Edge-based Multigrid schemes for Hybrid Grids

Jens-Dominik Müller and Michael B. Giles
Oxford University Computing Laboratory

Abstract

A Navier-Stokes multigrid method that has been developed for triangles and tetrahedra is extended to be used on hybrid grids. Various choices in extending the linear preserving flux calculation of the simplex scheme are discussed and a flexible way of expressing edge weights is presented. An edge-collapsing algorithm that generates the coarser levels of hybrid grids for a multigrid algorithm is described.

1 Introduction

Crumpton et. al. [2] have presented an unstructured triangular or tetrahedral multigrid method for the Navier-Stokes equations. The solution is obtained with a node-centered finite volume method that employs an edge based data structure. This data structure makes for very efficient programming. The implementation does not change for a 2D or 3D version.

An important characteristic of the discretization is that all spatial operators are "linear preserving" (LP). That is, a linear variation of a flux or state variable is integrated (or differentiated) exactly on an arbitrary mesh. This property guarantees that the order of accuracy of the scheme is preserved on an irregular mesh, a highly stretched mesh or an adapted mesh. Consider a solution u that is stored at the nodes and varies linearly over each element (cf. fig. 1). A Galerkin discretization of the gradient on linear elements corresponds

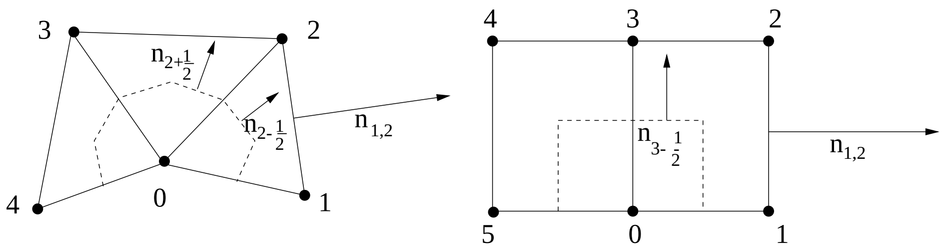

Figure 1: Median dual around an internal node on a simplex (a) and a primitive grid (b).

to a Green-Gauß integration around the neighboring nodes, which is LP by construction.

$$\int_\Omega \nabla u_0 dA = \sum_i \frac{1}{2}(u_i + u_{i+1})\mathbf{n}_{i,i+1} = \sum_i \frac{1}{2}(u_i + u_0)\left(\mathbf{n}_{i-1,i} + \mathbf{n}_{i,i+1}\right) \tag{1.1}$$

On a simplex mesh the integral around the neighboring nodes can be formulated equivalently around the median dual volume, indicated by dashed lines in figure 1. It can then be expressed as a sum over each edge connected to node i,

$$\int_\Omega \nabla u_0 dA = \sum_i \frac{1}{2}(u_i + u_0)\left(\mathbf{n}_{i-\frac{1}{2}} + \mathbf{n}_{i+\frac{1}{2}}\right). \tag{1.2}$$

Boundary terms have to be added for boundary faces of a simplex. This formulation can be used very efficiently in the context of an edge based data structure. An extensive discussion of edge-based formulae for simplices has been presented by Barth [1].

On a non-simplex mesh the Gauß-integration cannot be expressed on the edges of the mesh. As can be seen from figure 1, there exist neighboring nodes that are connected by a cell, but not by an edge, e.g. the node diagonally opposite in a quadrilateral. Conversely, integration on just the edges as in eq. 1.2 is not LP except for regular meshes formed by parallelepipeds. In these cases errors of opposite edges cancel.

A standard way to recover LP for non-simplex meshes, is to "triangulate" the "primitive". A primitive in our context is a quadrilateral in 2D, a pyramid, prism and a hexahedron in 3D. Triangulation here means the decomposition into triangles in 2D or tetrahedra in 3D. This approach was also adopted in [2]. In the case of a quadrilateral this corresponds to changing the bilinear variation over the element to a piecewise linear one with an arbitrary choice of diagonals. While this approach does recover LP, it is wrought with various difficulties. The decomposition is not possible in general, certain pathological cases cannot be triangulated. Furthermore, the triangulation adds diagonal edges also for regular primitives that are LP. Triangulation of a hexahedron, e.g., adds 4 edges per node to the 3 edges per node of the hexahedron. Thus the triangulation of a hexahedral grid approximately roughly doubles the cost of the calculation. Moreover, the orthogonality of quasi-structured, rectangular grids is desirable for the calculation of boundary layers.

