



# Extension of the Flux Reconstruction Method to Triangular Elements using Collapsed-Edge Quadrilaterals

Joshua Romero\*, Antony Jameson†

*Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, 94305*

**This paper presents an extension of the Flux Reconstruction (FR) method to viscous flow problems on triangular elements using collapsed-edge tensor product quadrilaterals. First, an overview of the direct Flux Reconstruction (DFR) formulation of the FR method is given. Following this, a description of the procedure to apply the tensor product FR formulation on quadrilaterals directly to triangular elements is provided. Results verifying that the developed methodology maintains the expected order of accuracy for linear and nonlinear fluxes on triangular meshes are provided, specifically for an advection-diffusion problem and planar Couette flow problem. Following this, the results of an additional test case solving the Navier-Stokes equations viscous flow around a NACA0012 airfoil geometry using curved elements and a mixed mesh are presented, illustrating the performance of the developed methodology for more realistic flows.**

## I. Introduction

In recent years, high-order numerical methods have attracted significant attention in the CFD community as an alternative way to simulate fluids problems. In general, the less dissipative nature of these methods makes them better suited than lower order schemes for solving certain fluid problems, such as those involving vortex-dominated flows. In some cases, high-order solution methods can be more computationally efficient than lower-order schemes at achieving a desired error tolerance. In general, high-order methods have the potential to open up the study of a much broader field of problems while improving the efficiency and accuracy of the solutions obtained to fluid problems relevant to engineering design.

Of the many popular high-order schemes, the Flux Reconstruction (FR) method, introduced by Huynh in 2007, has proven to be an attractive method to obtain high order solutions to flow problems of engineering interest.<sup>1</sup> The originally proposed scheme for one-dimensional problems can be easily extended to quadrilateral and hexahedral elements using tensor products. The tensor product formulation, since it is built up directly from the one-dimensional scheme, is relatively simple to implement in a computer program.

However, many real-world problems involve complex geometries, which generally require unstructured meshes that include triangular elements in 2D and tetrahedral elements in 3D. To this end, many efforts have been made to extend FR methods to triangular and tetrahedral elements.<sup>2,3</sup> In these extensions of the FR method, the formulation on triangles and tetrahedra is generally far removed from the simple tensor product formulation used on quadrilaterals and hexahedra due to the absence of a natural tensor product basis on elements of this type. As such, these extensions require the definition of a fully multi-dimensional methodology involving vector valued operations. While these extensions have been successful, they are more analytically complex and generally more difficult to implement than the simple extension to higher dimensional elements offered through tensor products. One alternative approach that has been employed for spectral element methods is the use of collapsed-edge elements, which allow the application of a tensor product based method directly to triangular and tetrahedral elements.<sup>4,5</sup>

In this paper, an extension of the FR method to triangular elements using collapsed-edge tensor product quadrilaterals is presented. This method leverages the simplicity of implementing the method on quadrilateral elements and defines an implementation framework for extending tensor product formulations of the FR method to alternative elements using edge collapsing. First, an overview of the FR method is provided followed by a brief description of

\*Ph.D. Candidate, Department of Aeronautics and Astronautics, Stanford University, AIAA Student Member; jdromero@stanford.edu

†Professor, Department of Aeronautics and Astronautics, Stanford University, AIAA Member

the extension to quadrilateral elements using tensor products. Next, a description of the method to apply the tensor product formulation to triangular elements using edge collapsing is discussed. Following this, the numerical results of several test cases are presented, demonstrating the performance and limitations of the developed methodology.

## II. Methods

### A. Direct Flux Reconstruction Method in 1D for second-order fluxes

To begin, a brief review of the direct Flux Reconstruction (DFR) method for second-order fluxes is required. In this study, the DFR formulation of the FR method is used as this method simplifies the implementation of the standard FR methodology and has been proven to recover the nodal discontinuous Galerkin (DG) variant of the FR method exactly<sup>6</sup>. Detailed descriptions of the DFR method and the standard FR method for convective fluxes can be found in the Refs. 1, 6 and 7. Additionally, descriptions of the standard FR method applied to diffusive fluxes can be found in the Refs. 8 and 9.

Consider the 1D scalar conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = 0 \quad (1)$$

where  $x$  is the spatial coordinate,  $t$  is time,  $u = u(x, t)$  is a conserved scalar quantity and  $f = f(u, \frac{\partial u}{\partial x})$  is the flux, defined on a closed interval  $\Omega$ . Recast Eq.(1) as a system of first-order equations

$$\frac{\partial u}{\partial t} + \frac{\partial f(u, q)}{\partial x} = 0 \quad (2)$$

$$q - \frac{\partial u}{\partial x} = 0 \quad (3)$$

where  $q = q(x, t)$  is a new auxiliary variable.

Consider partitioning  $\Omega$  into  $N$  non-overlapping elements, each denoted by  $\Omega_j = \{x | x_j < x < x_{j+1}\}$  such that

$$\Omega = \bigcup_{j=1}^N \Omega_j \quad (4)$$

With the domain partitioned, the exact solution  $u$  in Eq.(2) can be approximated by numerical solutions  $u_j^{\delta D}$  which are defined as polynomials of degree  $P$  within each  $\Omega_j$  and exactly zero outside the element. The piecewise sum of such polynomials results in the global approximation  $u^\delta$  to the exact solution which is generally discontinuous between elements. The exact flux  $f(u, q)$  in Eq.(2) can be approximated by fluxes  $f_j^\delta$  which are defined as polynomials of degree  $P + 2$  within each  $\Omega_j$  and exactly zero outside the element. The piecewise sum of these polynomials results in the global approximation  $f^\delta$  which is generally  $C^0$  continuous between elements. In similar fashion, the exact value of the auxiliary variable  $q$  in Eq.(3) can be approximated by polynomials  $q_j^{\delta D}$  which are of degree  $P$  within each  $\Omega_j$  and exactly zero outside the element. The piecewise sum of these polynomials results in the global approximation  $q^\delta$  which is generally discontinuous between elements. Like the flux in Eq.(2), the solution  $u$  in Eq.(3) can be approximated by numerical solutions  $u_j^\delta$  which are polynomials of degree  $P+2$  within each  $\Omega_j$  and exactly zero outside the element. The piecewise sum of these polynomials results in a global approximation  $u^\delta$  which is generally  $C^0$  continuous between elements. Note that for the DFR method,  $f_j^\delta$  and  $u_j^\delta$  are of degree  $P + 2$  due to a modified correction procedure which is explained in what follows.

