting at point p - Fig. 3 (a). Now, by differentiating of the functional Φ_p^L with respect to two components of $\mathbf{w}_p$ and four components of $\nabla\mathbf{w}_p$, we obtain six linear equations for six unknown components of $\mathbf{w}_p$ and $\nabla\mathbf{w}_p$.

The situation for the points on the boundary is more complicated. Consider as an example a point $\mathbf{p}$ on the left boundary of the orthogonal mesh. If we included only 7 edges of two cells joining this point, then the y component v_p would be considered on 3 horizontal edges of these two cells and all three centers of these edges would have the same x coordinate. Thus we would not be able to approximate the x derivative of the v_p component (one component of the gradient $\nabla \mathbf{w}_p$) in any way. To avoid this problem, we extend the 7 edges stencil (now back on the logically rectangular mesh) by one more edge, namely by the logically horizontal edge going logically right from the edge which joins the two cells on the left boundary. In other words (enumerating nodes by (i, j) starting with 1), if our point $\mathbf{p}$ on the left boundary has logical indices $(1, j)$, then 7 original edges of the stencil would include three logically horizontal edges $(3/2, j-1), (3/2, j), (3/2, j+1)$ and four logically vertical edges $(1, j-1/2), (1, j+1/2), (2, j-1/2), (2, j+1/2)$ and the additional eighth edge would be $(5/2, j)$, see Fig. 3 (b). On the other boundaries (right, bottom, top) away from corners, we proceed in a similar way as on the left boundary.

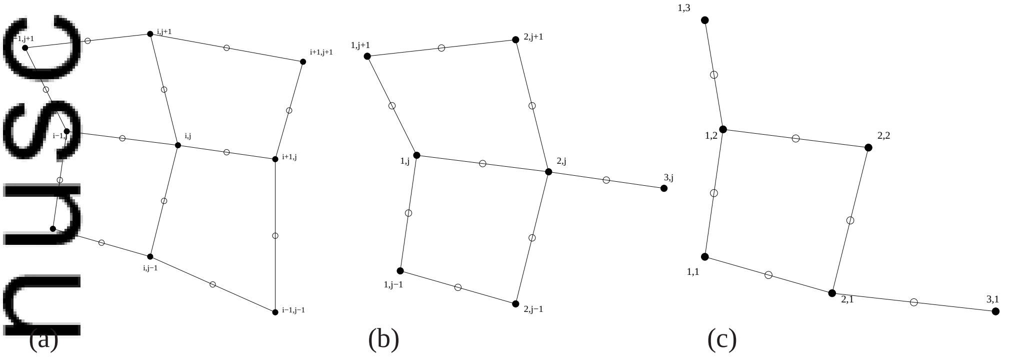


Figure 3. Stencils $\mathcal{S}(p)$ of the functional Φ_p^L: (a) internal points; (b) left boundary points; (c) the lower-left corner point. The midpoints of the edges included in the stencil are marked by a small empty circle.

At the lower left corner of the logically rectangular mesh (see Fig. 3 (c)), we naturally include in the stencil $\mathcal{S}(p)$ four edges of the corner cell and add to them two more boundary edges attached to the corner cell, The stencils at the other corners are analogous.

For each boundary point $\mathbf{p}$ we reconstruct here the full vector $\mathbf{w}_p$ from the components normal to the edges. We call this treatment of boundaries the free boundary conditions. Note, that the described procedure is not an extrapolation, it is an interpolation on the boundary. In the context of hydrodynamics, these boundary conditions are needed for the outer pressure boundary conditions. The normal velocity components on the boundary edges are obtained from the outer pressure and then interpolated to the boundary nodes. The additional interior edges (see Fig. 3 (b)) are included in the stencil of the functional (3) in order to achieve the second order convergence and for the reason explained above. In Section 6, we will consider also piston boundary conditions. The local method is exact for any linear vector field.