In section 2 we propose an alternative way of integration over primitives that is LP. This approach does not require the triangulation of all primitive elements. Diagonal edges are added to the nodes where the integration turns out not to be LP. The weights on these edges are formulated such that they can express a one-sided contribution as opposed to the anti-symmetric standard edges.

Another key ingredient in [2] is an edge-collapsing multigrid algorithm, that generates coarser levels starting from an arbitrary simplex grid of the finest level. The basic algorithm is to collapse the two nodes of an edge into one and to retriangulate the cavity of all cells that were connected to the edge with that new node. Coloring of the nodes ensures that specified coarsening ratios in isotropic and stretched regions are achieved. The cardinality of the algorithm is optimal for simplex meshes in the sense that the ratio of number of cells to nodes remains constant during the coarsening process. However, the straightforward generalization to hybrid grids is not optimal in that sense: the retriangulation of the cavity will replace hexahedra with pyramids and tetrahedra. Since a tetrahedral grid has more than twice as many edges per node than a hexahedral grid, a semi-coarsening of a regular hexahedral region would decrease the number of nodes by a factor of 2, but would not decrease the number of edges to be computed. Furthermore the algorithm of [2] does not recover the regularity of the mesh on a coarser level.

The hybrid edge-collapsing algorithm proposed in section 3 is designed to avoid these problems. This is done by considering primarily the graph problem of nodes connected by edges. Collapsing edges is then constrained by not allowing neighboring edges to exceed a certain factor of their initial length, usually the double. Partially collapsed elements that have not completely disappeared have to be considered.

2 Spatial Discretization

As shown in section 1, the loss of linearity preservation can be viewed as an effect of the decoupling of nodes that do not have an edge in common, such as the node diagonally opposite in a quadrilateral. If the geometry is regular, such as a node surrounded by parallelepipeds, the errors associated with the simplified quadrature cancel and a simple integration is LP. There is no need to triangulate these elements. In a typical application of hybrid grids, a flow simulation with boundary layers, about half the elements will be in the boundary layer.

Most of these elements will be very orthogonal hexahedra, only the elements on the boundary and on the hybrid interface will require modification.

2.1 Symmetric Edges

The simple edge-weights **a**, as they are used in simplex implementations, are anti-symmetric. That is, the contribution of the edge appears with opposite signs at either end, ensuring conservation by construction.

$$A_i \nabla u_i \leftarrow A_i \nabla u_i + \mathbf{a}_{ij} \frac{1}{2}(u_i + u_j), \qquad A_j \nabla u_j \leftarrow A_j \nabla u_j - \mathbf{a}_{ij} \frac{1}{2}(u_i + u_j) \tag{2.1}$$

In order to obtain complete flexibility in setting up the quadrature around the median dual, we define a new type of edge that brings a symmetric contribution **s** to both nodes.

$$A_i \nabla u_i \leftarrow A_i \nabla u_i + \mathbf{s}_{ij} \frac{1}{2}(u_i + u_j), \qquad A_j \nabla u_j \leftarrow A_j \nabla u_j + \mathbf{s}_{ij} \frac{1}{2}(u_i + u_j) \tag{2.2}$$

We can express any contribution from the node j on one end of the edge to node i on the other with a combination of the two types of weights. Note that conservation is no more guaranteed by construction, but has to be provided by other symmetric edges. While the form of eqs. 2.1, 2.2 is computationally very efficient, it is conceptually easier to calculate the weights as two one-sided weights from $i \to j$, $\mathbf{w}_{ij}$, and $j \to i$, $\mathbf{w}_{ji}$, and to convert it to $\mathbf{a}, \mathbf{s}$.

$$\mathbf{a} = \frac{1}{2}(\mathbf{w}_{ij} - \mathbf{w}_{ji}), \qquad \mathbf{s} = \frac{1}{2}(\mathbf{w}_{ij} + \mathbf{w}_{ji}) \tag{2.3}$$