Introduce an isoparametric mapping to transform each element  $\Omega_j$  to a standard element  $\Omega_{js} = \{r | -1 < r < 1\}$

$$x = \Gamma_j(r) = \left(\frac{1-r}{2}\right) x_j + \left(\frac{1+r}{2}\right) x_{j+1} \quad (5)$$

$$r = \Gamma_j^{-1}(x) = 2 \left(\frac{x - x_j}{x_{j+1} - x_j}\right) - 1 \quad (6)$$

Application of this transformation to Eq.(2) and Eq.(3) results in a transformed system within the  $j^{th}$  standard element

$\Omega_{js}$  of the following form

$$\frac{\partial \hat{u}_j^{\delta D}}{\partial t} + \frac{1}{J_j} \frac{\partial \hat{f}_j^{\delta}}{\partial r} = 0 \quad (7)$$

$$\hat{q}_j^{\delta D} - \frac{1}{J_j} \frac{\partial \hat{u}_j^{\delta}}{\partial x} = 0 \quad (8)$$

where

$$\hat{u}_j^{\delta D} = u_j^{\delta D}(\Gamma(r), t) \quad (9)$$

$$\hat{f}_j^{\delta} = f_j^{\delta}(\Gamma(r), t) \quad (10)$$

$$\hat{q}_j^{\delta D} = q_j^{\delta D}(\Gamma(r), t) \quad (11)$$

$$\hat{u}_j^{\delta} = u_j^{\delta}(\Gamma(r), t) \quad (12)$$

and  $J_j$  is the determinant of the element Jacobian,  $J_j = \frac{1}{2}(x_{j+1} - x_j)$ . For the remainder of this paper, the hat notation to denote the variables in the transformed space will be dropped for brevity.

In the first step of the DFR method, introduce a discretization for  $u_j^{\delta D}$  as an expansion using Lagrange polynomials defined on a set of  $P + 1$  interior solution points

$$u_j^{\delta D}(r) = \sum_{n=1}^{P+1} u_{jn}^{\delta} l_n \quad (13)$$

where  $u_{jn}^{\delta}$  are the known solution values at interior solution points,  $r_n$ , and  $l_n$  are the corresponding Lagrange polynomials.

In the next step, common interface solution values are introduced by computing the discontinuous solution values on the element interfaces using Eq.(13) and using corresponding interface values in neighboring elements

$$u_j^{\delta L} = U(u_{j-1}^{\delta D}(1), u_j^{\delta D}(-1)) \quad (14)$$

$$u_j^{\delta R} = U(u_j^{\delta D}(1), u_{j+1}^{\delta D}(-1)) \quad (15)$$

where  $U(L, R)$  is the function used to compute the common solution and  $u_j^{\delta L}$  and  $u_j^{\delta R}$  are the common interface solution on the left and right boundaries of the  $j^{\text{th}}$  element respectively. There are several approaches used to compute common solution, and later, common interface flux values, such as the central flux (CF)<sup>10</sup> and local discontinuous Galerkin (LDG)<sup>11</sup> methods. In this work, the CF approach is used, which computes the common solution using a simple average

$$u_j^{\delta L} = \frac{1}{2}(u_{j-1}^{\delta D}(1) + u_j^{\delta D}(-1)) \quad (16)$$

$$u_j^{\delta R} = \frac{1}{2}(u_j^{\delta D}(1) + u_{j+1}^{\delta D}(-1)) \quad (17)$$

The following step is to construct a continuous solution  $u_j^{\delta}$  such that a piecewise sum results in a globally  $C^0$  continuous solution,  $u^{\delta}$ , that passes through the common interface solution values at element interfaces. This can be accomplished using a single Lagrange interpolation of the form

$$u_j^{\delta}(r) = u_j^{\delta L} \tilde{l}_0(r) + \sum_{n=1}^{P+1} u_{jn}^{\delta D} \tilde{l}_n(r) + u_j^{\delta R} \tilde{l}_{P+2}(r) \quad (18)$$

where  $\tilde{l}_n(r)$  are the Lagrange interpolating polynomials defined at  $P + 3$  collocation points  $\{-1, r_1, \dots, r_{P+1}, 1\}$ , which are a combined set of the interior solution points and interface flux points and  $u_j^{\delta}$  is the resulting continuous solution. This continuous solution is a polynomial of degree  $P + 2$  which takes the values of the common interface

solution at the boundaries and preserves the original values of the discontinuous solution in the element interior exactly. Note that this step is what differentiates the DFR method from the standard FR method, where a Lagrange interpolation is used in place of left and right correction polynomials to generate  $u_j^\delta$ .

The continuous solution, described by Eq.(18), can be substituted into Eq.(8) in the transformed system to obtain an expression for  $q_j^\delta$ , which is of order  $P + 1$

$$q_j^\delta(r) = \frac{1}{J_j} \left[ u_j^{\delta L} \frac{d\tilde{l}_0}{dr} + \sum_{m=1}^{P+1} u_{j_n}^{\delta D} \frac{d\tilde{l}_n}{dr} + u_j^{\delta R} \frac{d\tilde{l}_{P+2}}{dr} \right] \quad (19)$$

The discontinuous auxiliary variable  $q_j^{\delta D}$  required for the remainder of the scheme is obtained by undersampling  $q_j^\delta$  at the interior solution points  $r_n$  only using corresponding Lagrange polynomials  $l_n$

$$q_j^{\delta D} = \sum_{n=1}^{P+1} q_j^\delta(r_n) l_n \quad (20)$$

This procedure results in a collocation projection of the polynomial  $q_j^\delta$  which is of degree  $P + 1$  to the space of polynomials of degree  $P$ , which is required to maintain consistency with the standard FR methodology.

With  $q_j^{\delta D}$  now defined, the next step of the DFR method is to introduce a discretization for  $f_j^{\delta D}$  as an expansion using Lagrange polynomials defined on a set of  $P + 1$  interior solution points

$$f_j^{\delta D}(r) = \sum_{n=1}^{P+1} f_{j_n}^\delta l_n \quad (21)$$

where  $f_{j_n}^\delta$  are the computed flux values at interior solution points,  $r_n$ , and  $l_n$  are the corresponding Lagrange polynomials.

Next, common interface flux values are introduced by computing values of the discontinuous auxiliary variable  $q_j^{\delta D}$  on the element interfaces using Eq.(20) and using corresponding interface values from neighboring elements as left and right states in an appropriate approximate numerical flux formulation for the equation being solved

$$f_j^{\delta L} = F(q_{j-1}^{\delta D}(1), q_j^{\delta D}(-1)) \quad (22)$$

$$f_j^{\delta R} = F(q_j^{\delta D}(1), q_{j+1}^{\delta D}(-1)) \quad (23)$$

where  $F(L, R)$  is the function used to compute the common flux and  $f_j^{\delta L}$  and  $f_j^{\delta R}$  are the common interface solution on the left and right boundaries of the  $j^{th}$  element respectively. As noted with the common solution, there are several approaches used to compute common flux. In this work, corresponding with the the CF method, the common flux is computed as

$$f_j^{\delta L} = \frac{1}{2}(f_{j-1}^{\delta IR} + f_j^{\delta IL}) + \tau (u_{j-1}^{\delta D}(1) - u_j^{\delta D}(-1)) \quad (24)$$

$$f_j^{\delta R} = \frac{1}{2}(f_j^{\delta IR} + f_{j+1}^{\delta IL}) + \tau (u_j^{\delta D}(1) - u_{j+1}^{\delta D}(-1)) \quad (25)$$

where  $f_j^{\delta IL}$  and  $f_j^{\delta IR}$  are the fluxes computed using the values of the discontinuous auxiliary variable,  $q_j^{\delta D}$ , and discontinuous solution  $u_j^{\delta D}$  at the left and right interfaces of the  $j$ -th element respectively and  $\tau$  is a penalty parameter controlling the jump in solution at element interfaces.