4. STATEMENT OF THE PROBLEM

Let us assume that the normal components w_f of the vector $\mathbf{w}$ are given and therefore $(\mathbf{D}^f\, \mathbf{w})_z$ is given. *The goal* is to construct the nodal components of the vector field $\mathbf{w}_p = (u_p,\, v_p)$ such, that its nodal divergence (2) is the same as its edge divergence (1).

Because both divergences have the same factor $1/V_z$, it is sufficient to require

$$\left(\mathbf{D}^p\,\mathbf{w}\right)_z = \left(\mathbf{D}^f\,\mathbf{w}\right)_z\,, \tag{4}$$

where RHS is the given number which is explicitly computed from the given w_f, using formula (1).

There is no unique solution to this problem, and it is not clear how to formally define the accuracy. We suggest to find the stationary point of the following global functional

$$\Phi(\mathbf{w}_p,\lambda_z) = \frac{1}{2}\sum_{p\in\mathcal{P}}(\mathbf{w}_p - \mathbf{w}_p^{\rm ref})^2 V_p + \sum_{z\in\mathcal{Z}}\left[\lambda_z\left((\mathbf{D}^p\mathbf{w}_p)_z - (\mathbf{D}^f w_f)_z\right)\right]\,, \tag{5}$$

where we use an accurate reference approximation $\mathbf{w}_p^{\rm ref}$ of the vector field in the nodes, obtained by the local algorithm described in Section 3, from the normal components w_f. This reference approximation in general does not satisfy the divergence constraints (4). By finding the stationary point of the functional Φ (5), we are constructing the approximation $\mathbf{w}_p$, which satisfies the divergence constraints and is as close to the reference approximation $\mathbf{w}_p^{\rm ref}$ as possible. In the definition of the functional (5), the first sum goes over all points (nodes) of the mesh and the second one over all cells of the mesh. The second sum is introduced to take into account the constraints (4) using the Lagrange multiplier approach. The Lagrange multiplier λ_z is defined at the zone z and V_p is the nodal area, which is taken to be one-fourth (for the interior point p of the logically rectangular mesh) of the sum of the zonal areas which share the point p.

The local method described in Section 3 is exact for linear vector fields. The reference approximation $\mathbf{w}_p^{\rm ref}$ thus satisfies the divergence constraints (4) and the functional (5) has a simple stationary point $\mathbf{w}_p = \mathbf{w}_p^{\rm ref}$. This means that the proposed global method is also exact for linear vector fields.

5. DISCRETE EQUATIONS

The extrema conditions for the functional Φ (5) with respect to λ_z give us back equations (4). The extrema conditions with respect to u_p, v_p are

$$d\Phi/du_p = 0\,, \quad d\Phi/dv_p = 0\,. \tag{6}$$

Let us first consider the derivatives of the first sum in (5), which gives

$$V_p(\mathbf{w}_p - \mathbf{w}_p^{\rm ref}).$$

To differentiate the second term in (5), it will be useful to give an explicit expression for the operator $\mathbf{D}^p$. First of all, one can rearrange terms in the expression (2) to get

$$(\mathbf{D}^p\,\mathbf{w})_z = \sum_{p\in\mathcal{P}(z)}\left[\left(\mathbf{w}_p, \frac{\mathbf{S}_{f^-(p),z}+\mathbf{S}_{f^+(p),z}}{2}\right)\right]\,, \tag{7}$$