The calculation of the weights $\mathbf{w}_{ij}$ and $\mathbf{w}_{ji}$ can be done straightforwardly. Considering in figure 1(b) the right half of the edge $0 \to 3$, $e_{3-\frac{1}{2}}$, one can define the bilinear interpolation value that is LP for a trapezium rule integration at that edge,

$$u_{3-\frac{1}{2}} = \frac{1}{2}\left(\frac{1}{2}(u_0 + u_3) + \frac{1}{4}(u_0 + u_1 + u_2 + u_3)\right). \tag{2.4}$$

The edges $0 \to 1$ and $0 \to 3$ exist as simple edges, the diagonal edge $0 \to 2$ and the "reflex" edge $0 \to 0$ have to be added. The edge-based data structure only knows about connections between two nodes at a time. We have to scatter the contribution from edge $e_{3-\frac{1}{2}}$ to each of the four contributing edges,

$$\mathbf{w}_{00} \leftarrow \mathbf{w}_{00} + \frac{1}{2}(\frac{1}{2} + \frac{1}{4})\mathbf{n}_{3-\frac{1}{2}}, \qquad \mathbf{w}_{01} \leftarrow \mathbf{w}_{01} + \frac{1}{2}(\frac{1}{4})\mathbf{n}_{3-\frac{1}{2}}, \tag{2.5}$$

$$\mathbf{w}_{02} \leftarrow \mathbf{w}_{02} + \frac{1}{2}(\frac{1}{4})\mathbf{n}_{3-\frac{1}{2}}, \qquad \mathbf{w}_{03} \leftarrow \mathbf{w}_{03} + \frac{1}{2}(\frac{1}{2} + \frac{1}{4})\mathbf{n}_{3-\frac{1}{2}}. \tag{2.6}$$

For the receiving end at node 3 we find the same contributions with the opposite sign, thus the integration is conservative. The edge $1 \to 3$ will have to be created as well. The extension to three dimensions is straightforward once a suitable interpolation over the element has been defined.

In the example above we have chosen to interpolate the the unknown bilinearly over the quadrilateral. This interpolation involves both diagonal edges in the element, $0 \to 2$ and $1 \to 3$. However, we are only interested in recovering a linear function which can be expressed exactly on two triangles defined over the quadrilateral with either diagonal. Selecting e.g. $0 \to 2$, we find the interpolation value

$$u_{3-\frac{1}{2}} = \frac{1}{2}\left(\frac{1}{2}(u_0 + u_3) + \frac{1}{2}(u_0 + u_2)\right). \tag{2.7}$$

The diagonal edge $1 \to 3$ does not appear. This choice does introduce some bias into the grid, but can offer significant savings in 3D. E.g. a trilinear interpolation over a hexahedron requires all nodes of the hexahedron to be connected to each other with an edge. This generates 8 additional edges per node, while a simplex decomposition or a piecewise linear interpolation only add 4 diagonal edges. Note that with all choices of interpolation schemes, the geometry of the dual volume will keep its desirable rectangular shape. This is not the case with the 'real' triangulation of [2].

2.2 Finding non-LP edges

It remains to be addressed how to select the faces of the median dual that need to be treated LP. Since we would like to use the simple integration as much as possible, we start out by calculating non-LP weights in all non-simplex elements. Boundary faces have to be treated LP in any case. We can then check the accuracy of our interpolation by calculating gradients of unit magnitude in each coordinate direction, $u_i = x_i, y_i, z_i$. The accuracy of the current integration is evaluated by comparing $\nabla y_i / \nabla x_i$ and $\nabla z_i / \nabla x_i$. All nodes that do not have a set of sufficiently accurate weights are identified.

It is rather involved to find out which edge introduces the error, and which edge has its error canceled by the error of another edge as is the case in regular grids. We thus proceed to "upgrade" to LP all edges of primitive elements that are formed with this node. This procedure gives rise to some propagation: conservation requires that the integration for the the node at the other end of the edge is treated symmetrically. Since this in turn can remove error cancellation for the neighboring node, the upgrade can propagate.