The following step is to construct a continuous solution  $f_j^\delta$  such that a piecewise sum results in a globally  $C^0$  continuous flux,  $f_j^\delta$ , that passes through the common interface flux values at element interfaces. This can be accomplished using the same procedure described by Eq.(18), resulting in

$$f_j^\delta(r) = f_j^{\delta L} \tilde{l}_0(r) + \sum_{n=1}^{P+1} f_{j_n}^{\delta D} \tilde{l}_n(r) + f_j^{\delta R} \tilde{l}_{P+2}(r) \quad (26)$$

As with the continuous solution,  $f_j^\delta$  is a polynomial of degree  $P + 2$  which takes the values of the common interface fluxes at the boundaries and preserves the original values of the discontinuous flux in the element interior exactly.

In the final step of the DFR method,  $f_j^\delta$  can be substituted into Eq.(7), yielding the following semi-discrete update equation, which can be advanced using an appropriate time integration scheme

$$\frac{d\hat{u}_j^{\delta D}}{dt} + \frac{1}{J_j} \left[ f_j^{\delta L} \frac{d\tilde{l}_0(r)}{dr} + \sum_{n=1}^{P+1} f_{jn}^{\delta D} \frac{d\tilde{l}_n(r)}{dr} + f_j^{\delta R} \frac{d\tilde{l}_{P+2}(r)}{dr} \right] = 0 \quad (27)$$

### 1. Extension to quadrilateral elements

The DFR scheme described can be extended to quadrilateral and hexahedral elements using tensor products as described in previous works by Huynh.<sup>1,8</sup> While the cited reference describes the extension of the standard FR method, the extension of the DFR variant follows an identical process, albeit with the correction procedure using correction polynomials switched to the corresponding Lagrange interpolation correction procedure used in the DFR method. While the details of this extension are left to the cited references, the salient aspects pertaining to the content in this paper are outlined below.

To extend the DFR method to quadrilateral elements, a mapping is introduced to a reference parent space element and a transformed update equation is solved within each element. This procedure mirrors that described in the one-dimensional case. The mapping for the  $j$ -th element,  $\Gamma_j$  is

$$\begin{pmatrix} x \\ y \end{pmatrix} = \Gamma_j(\xi, \eta) = \sum_{n=1}^K M_n(\xi, \eta) \begin{pmatrix} x_{n,j} \\ y_{n,j} \end{pmatrix} \quad (28)$$

where  $M_n(\xi, \eta)$  are the element shape functions and  $K$  is the number of points used to define the physical space element. In general,  $K$  can be varied to represent elements with higher order boundaries. For bilinear quadrilateral elements,  $K$  is equal to four, corresponding to the four corner nodes of the element. A visual depiction of this mapping can be seen in Figure 1.

After this transformation, all steps of the DFR method are carried out along lines in the parent element using tensor product formulations of the operations outlined in the one dimensional case. An important point to note is that the tensor product formulation allows for a direct extension of the one-dimensional DFR methodology to higher dimensions. The locations of the solution points, along with lines across which the operations are carried out are depicted in Figure 2.

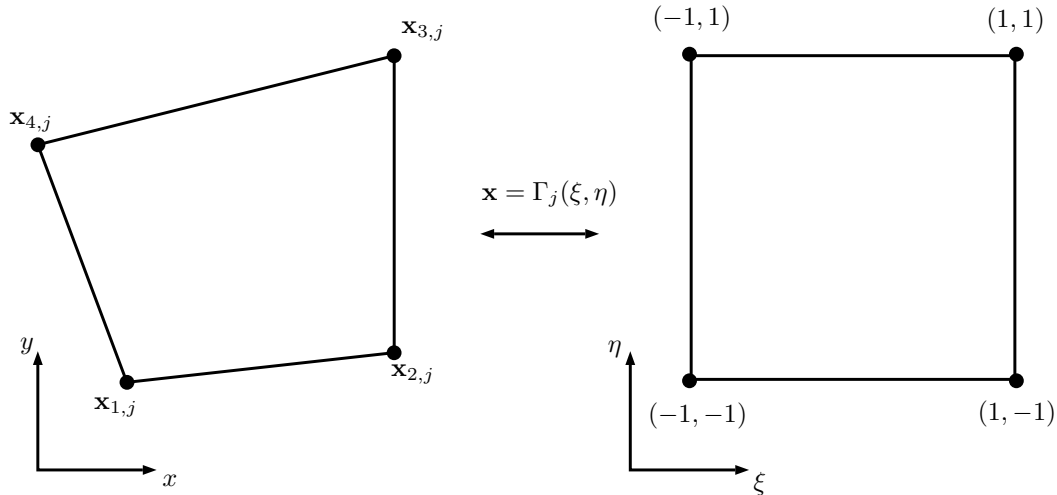
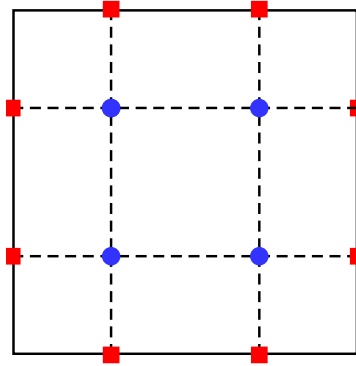


Figure 1. Mapping of physical space quadrilateral element to reference quadrilateral using isoparametric mapping  $\Gamma(\xi, \eta)$

## III. Extension of tensor-product formulation to triangular elements

Consider applying the mapping used for bilinear quadrilateral elements, described by Eq.(28) to a three node triangular element. Since the bilinear mapping requires four nodes, let one of the nodes of the triangle be repeated and



**Figure 2. Reference quadrilateral element with interior solution points (blue circles) and interface flux points (red squares). Tensor products of the 1D DFR operations are carried out along the dashed lines.**

treated as two distinct nodes to complete the sum. A visual depiction of this mapping can be seen in Figure 3. Due to the use of a repeated node, a non-physical edge, henceforth referred to as a ghost edge, is generated in the reference space. This edge corresponds to a zero length edge that is generated, in some sense, between the two instances of the repeated node. The solution points and flux points are located in the same locations as used in the quadrilateral case; however, several flux points are located along the ghost edge. These flux points will be correspondingly referred to as ghost flux points for the remainder of this description. The solution point, flux point, and ghost flux point locations on the parent space element and the mapping of these point locations back to the physical space triangular element are depicted in Figure 4. Note that on the physical triangular element, the ghost edge collapses back to a point collocated with the repeated node, with all corresponding ghost flux points collocated at this node location as well.