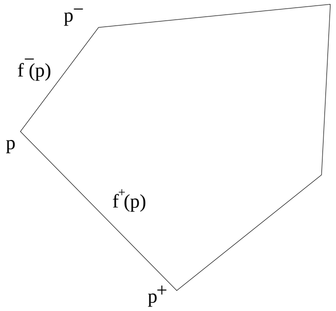


Figure 4. Vertex p and vertexes p^+ and p^-; edges $f^-(p)$ and $f^+(p)$;

where the subscripts $f^-(p)$ and $f^+(p)$ denote the edges which share the point p in the counter-clockwise order with respect to the zone z, see Fig. 4. If we introduce operators δ_x and δ_y, which act on a nodal function g as

$$\left(\delta_x\, g\right)_z = \frac{1}{2} \sum_{p \in \mathcal{P}(z)} \left(y_{p^+} - y_{p^-}\right) g_p\,, \quad \left(\delta_y\, g\right)_z = -\frac{1}{2} \sum_{p \in \mathcal{P}(z)} \left(x_{p^+} - x_{p^-}\right) g_p\,, \tag{8}$$

then the operator $\mathbf{D}^p$ in the form (7) can be written as

$$\left(\mathbf{D}^p\, \mathbf{w}\right)_z = \left(\delta_x\, u\right)_z + \left(\delta_y\, v\right)_z\ . \tag{9}$$

The part of the second sum of (5) containing $\mathbf{D}^p$ can be rearranged to summation over the nodes

$$\sum_{z \in \mathcal{Z}} \lambda_z (\mathbf{D}^p \mathbf{w}_p)_z = \frac{1}{2} \sum_{p \in \mathcal{P}} \sum_{z \in \mathcal{Z}(p)} \lambda_z \left[\left(y_{p^+(z)} - y_{p^-(z)}\right) u_p - \left(x_{p^+(z)} - x_{p^-(z)}\right) v_p \right],$$

which can be differentiated with respect to u_p and v_p, producing the expression

$$\left(\mathbf{G}^z\, \lambda\right)_p = \left(\mathbf{D}^p\right)^\dagger_p \lambda = \left\{ \begin{array}{l} \left(\delta^\dagger_x\, \lambda\right)_p \\ \left(\delta^\dagger_y\, \lambda\right)_p \end{array} \right. , \tag{10}$$

where for the zonal function λ the operators $\left(\delta^\dagger_x\, \lambda\right)_p$ and $\left(\delta^\dagger_y\, \lambda\right)_p$ are defined as

$$\left(\delta^\dagger_x\, \lambda\right)_p = \frac{1}{2} \sum_{z \in \mathcal{Z}(p)} \lambda_z \left(y_{p^+(z)} - y_{p^-(z)}\right)\,, \tag{11}$$

$$\left(\delta^\dagger_y\, \lambda\right)_p = -\frac{1}{2} \sum_{z \in \mathcal{Z}(p)} \lambda_z \left(x_{p^+(z)} - x_{p^-(z)}\right)\,, \tag{12}$$

where the vertexes $p^-(z)$, $p^+(z)$ are defined with respect to the zone z and are shown in Fig. 5. These expressions correspond to the contour around the node p presented in Fig. 5.

Now we are ready to write the equations (6) in the vector form

$$V_p(\mathbf{w}_p - \mathbf{w}^{\mathrm{ref}}_p) + \left((\mathbf{D}^p)^\dagger\, \lambda\right)_p = 0, \tag{13}$$

DOI: 10.1002/fld

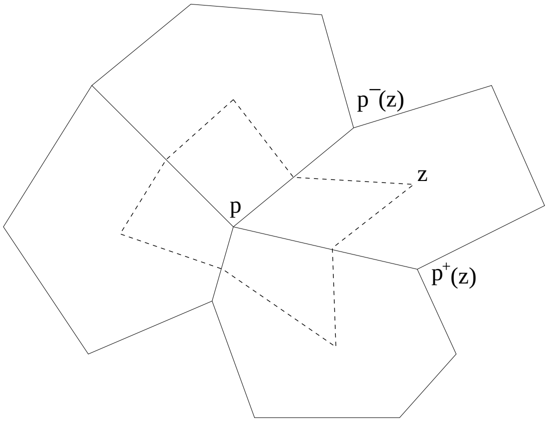


Figure 5. Contour for the $\mathbf{G}^z = (\mathbf{D}^p)^\dagger$ operator

which allows to express $\mathbf{w}_p$ as

$$\mathbf{w}_p = \mathbf{w}_p^{\mathrm{ref}} - \frac{1}{V_p}\left((\mathbf{D}^p)^\dagger \lambda_z\right)_p . \tag{14}$$