Table 2.2 shows the number of added edges for a hybrid grid around a RAE 2822 airfoil with a tolerance fixed at 1 %. Three successively coarser grids are considered with about half the elements being quadrilaterals in the viscous layer. A close-up of the finest grid can seen in figure 5. Bilinear integration for LP-quadrilaterals has been chosen. The rightmost column

tris	quads	antim.,	symm.,	bnd. Edges	added Eg./quad
10692	13647	46817	5948	266	.397
3788	3452	14288	2783	138	.726
1520	884	3024	1412	72	1.43

Table 1: Number of added edges for hybrid grids around a RAE 2822, LP-tolerance at 1 %.

lists the number of added edges normalized by the number of quadrilaterals. Triangulation to simplices would have added one edge per quadrilateral. The selective approach generates in general fewer edges than a complete triangulation, except for the coarsest grid. Note that reverting to a piecewise linear interpolation adds only one edge per quadrilateral, rather than the two added in the bilinear case.

Tolerance	antim.	symm. Edges	added Eg./quad
.1	9156	1805	.300
.05	12640	6057	1.34
.01	16196	8596	1.96
.001	16528	8665	1.97

Table 2: Added edges in a structured grid for a NACA 0012, varying LP tolerance.

Table 2.2 shows the number of added edges for a structured 128x33 C-type grid around a NACA 0012 airfoil with varying tolerance. The mesh contains 4096 quadrilaterals and 288 boundary edges.

Details of the corresponding meshes are shown in figure 2 With the largest tolerance of

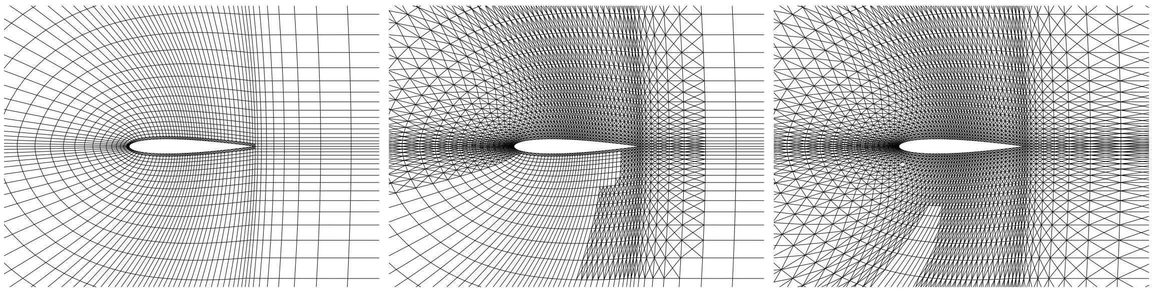

Figure 2: Added edges in a quad. mesh to achieve LP. Tolerance .1, .05, .01, from left.

a 10 % error in gradient calculation, there are essentially only boundary edges that are fixed up. All the quadrilaterals on the boundaries carry the two diagonal edges. Increasing the tolerance quickly leads to a grid that prescribes the more costly LP integration for nearly all cells. Computational experiments are being currently undertaken to investigate the influence of the LP threshold on the accuracy of the solution.

3 Edge-collapsing Multigrid

The new edge-collapsing algorithm that we propose works primarily on the graph of the mesh. In this graph, any edge can be collapsed if after the collapse the geometry is still valid and none of the neighboring edges exceeds a maximum length it has been allowed. The first criterion is obvious, we cannot tolerate negative volumes due to folded grids. The second criterion expresses the design principle of multigrid, usually the coarse grid doubles the length of the edges in a mesh.

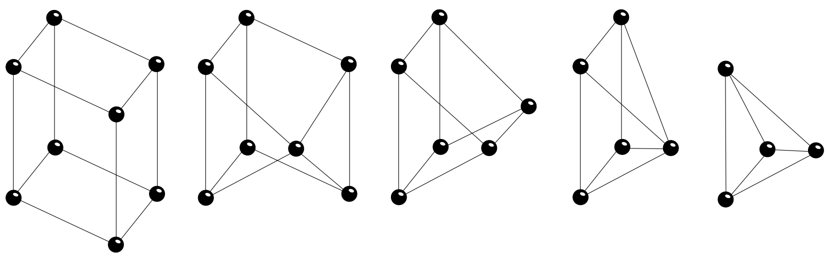

Figure 3: Collapsing edges on a hexahedron.