With this mapping procedure, the operations of the tensor product formulation of the DFR or standard FR scheme can be applied directly to triangular elements without modification; however, some special considerations must be made to handle the values at the ghost flux points. In the DFR scheme, values at the flux points are used to compute the common numerical solution and flux values used to generate the continuous solution  $u_j^\delta$  and continuous flux  $f_j^\delta$  and the formulas to compute these common values generally require values on both sides of the interface. Due to the location of the ghost flux points on element vertices, the opposing state from neighboring elements is not generally available as the flux points on non-collapsed edges are typically located on edge interiors. In other words, a collocated flux point from a neighboring element is typically not defined for the ghost flux points. This leads to the question of how to sensibly compute the necessary common interface values at these points.

The procedure to compute the common interface flux values  $f_j^{\delta L}$  or  $f_j^{\delta R}$  at the ghost flux points is straightforward. Since the ghost flux points are located on edges with zero length, it follows that there should be no flux passing through these points. Based on this argument, the common interface flux values at the ghost flux points can be set to zero. Note that this completes the framework for first-order fluxes, for which only common flux values and not common solution values need to be defined.

The complication arises in formulating a method to compute the common interface solution values  $u_j^{\delta L}$  or  $u_j^{\delta R}$  at the ghost flux points. Unlike with the common flux, the common solution can not be argued to be zero at the ghost flux points, as the value of the solution does not depend on the edge length. One possible method to handle this is to define additional flux points at element vertices, providing the necessary opposing states to be used to compute a true common solution value at the ghost flux point locations. While such a method would succeed at computing the required interface solution values, the creation of additional flux points at the vertices would add complexity and additional work to the computational implementation of the resulting scheme, detracting from the simplicity of the existing tensor product formulation that makes it attractive.

Thus, an alternative method is proposed. Instead of strictly requiring the computation of a true common solution value between elements at the ghost flux points, consider simply setting the common solution value used for a given element to be the value of the discontinuous solution,  $u_j^{\delta D}$  extrapolated to the ghost flux point location within said element. Using this methodology, the common solution value at ghost flux points is able to be multivalued between elements, as the extrapolated values of  $u_j^{\delta D}$  on each side of the element interface are generally not  $C^0$  continuous. As a result, the corrected solution  $u^\delta$  used to compute the auxiliary variable  $q^{\delta D}$  is no longer  $C^0$  continuous, which can impact the order of accuracy of the scheme. In the following section, results of several numerical test cases reveal that this degradation in continuity does not impact the resulting rate of convergence of the developed method for the tested

cases.

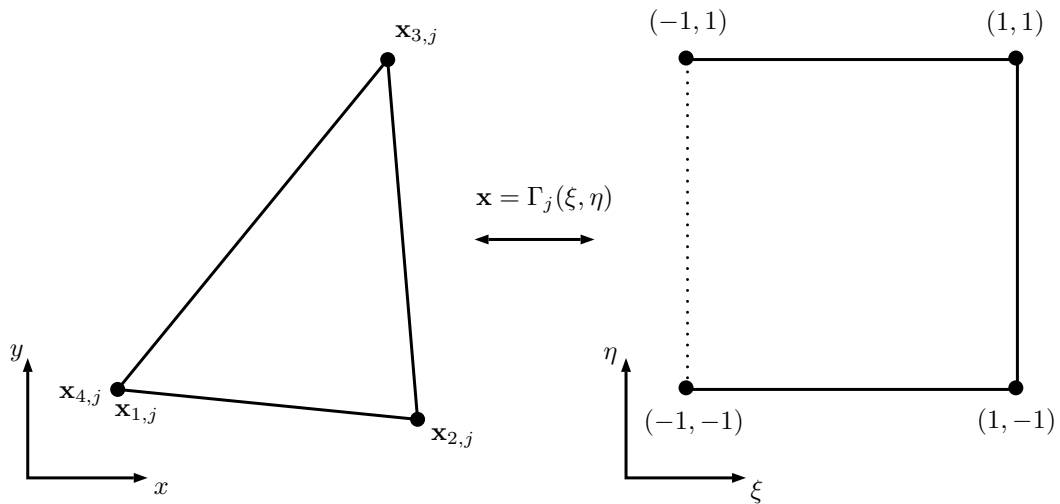


Figure 3. Mapping of physical space triangular element to reference quadrilateral using isoparametric mapping  $\Gamma(\xi, \eta)$  with the bottom left node repeated. The dotted line depicts a generated ghost edge.

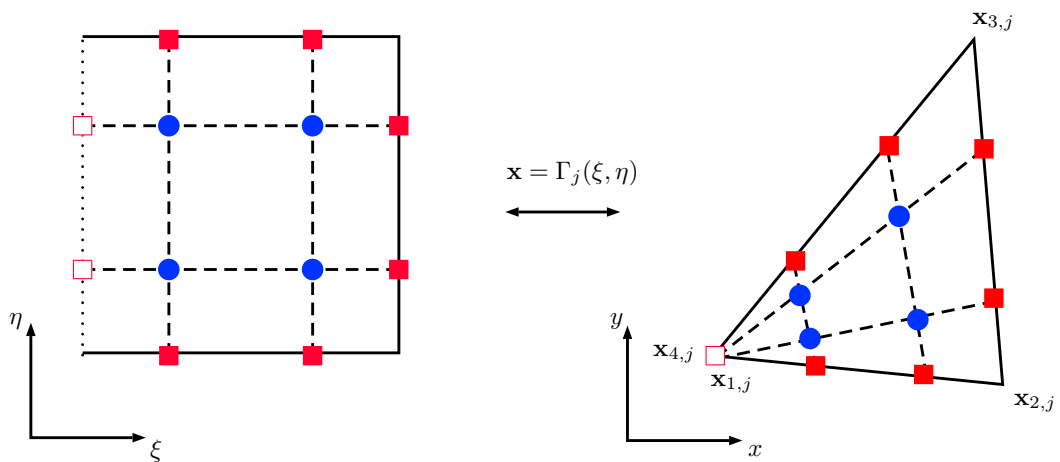


Figure 4. Reference quadrilateral and physical space triangular element with interior solution points (blue circles), interface flux points (red squares), interface ghost flux points (hollow red squares). Tensor products of the 1D DFR operations are carried out along the dashed lines.

## A. Higher Order Elements

To generate triangular elements with higher order boundaries, a similar collapsing procedure can be applied to some existing high order quadrilateral element types. In particular, applying the collapsing procedure to an eight-node serendipity quadrilateral, which has quadratic boundaries, is straightforward. The node indexing associated with this case can be seen in Figure 5. Note that in comparison to the collapsed bilinear quadrilateral, the collapsed vertex now contains three collocated nodes. In extending this procedure to higher order elements, special care must be taken to ensure that after the mapping is applied, no solution points lie in fully collapsed regions, where the determinant of the element jacobian goes to zero.