Substituting this into the divergence constraint at the cell z (4) results in

$$-\left[\mathbf{D}^p \frac{1}{V_p} (\mathbf{D}^p)^\dagger\right] \lambda_z = -\mathbf{D}^p \mathbf{w}_p^{\mathrm{ref}} + (\mathbf{D}^f w_f)_z, \tag{15}$$

which has to be valid for any zone z. After solving this global system for all λ_z, we obtain the desired $\mathbf{w}_p$ from (14). The equation (15) written in all zones z gives a system analogous to the system obtained for solving an elliptic equation with cell centered discretization of the scalar function and nodal discretization of the vector function (what is called the cell-node discretization in [6]).

The equation (14) has a different stencil and coefficients for the points which are strictly inside the mesh, on the boundary and at the corners of the mesh. Also the equation (15) has a different stencil and coefficients for the zones strictly inside (i.e., all edges of the zone are internal edges), the cells on the boundary and the corner cells.

Solving the linear (elliptic like) system (15) might be considered costly for the interpolation. We should however note, that the system has a symmetric, positive definite matrix and it is very well conditioned. Very efficient methods exist for such systems. In particular, for the tests we use the conjugate gradient method, which might be further speeded up by a preconditioner. The user of this method will need to decide if the gain obtained is worth the cost.

6. PISTON BOUNDARY CONDITIONS

Let us consider the case when the normal velocity to the boundary is given, which we call the piston boundary condition. In the context of the hydrodynamics we consider a generic piston, which can be either a standing or a moving wall. Namely, for these cases we generalize our methods to include boundary conditions on the normal components to the boundary. In this and the next section we

DOI: 10.1002/fld

consider only the simplest case of a rectangular domain with horizontal and vertical boundaries. On the horizontal boundary the y component v_P is given at boundary nodes $\mathbf{p}$, while on the vertical boundary the x component u_P is given at boundary nodes $\mathbf{p}$. At the corners of the rectangular domain both components are given.

For the local method, described in Section 3, the normal component of $\mathbf{w}_p$ and its tangential derivative (i.e., the x component u_p of $\mathbf{w}_p$ and its y derivative on the vertical boundaries or the y component v_p of $\mathbf{w}_p$ and its x derivative on the horizontal boundaries) are given at the boundary point p. Thus we differentiate the functional (3) according to the four remaining components of $\mathbf{w}_p$ and $\nabla\mathbf{w}_p$ to obtain four linear equations for these four remaining components. This system is being solved for these four components.

For the divergence preserving method described in Section 5, the normal component $\mathbf{w}_p^{\perp} = \mathbf{w}_p^{\mathrm{ref}\perp}$ of (14) at the boundary point p is known. Thus at the boundary point $\mathbf{p}$ only the parallel component of (14) is used when substituting into the divergence constraint (4) $(\mathbf{D}^p\,\mathbf{w})_{z(p)} = \left(\mathbf{D}^f\,\mathbf{w}\right)_{z(p)}$ at the boundary cells $z(p) \in \mathcal{Z}(p)$. For the normal component of (14) we use $\mathbf{w}_p^{\perp} = \mathbf{w}_p^{\mathrm{ref}\perp}$, or in other words, for the boundary point p the contribution from the normal component of the operator $((\mathbf{D}^p)^{\dagger}\lambda)_p$, i.e., either (11) or (12) is zero.