The implementation of this algorithm for isotropic meshes is straightforward. Given a fine mesh, we tag each edge with its length times a growth factor, say 2, as maximum length. The edges are then sorted in a heap list for shortest edge-length and we try to collapse the shortest edge. With a list of edges that is addressed by node numbers, the length test can be executed as a $O(1)$ operation. Similarly, we construct a list of elements addressed by nodes to perform the geometry test. Fixing a certain maximum angle for the elements in the collapsed geometry, say 135°, guarantees a minimum quality of the coarser mesh as well as positive volumes. This test is done by looping over all elements that are formed with the collapsed edge and considering the “remaining” element. Other edges on these elements may have been collapsed in earlier steps. E.g. a quadrilateral with one collapsed edge becomes a triangle, a doubly collapsed quadrilateral vanishes. Various collapsed shapes derived from a hexahedron are shown in figure 3. The algorithm terminates once there are no edges left to be collapsed. All remaining elements and nodes are then identified and a coarsened grid is created. A sequence of three levels of a triangular grid around a NACA 0012 with initially 11400 triangles is shown in figure 4.

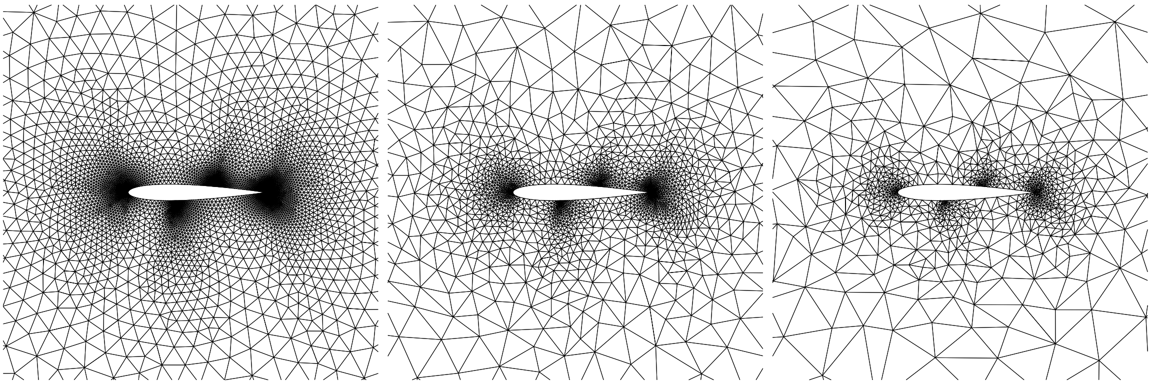

Figure 4: Coarsening a triangular grid around a NACA 0012.

The algorithm as described above for the isotropic case will overcoarsen anisotropic grids with a high gradation as is typically found in boundary layers. Larger edges have to be prevented from "eating" the smaller ones that limit the coarsening. This is achieved by identifying short edges in stretched regions. A first criterion is that these edges are shorter by a given factor, say 3, compared to the largest neighboring edge. Additionally we require that there is at least one other neighboring edge that is short and points into the same direction. This criterion ensures that single short edges in very irregular unstructured grids are not considered stretched.