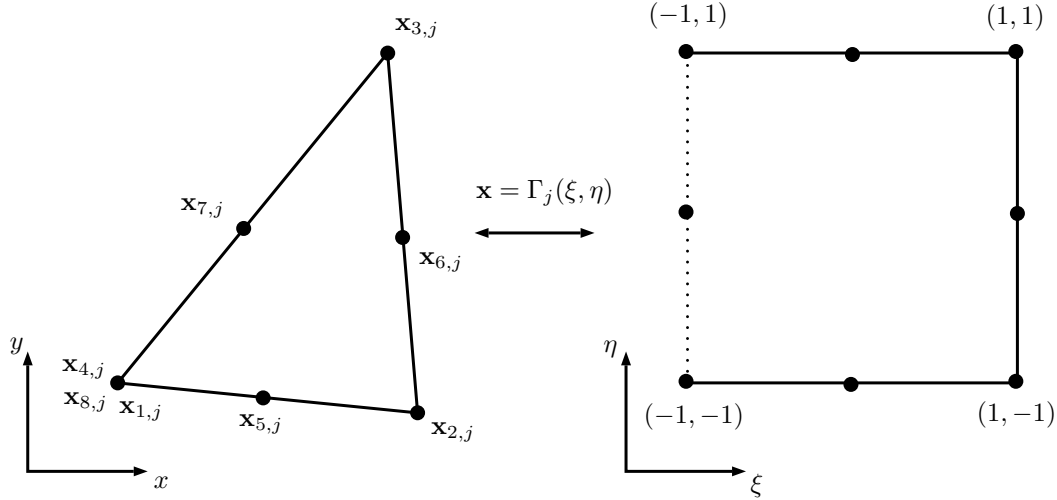


Figure 5. Mapping of physical space six-node quadratic triangular element to reference 8-node serendipity quadrilateral.

## IV. Results

### A. 2D Advection-Diffusion Problem

#### 1. Test Case Description

Consider the 2D linear advection-diffusion equation

$$\frac{\partial u}{\partial t} + \nabla \cdot (\mathbf{a}u - D\nabla u) = 0 \quad (29)$$

defined on a square domain  $\Omega = \{x, y \mid -1 \leq x \leq 1, -1 \leq y \leq 1\}$  where  $u = u(\mathbf{x}, t)$  is a scalar,  $\mathbf{a} = (a_x, a_y)$  is a constant vector of the convective wavespeeds, and  $D$  is the diffusivity. The initial condition is defined as

$$u(\mathbf{x}, 0) = \sin(\pi x) \sin(\pi y) \quad (30)$$

and periodic boundary conditions are imposed on all sides of the domain. This choice of initial and boundary conditions yields an exact solution of the form

$$u^e(\mathbf{x}, t) = e^{-2D\pi^2 t} \sin(\pi(x - a_x t)) \sin(\pi(y - a_y t)) \quad (31)$$

Two variations of this test case were solved using the DFR method: a pure diffusion problem with  $D = 0.1$  and  $\mathbf{a} = (0, 0)$  and an advection-diffusion problem with  $D = 0.1$  and  $\mathbf{a} = (1, 1)$ . For each case, the solution was advanced in time using a standard fourth order Runge Kutta scheme until  $t = 0.25$ . To compute the common convective flux, the Rusanov flux was used and for the common diffusive flux and solution, the CF method was used with the penalty parameter  $\tau = 1$ . At ghost flux points, the treatment described in the previous section is used to set all common values.



## 2. Order of Accuracy

To illustrate the nature of the collapsed-edge procedure on triangular elements, the described test case was carried out for both quadrilateral and triangular meshes of increasing resolution to quantify the resulting order of accuracy. The quadrilateral meshes used were  $N \times N$  cartesian grids of  $N^2$  elements with  $N = 8, 16, 32, 64$  and  $128$ . The triangular meshes were generated directly from the quadrilateral meshes by splitting each quadrilateral element into two triangles, resulting in meshes with  $2N^2$  elements. Note that since this is an unsteady simulation, the timestep was set to be sufficiently small such that errors introduced from the timestepping scheme were small relative to spatial errors. An example of the meshes used can be seen in Figure 6.

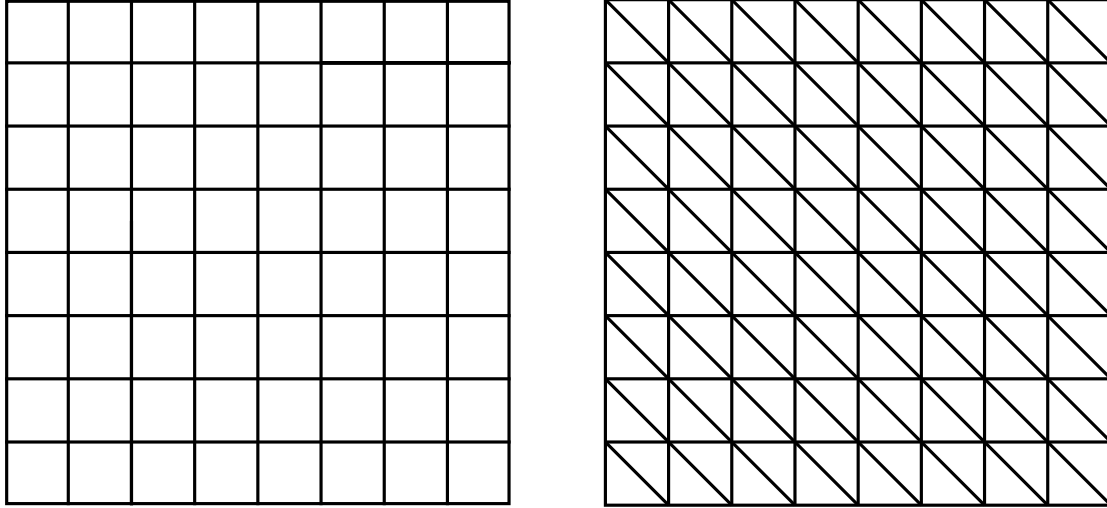


Figure 6. Quadrilateral mesh with  $N = 8$  and corresponding triangular mesh

To determine the order of accuracy, the following error norms were computed at the final time  $t = 0.25$

$$E_{L2} = \sqrt{\sum_{j=1}^{N_{eles}} \int_{\Omega_j} (u^e - u_j^{\delta D})^2 d\Omega_j} \quad (32)$$

$$E_{H1} = \sqrt{\sum_{j=1}^{N_{eles}} \int_{\Omega_j} \left[ (u^e - u_j^{\delta D})^2 + \left( \frac{\partial u^e}{\partial x} - \frac{u_j^{\delta D}}{\partial x} \right)^2 + \left( \frac{\partial u^e}{\partial y} - \frac{\partial u_j^{\delta D}}{\partial y} \right)^2 \right] d\Omega_j} \quad (33)$$

with integrals over each element  $\Omega_n$  computed using quadrature rules of sufficient strength. The expected order of accuracy for  $E_{L2}$ , the L2 error norm of the solution, is  $P + 1$  while the expected order of accuracy for  $E_{H1}$ , the H1 error norm, which includes the solution and gradient, is  $P$ .