7. NUMERICAL TESTS

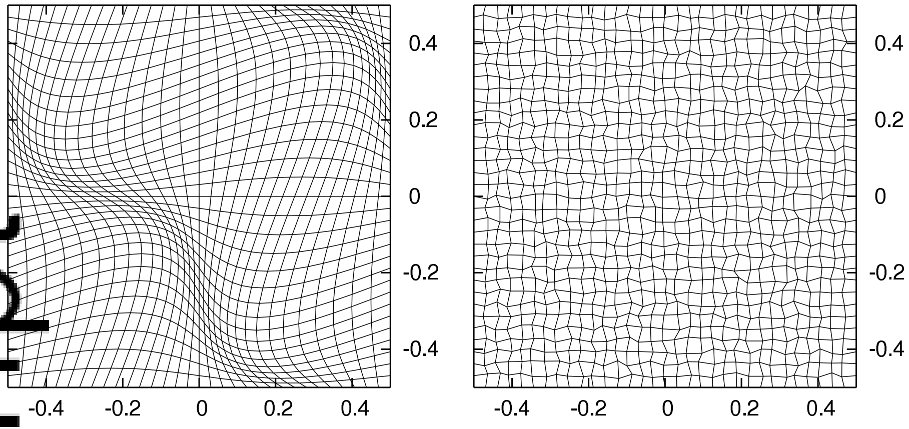
In this section we show several numerical tests of the developed method on different vector fields. The vector fields are defined analytically. The nodal components of the vector field computed from the normal components are being compared with the exact point values at the mesh nodes.

On a logically rectangular mesh the explicit form of discrete derivatives of a nodal function g at cell $i+1/2, j+1/2$ is

$$\begin{aligned}
\left(\frac{\delta g}{\delta x}\right)_{i+1/2,j+1/2} &= \frac{(g_{i+1,j+1}-g_{i,j})(y_{i,j+1}-y_{i+1,j}) + (g_{i+1,j}-g_{i,j+1})(y_{i+1,j+1}-y_{i,j})}{2V_{i+1/2,j+1/2}},\\
\left(\frac{\delta g}{\delta y}\right)_{i+1/2,j+1/2} &= \frac{(g_{i+1,j+1}-g_{i,j})(x_{i+1,j}-x_{i,j+1}) + (g_{i,j+1}-g_{i+1,j})(x_{i+1,j+1}-x_{i,j})}{2V_{i+1/2,j+1/2}},
\end{aligned}$$

where $V_{i+1/2,j+1/2}$ is the area of the cell $i+1/2, j+1/2$. The discrete divergence and curl acting from nodes to cells are computed from these discrete derivatives. The discrete divergence computed from these formulas is equivalent to the divergence defined by (2). The discrete divergence computed from the nodal components is compared with the discrete divergence in the cell computed from the normal components (1). The discrete curl is compared with the exact value at the cell center. The errors are evaluated in the max norm.

The normal components used here in the tests are given by the point values at the edges midpoints, however they can be also defined as the integral averages of the normal components over the edge. The resulting errors in the tables below are however very close for both definitions of the normal components. For all tests, unless otherwise noted, we apply the free boundary conditions.



(a) (b)

Figure 6. Types of meshes: (a) - the smooth mesh; (b) - the random mesh

7.1. Types of meshes

Everywhere the mesh covers the square $(x, y) \in [-1/2, 1/2]^2$. The simplest mesh is an uniform mesh with $M \times M$ square cells. A smooth non-orthogonal mesh [8] is obtained by the mapping

$$x(\xi, \eta) = \xi + 0.1 \sin(2\pi\xi) \sin(2\pi\eta) , \qquad y(\xi, \eta) = \eta + 0.1 \sin(2\pi\xi) \sin(2\pi\eta) \tag{16}$$

from the uniform grid on the square $[-1/2, 1/2]^2$ in the space (ξ, η) into the same square in (x, y), see Fig. 6 (a). A random non-smooth non-orthogonal mesh is created from the uniform mesh (consisting of squares $h \times h$, with $h = 1/M$) by randomly moving all internal nodes in squares with sides $h/4$ centered at the position of the node in the uniform mesh, see Fig. 6 (b).