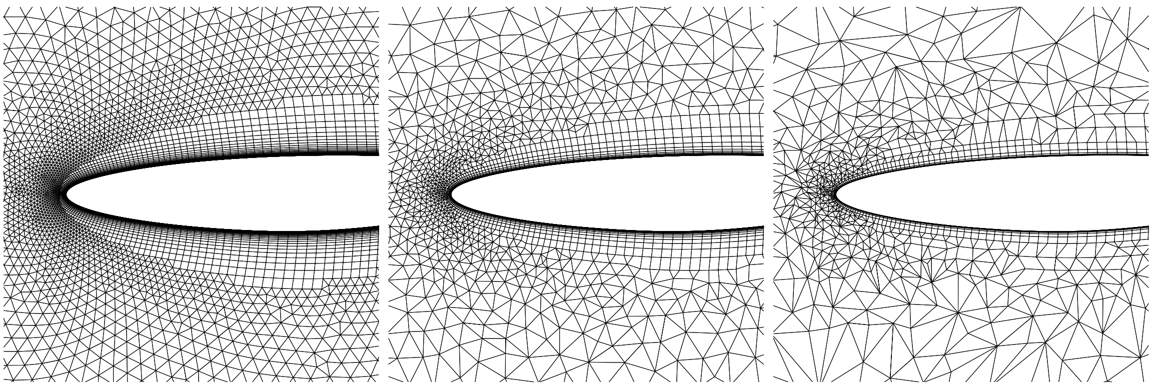

Figure 5: Coarsening a hybrid grid around a RAE 2822.

Starting with the shortest stretched edge, the string of short edges is followed in both directions of the edge. Keeping the shortest edge, every other edge in the string is collapsed onto the outward end, and all neighboring long edges are given a maximum length that prohibits any further collapse in the region. Once the stretched regions have been directionally coarsened in this way, the isotropic process collapses the rest of the domain. Figure 5 shows a sequence of collapses for a hybrid grid around a RAE 2822 airfoil. It can be seen that the stretched part of the grid close to the airfoil remains regular and is coarsened exactly 1:2. The outer part of the structured region which is not stretched loses some regularity and the quadrilaterals collapse into larger quadrilaterals and triangles.

Table 3 shows the coarsening ratios of the two examples. The length ratio has been fixed at 2.2, the maximum angle at 135°. This angular tolerance is quite low, the specified value is often considered acceptable even for the finest grid. As can be seen, there is a trade-off

tri.	ratio
11400	1.
4300	.37
1900	.44
1100	.57
700	.64
400	.57

tri.	quad.	elements	tri.	quad.	total ratio
10692	13647	24339	1.	1.	1.
5527	5957	11484	.54	.44	.47
2795	2733	5528	.51	.45	.47
1599	1295	2894	.57	.48	.52
1053	672	1725	.65	.51	.60
832	291	1123	.79	.43	.65
651	50	701	.78	.17	.64

Table 3: Coarsening ratios for a tri. NACA 0012 on the left, hybrid RAE 2822 on the right.

between the desired coarsening ratio, ideally 1:4 on a 2-D isotropic grid, and the quality on the coarser meshes. Numerical experiments will show how much grid quality is needed on the coarser levels.

The hybrid example exhibits a perfect coarsening ratio in the structured part. The coarsening ratio is slightly below 1:2 since some of the quadrilaterals at the outer, non-stretched part collapse into triangles. The coarsening ratio on the triangles is again rather high because of the strict angular tolerance. Coarsening is also restricted by the perimeter of the structured layer which always carries the same number of nodes, coarsening happens only normal to the airfoil surface. Furthermore, partly collapsed quadrilaterals add to the number of triangles.

4 Conclusions

A flexible way of defining edge weights for edge-based solvers on hybrid grids has been developed. It allows to choose an appropriate trade-off between accuracy, namely a gradient operator is linear preserving on an arbitrary mesh, and cost in the form of added edges that have to be computed. While the choice of integration is local, conservation leads to a propagation of the more accurate treatment over large parts of the domain when a more stringent tolerance for the linearity preservation is chosen. Numerical experiments will have to show the influence of the tolerance on the solution accuracy.

An edge-collapsing algorithm to generate coarsened meshes for multigrid methods has been developed that can deal with arbitrary meshes. The method applies directional coarsening in stretched regions and preserves the regularity of the mesh there. Grid quality on the coarser meshes can be guaranteed at the price of an increase of the coarsening ratio. Two-dimensional examples are shown.

References

[1] T.J. Barth. Aspects of unstructured grids and Finite-Volume solvers for the Euler and Navier-Stokes equations. In *Special Course on Unstructured Grid Methods for Advection Dominated Flows*, Neuilly sur Seine, France, May 1992. AGARD. Report 787.

[2] P.I. Crumpton, P. Moinier, and M.B. Giles. An unstructured algorithm for high Reynolds number flows on highly-stretched grids. In C. Taylor and J.T. Cross, editors, *Numerical Methods in Laminar and Turbulent Flow*, pages 561–572. Pineridge Press, 1997.