Results were obtained for both test cases using  $P = 2$  and  $P = 3$  solution representations on all listed meshes. Plots of  $E_{L2}$  vs grid spacing, defined as  $h = 2/N$ , for all tested cases can be seen in Figures 7 and 9. Similar plots of  $E_{H1}$  vs grid spacing can be observed in Figures 8 and 10. Additionally, tabulated data and the computed order of accuracy between grids can be found in Tables 1 through 4. The key observation to be drawn from these results is that the collapsed edge formulation on triangles preserves the expected order of accuracy in both  $E_{L2}$  and  $E_{H1}$  norms. An additional observation is that the magnitude of the errors using triangular grids, in both cases, is slightly higher than that of the quadrilateral grids.

## 3. Stable Timestep Limit

To understand the impact of the collapsed-edge formulation on the stable timestep limit, additional simulations of the advection-diffusion test case ( $D = 0.1$ ,  $\mathbf{a} = (1, 1)$ ) were carried out and approximate maximum stable timesteps,  $\Delta t_{max}$ , were determined. A timestep was considered stable if the solution remained bounded after a full period through the domain. For these cases, two additional quadrilateral grids were used with  $N = 23$  and  $45$ . For cases

using triangular elements, the previously generated grids using  $N = 16$  and  $32$  were used. These grid sizes were selected as they result in grids of approximately equivalent number of elements between the two element types. The approximate maximum stable timestep on these grids, using  $P = 2$  and  $P = 3$  solution representations, can be seen in Table 5.

These results elucidate one of the primary issues hindering the collapsed edge formulation, namely, a strong reduction in the stable timestep size with increasing grid resolution and polynomial order, as compared to the original scheme on pure quadrilateral grids. One potential cause of this reduction is that the collapsed edge formulation tends to cluster the solution and flux points, especially near the collapsed vertex. If one considers the underlying one-dimensional operations taking place along the lines of the tensor product, it can be noted that the lines parallel to the collapsed edge experience a strong contraction, with the length of these lines becoming smaller nearer to the collapsed vertex. This contracting is only worsened with increasing polynomial order, as new solution and flux points tend to approach the edges/corners of the element. This contraction is likely one of the limiters impacting the resulting stable timestep of the scheme. Potential remedies for this limitation are the use of alternative solution points, located further from the element boundaries or coupling the collapsed edge formulation with an implicit timestepping scheme.

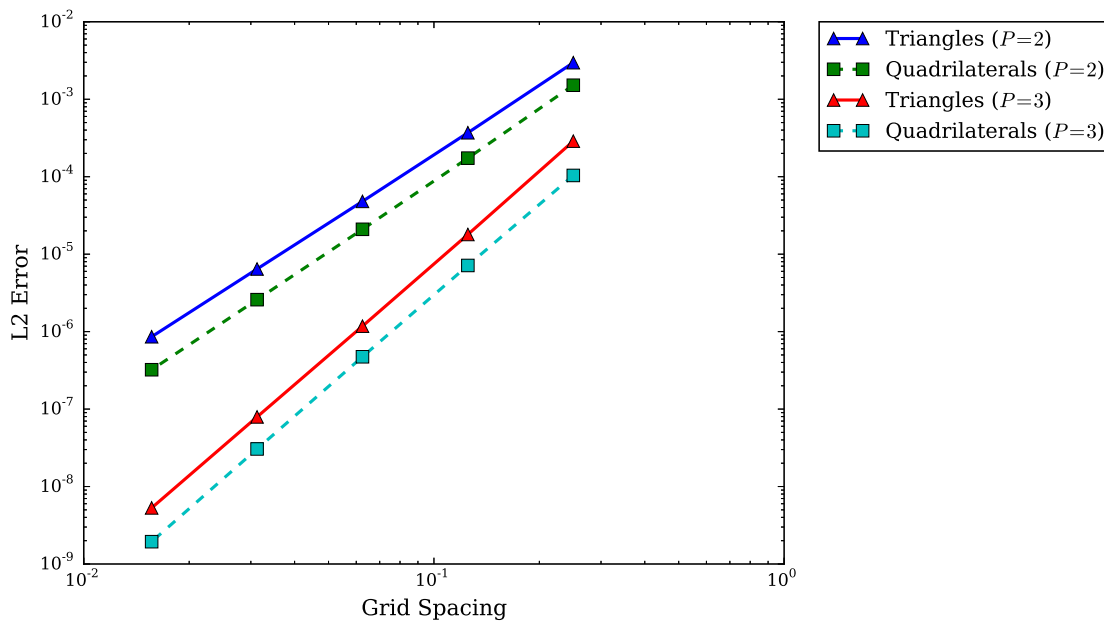


Figure 7.  $E_{L2}$  vs grid spacing for pure diffusion problem ( $D = 0.1, \mathbf{a} = 0$ )

Triangles, $P = 2$			Triangles, $P = 3$			Quadrilaterals, $P = 2$			Quadrilaterals, $P = 3$		
$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order
8	2.967e-3	—	8	2.879e-4	—	8	1.517e-3	—	8	1.038e-4	—
16	3.694e-4	3.01	16	1.796e-5	4.00	16	1.738e-4	3.13	16	7.149e-6	3.86
32	4.803e-5	2.94	32	1.176e-6	3.93	32	2.094e-5	3.05	32	4.735e-7	3.92
64	6.436e-6	2.90	64	7.920e-8	3.89	64	2.586e-6	3.02	64	3.055e-8	3.95
128	8.587e-7	2.91	128	5.297e-9	3.90	128	3.221e-7	3.01	128	1.941e-9	3.98

Table 1.  $E_{L2}$  and Order of Accuracy results for pure diffusion problem ( $D = 0.1, \mathbf{a} = 0$ )