7.2. Smooth test from [8]

In Table I we present the results for the smooth and random meshes with the smooth vector function from [8]. The vector field is defined by

$$\mathbf{w} = (x - y + x^2 - y^2, x + y + x^2 + y^2). \tag{17}$$

We present the results for the local reconstruction method as well as for the global divergence preserving method. The results presented in Table I show that for all cases we observe the second order convergence for the vector function itself and the first order convergence for its curl. For the local method we observe the first order convergence for the divergence. For the divergence preserving global method the divergence is preserved up to the machine precision. It is interesting to note that the absolute values of the error for the same mesh and resolution are almost the same for the local and global divergence preserving methods.

DOI: 10.1002/fld

Table I. Maximum errors for the smooth test (17) vector field.

grid		smooth			non-smooth		
method	M	**w**	DIV	**CURL**	**w**	DIV	**CURL**
local	32	0.32E-2	0.10E-1	0.98E-2	0.20E-2	0.39E-1	0.21E-1
	64	0.80E-3	0.26E-2	0.26E-2	0.54E-3	0.23E-1	0.14E-1
	128	0.20E-3	0.64E-3	0.65E-3	0.14E-3	0.11E-1	0.74E-2
	256	0.50E-4	0.16E-3	0.16E-3	0.34E-4	0.64E-2	0.42E-2
div-preser	32	0.31E-2	0.65E-13	0.97E-2	0.21E-2	0.17E-13	0.40E-1
	64	0.80E-3	0.99E-13	0.26E-2	0.57E-3	0.28E-13	0.33E-1
	128	0.20E-3	0.18E-12	0.65E-3	0.14E-3	0.77E-13	0.17E-1
	256	0.50E-4	0.43E-12	0.16E-3	0.39E-4	0.18E-12	0.94E-2

7.3. Shock like test from [8]

This is a 1D non-smooth shock like test from [8]. The vector field is defined by

$$\mathbf{w} = (e^{20x}/(1+e^{20x}), 0). \tag{18}$$

In Table II we present the results for this function - the results are arranged in the same way as in Table I. The convergence for the vector function is again second order in all cases. For the local method and the smooth mesh the convergence for the divergence is a little bit better than the first order. For the local method and the random mesh the convergence for the divergence is about first order. For curl it is about the first order for all cases. We would like to mention that for this "non-smooth" function the value of error may oscillate depending on the position of the mesh with respect to "front" of the "shock" and it is especially sensitive in the case of the random mesh. Again, for the global divergence preserving method the discrete divergence is indeed preserved up to the machine precision.

Table II. Maximum errors for the shock like test (18) vector field.

grid		smooth			non-smooth		
method	M	**w**	DIV	**CURL**	**w**	DIV	**CURL**
local	32	0.27E-1	0.54E+0	0.15E+0	0.19E-1	0.37E+0	0.20E+0
	64	0.77E-2	0.19E+0	0.47E-1	0.50E-2	0.12E+0	0.15E+0
	128	0.20E-2	0.52E-1	0.13E-1	0.14E-2	0.81E-1	0.72E-1
	256	0.51E-3	0.13E-1	0.32E-2	0.35E-3	0.41E-1	0.46E-1
div-preser	32	0.24E-1	0.65E-13	0.15E+0	0.17E-1	0.17E-13	0.19E+0
	64	0.65E-2	0.93E-13	0.47E-1	0.45E-2	0.30E-13	0.17E+0
	128	0.17E-2	0.18E-12	0.13E-1	0.13E-2	0.74E-13	0.91E-1
	256	0.41E-3	0.32E-12	0.32E-2	0.32E-3	0.16E-12	0.47E-1
div-preser	32	0.74E-2	0.82E-13	0.15E+0	0.61E-2	0.22E-13	0.19E+0
piston	64	0.20E-2	0.71E-13	0.47E-1	0.19E-2	0.29E-13	0.17E+0
BCs	128	0.52E-3	0.17E-12	0.13E-1	0.50E-3	0.65E-13	0.91E-1
	256	0.13E-3	0.32E-12	0.32E-2	0.14E-3	0.19E-12	0.45E-1

Table II presents also results for the divergence preserving method with the piston boundary conditions, which are presented in Section 6. The errors with the piston boundary conditions are smaller than the results with the free boundary conditions for the divergence preserving method.

The local method with the piston boundary conditions gives very similar errors (not shown in Table II) as the local method with the free boundary conditions.

7.4. *Noh-like velocity field test*

This is a radially symmetric inward directed vector field $\mathbf{w} = -|\mathbf{w}|(x, y)/r$, where $r = \sqrt{x^2 + y^2}$ with the magnitude

$$|\mathbf{w}| = \begin{cases} 0 & \text{for } r \leq r_{\min} \\ \frac{1-\cos(\pi(r-r_{\min})/a)}{2} & \text{for } r_{\min} \leq r \leq r_{min} + a \\ 1 & \text{for } r \geq r_{\min} + a \end{cases} ,$$

where the circular front position is at $r = r_{\min} + a/2$ and the width of the front is given by the parameter a. We use in our test the particular parameters $r_{\min} = 0.2$ and $a = 0.1$.

The maximum errors for this test are presented in Table III. The convergence for the vector function is second order and the divergence is preserved for the divergence preserving global method. We also present the convergence for the projection w_t of $\mathbf{w}$ to the tangential direction, which should be zero for the radial field.

Table III. Maximum errors for the Noh-like velocity field. The column labelled by w_t contains errors of the projection of $\mathbf{w}$ to the tangential direction.

grid		smooth			non-smooth		
method	M	$\mathbf{w}$	w_t	DIV	$\mathbf{w}$	w_t	DIV
local	32	0.15E+0	0.32E-1	0.67E+01	0.13E+0	0.31E-1	0.50E+1
	64	0.55E-1	0.11E-1	0.27E+01	0.41E-1	0.96E-2	0.28E+1
	128	0.16E-1	0.29E-2	0.14E+01	0.12E-1	0.27E-2	0.16E+1
	256	0.43E-2	0.76E-3	0.71E+00	0.33E-2	0.78E-3	0.75E+0
div-preser	32	0.72E-1	0.37E-1	0.85E-13	0.52E-1	0.50E-1	0.25E-13
	64	0.22E-1	0.12E-1	0.17E-12	0.18E-1	0.18E-1	0.42E-13
	128	0.68E-2	0.32E-2	0.42E-12	0.61E-2	0.59E-2	0.70E-13
	256	0.17E-2	0.83E-3	0.46E-12	0.18E-2	0.17E-2	0.15E-12

8. CONCLUSION

We have developed a new divergence preserving method for the reconstruction of the Cartesian components of the vector field at the mesh nodes from the orthogonal projections of the vector field to the normals to the edges. In this method the discrete divergences computed from the nodal and from the normal components are *exactly* the same. For the application of our method in hydrodynamics the vector field will be the fluid velocity and one might be interested in conserving also the circulation (curl), the momentum or the energy. Our method does not conserve these additional quantities and, e.g., the kinetic energy conservation would lead to non-linear optimization. One cannot expect to conserve all quantities.

We present numerical results for smooth and non-smooth vector fields as well as for smooth and non-smooth meshes. Numerical results demonstrate, that the discrete divergence is in fact preserved up to the machine precision and that it does not affect the convergence rate for the vector field itself and the discrete curl.

ACKNOWLEDGEMENTS

We thank two anonymous reviewers for their constructive comments. This work was carried out under the auspices of the National Nuclear Security Administration of the U.S. Department of Energy at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396 and the DOE Office of Science Advanced Scientific Computing Research (ASCR) Program in Applied Mathematics Research. This work has been partially supported by the Czech Science Foundation project 14-21318S and by the Czech Ministry of Education project RVO 68407700.