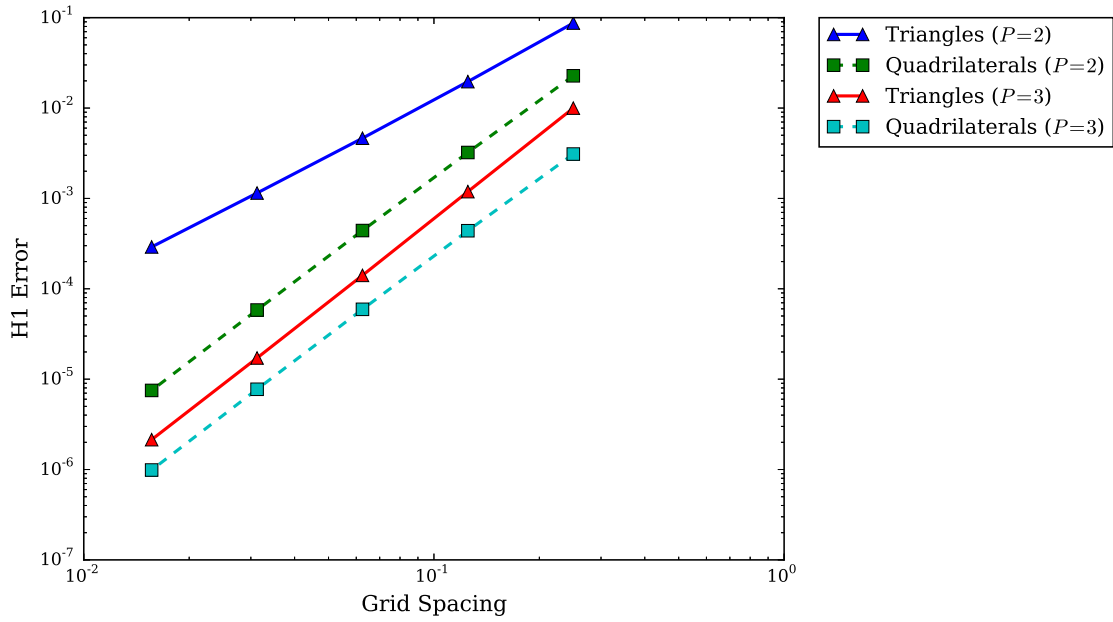


Figure 8.  $E_{H1}$  vs grid spacing for pure diffusion problem ( $D = 0.1, \mathbf{a} = 0$ )

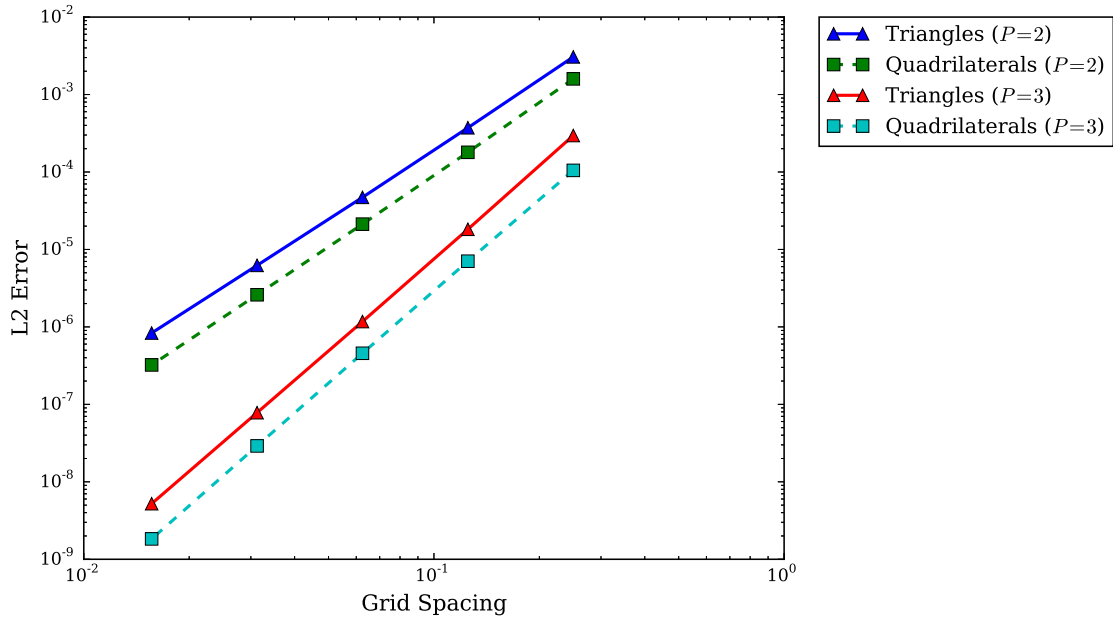


Figure 9.  $E_{L2}$  vs grid spacing for advection-diffusion problem ( $D = 0.1, \mathbf{a} = (1, 1)$ )

Triangles, $P = 2$			Triangles, $P = 3$			Quadrilaterals, $P = 2$			Quadrilaterals, $P = 3$		
$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order
8	8.721e-2	—	8	9.988e-3	—	8	2.274e-2	—	8	3.099e-3	—
16	1.961e-2	2.15	16	1.189e-3	3.07	16	3.223e-3	2.82	16	4.390e-4	2.82
32	4.636e-3	2.08	32	1.408e-4	3.08	32	4.411e-4	2.87	32	5.927e-5	2.89
64	1.147e-3	2.01	64	1.715e-5	3.04	64	5.825e-5	2.92	64	7.729e-6	2.94
128	2.904e-4	1.98	128	2.140e-6	3.00	128	7.504e-6	2.96	128	9.877e-7	2.97

Table 2.  $E_{H1}$  and Order of Accuracy results for pure diffusion problem ( $D = 0.1, \mathbf{a} = 0$ )

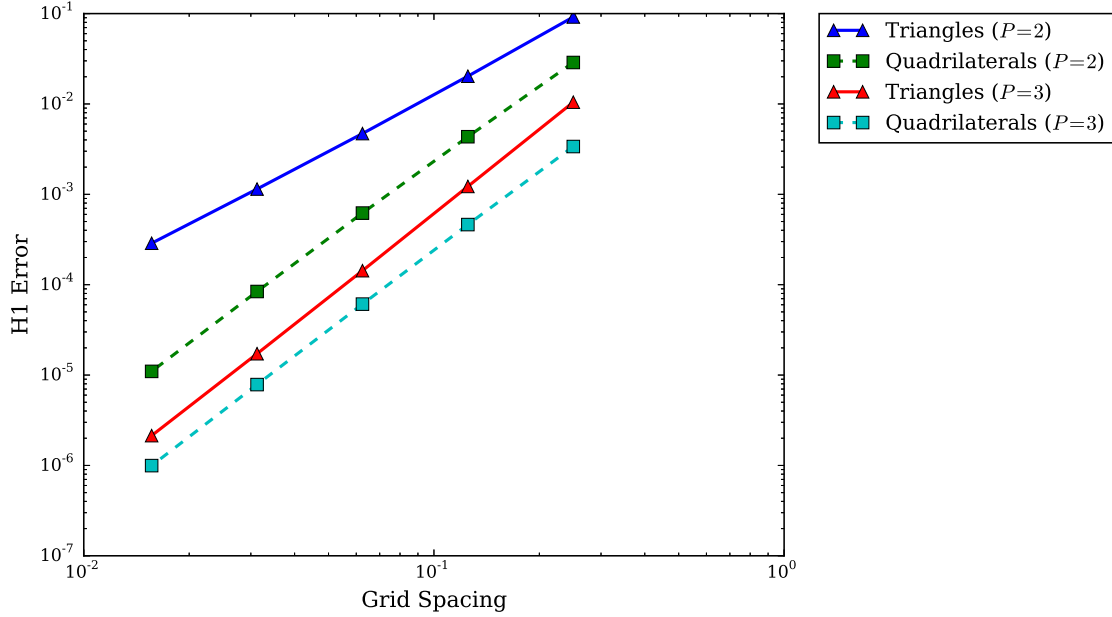


Figure 10.  $E_{H1}$  vs grid spacing for advection-diffusion problem ( $D = 0.1, \mathbf{a} = (1, 1)$ )