REFERENCES

1. Hyman J, Shashkov M. Natural discretizations for the divergence, gradient, and curl on logically rectangular grids. *International Journal of Computers & Mathematics with Applications* 1997; **33**(4):81–104.
2. Hyman J, Shashkov M. Adjoint operators for the natural discretizations of the divergence, gradient and curl on logically rectangular grids. *Applied Numerical Mathematics* 1997; **25**:413–442.
3. Hyman J, Shashkov M. The approximation of boundary conditions for mimetic finite difference methods. *Computers & Mathematics with Applications* 1998; **36**:79–99.
4. Hyman J, Shashkov M. The orthogonal decomposition theorems for mimetic finite difference methods. *SIAM J. Numer. Anal.* 1999; **36**(3):788–818.
5. Hyman J, Shashkov M. Mimetic discretizations for Maxwell's equations. *J. Comput. Phys.* 1999; **151**:881–909.
6. Hyman J, Shashkov M, Steinberg S. The numerical solution of diffusion problems in strongly heterogeneous non-isotropic materials. *J. Comput. Phys.* 1997; **132**:130–148.
7. Campbell J, Hyman J, Shashkov M. Mimetic finite difference operators for second-order tensors on unstructured grids. *Computers & Mathematics with Applications* 2002; **44**:157–173.
8. Shashkov M, Swartz B, Wendroff B. Local reconstruction of a vector field from its normal components on the faces of grid cells. *J. Comput. Phys.* 1998; **139**:406–409.
9. Beirao da Veiga L, Lipnikov K, Manzini G. *Mimetic finite difference method for elliptic problems*. Modeling, Simulation & Applications, Volume 11, Springer, 2014. 408 pages.
10. Lipnikov K, Manzini G, Shashkov M. Mimetic finite difference method. *J. Comput. Phys.* January 2014; **257**(Part-B):1163–1228.
11. Perot J, Vidovic D, Wesseling P. Mimetic reconstruction of vectors. *Compatible Spatial Discretizations*, Arnold D, Bochev P, Lehoucq R, Nicolaides R, Shashkov M (eds.). Springer, 2004; 173–188.
12. Dukowicz J, Meltz B. Vorticity errors in multidimensional Lagrangian codes. *J. Comput. Phys.* 1992; **99**:115–134.
13. Dukowicz J, Cline M, Addessio F. A general topology Godunov method. *J. Comput. Phys.* 1989; **82**:29–63.
14. Lesoinne M, Farhat C. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Comp. Meth. Appl. Mech. Engrg.* 1996; **134**(1):71–90.
15. Farhat C, Geuzaine P, Grandmont C. The discrete geometric conservation law and the nonlinear stability of ALE schemes for the solution of flow problems on moving grids. *J. Comput. Phys.* 2001; **174**(2):669–694.
16. Maire PH, Abgrall R, Breil J, Ovadia J. A cell-centered Lagrangian scheme for two-dimensional compressible flow problems. *SIAM Journal on Scientific Computing* 2007; **29**(4):1781–1824.
17. Maire PH. A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. *J. Comput. Phys.* 2009; **228**(7):2391–2425.
18. Burton D, Carney T, Morgan N, Sambasivan S, Shashkov M. A cell-centered Lagrangian Godunov-like method for solid dynamics. *Computers & Fluids* 2013; **83**:33–47.
19. Sambasivan S, Shashkov M, Burton D. A finite volume cell-centered Lagrangian hydrodynamics approach for solids in general unstructured grids. *Int. J. Numer. Meth. Fluids* 2013; **72**(7):770–810.
20. Chorin AJ. Numerical solution of the Navier-Stokes equations. *Math. Comp.* 1968; **22**:745–762.

DOI: 10.1002/fld