Triangles, $P = 2$			Triangles, $P = 3$			Quadrilaterals, $P = 2$			Quadrilaterals, $P = 3$		
$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order
8	3.060e-3	—	8	2.957e-4	—	8	1.592e-3	—	8	1.049e-4	—
16	3.723e-4	3.04	16	1.821e-5	4.02	16	1.795e-4	3.15	16	7.051e-6	3.90
32	4.704e-5	2.98	32	1.172e-6	3.96	32	2.123e-5	3.08	32	4.566e-7	3.95
64	6.226e-6	2.92	64	7.804e-8	3.91	64	2.597e-6	3.03	64	2.904e-8	3.97
128	8.337e-7	2.90	128	5.219e-9	3.90	128	3.225e-7	3.01	128	1.831e-9	3.99

Table 3.  $E_{L2}$  and Order of Accuracy results for advection-diffusion problem ( $D = 0.1, \mathbf{a} = (1, 1)$ )

Triangles, $P = 2$			Triangles, $P = 3$			Quadrilaterals, $P = 2$			Quadrilaterals, $P = 3$		
$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order
8	9.174e-2	—	8	1.047e-2	—	8	2.876e-2	—	8	3.380e-3	—
16	2.029e-2	2.17	16	1.223e-3	3.10	16	4.340e-3	2.73	16	4.626e-4	2.87
32	4.702e-3	2.11	32	1.429e-4	3.10	32	6.207e-4	2.81	32	6.098e-5	2.92
64	1.144e-3	2.04	64	1.722e-5	3.05	64	8.408e-5	2.88	64	7.844e-6	2.96
128	2.876e-4	1.99	128	2.137e-6	3.01	128	1.098e-5	2.94	128	9.952e-7	2.98

Table 4.  $E_{H1}$  and Order of Accuracy results for advection-diffusion problem ( $D = 0.1, \mathbf{a} = (1, 1)$ )

Triangles, $P = 2$		Triangles, $P = 3$		Quadrilaterals, $P = 2$		Quadrilaterals, $P = 3$	
$N$	$\Delta t_{max}$	$N$	$\Delta t_{max}$	$N$	$\Delta t_{max}$	$N$	$\Delta t_{max}$
16	1.20e-4	16	1.63e-5	23	1.31e-3	23	5.22e-4
32	3.00e-5	32	4.07e-6	45	3.75e-4	45	1.45e-4

Table 5. Approximate maximum timestep for advection-diffusion problem ( $D = 0.1, \mathbf{a} = (1, 1)$ )

## B. 2D Planar Couette Flow

### 1. Test Case Description

To investigate the performance of the collapsed-edge formulation for non-linear fluxes, simulations of the compressible Navier-Stokes equations, specifically a 2D planar Couette flow problem were carried out. The compressible Navier-Stokes equations in 2D can be expressed as

$$\frac{\partial U}{\partial t} + \nabla \cdot \mathbf{F}(U) = \nabla \cdot \mathbf{G}(U, \nabla U) \quad (34)$$

where  $U$  is a vector of conservative variables,  $\mathbf{F}(U)$  is a vector of the inviscid flux terms, and  $\mathbf{G}(U, \nabla U)$  is a vector of the viscous flux terms

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{bmatrix} \quad F_x = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E + p) \end{bmatrix} \quad F_y = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E + p) \end{bmatrix}$$

$$G_x = \mu \begin{bmatrix} 0 \\ 2\frac{\partial u}{\partial x} - \frac{2}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) \\ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \\ u\left[2\frac{\partial u}{\partial x} - \frac{2}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)\right] + v\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) + \frac{C_p}{Pr} \frac{\partial T}{\partial x} \end{bmatrix}$$

$$G_y = \mu \begin{bmatrix} 0 \\ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \\ 2\frac{\partial v}{\partial y} - \frac{2}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) \\ v\left[2\frac{\partial v}{\partial y} - \frac{2}{3}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)\right] + u\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) + \frac{C_p}{Pr} \frac{\partial T}{\partial y} \end{bmatrix}$$

$\rho$  is the fluid density,  $u$  and  $v$  are the x and y velocities respectively,  $p$  is the pressure,  $E$  is the total energy,  $\mu$  is the dynamic viscosity,  $C_p$  is the specific heat,  $Pr$  is the Prandtl number, and  $T$  is the temperature.

For the planar Couette flow case, consider the flow between two parallel plates, separated by a distance  $H$  with constant viscosity,  $\mu = \text{const}$ . The lower plate is kept stationary at a fixed temperature  $T_l$ . The upper plate travels in the x-direction with constant velocity  $U_u$  and is held at a fixed temperature  $T_u$ . Under these assumptions, the analytical solution for the velocity of the resulting steady-state flow field is

$$u = \frac{U_u y}{H} \quad v = 0 \quad (35)$$

This test case was simulated on a rectangular domain  $\Omega = \{x, y | -1 \leq x \leq 1, 0 \leq y \leq 1\}$  with fixed and moving isothermal boundary conditions for the lower ( $y = 0$ ) and upper plate ( $y = 1$ ) respectively and periodic boundaries at the left ( $x = -1$ ) and right ( $x = 1$ ) boundaries. The upper plate velocity  $U_u$  was set with a Mach number of 0.2 and the flow Reynolds number was set to 200. The temperatures of the lower and upper plate were 300K and 315K respectively. The flow field was initialized with the velocity of the top plate and the simulation was marched forward in time to steady state using a standard fourth order Runge Kutta scheme.

### 2. Order of Accuracy

Similar to the advection-diffusion test case, the described planar Couette flow case was carried out on both quadrilateral and triangular meshes of increasing resolution to quantify the resulting order of accuracy. For these cases, the quadrilateral meshes used were  $2N \times N$  cartesian grids of  $2N^2$  elements with  $N = 2, 4, 8, \text{ and } 16$ . Again, triangular meshes were generated directly from these meshes by splitting the quadrilaterals, resulting in meshes with  $4N^2$  elements.

Results were obtained for this test case using  $P = 2$  and  $P = 3$  solution representations on all listed meshes and the same error norms, described by Eqs. (32) and (33) were computed on the steady state x-velocity solution achieved on each grid relative to the expected analytical x-velocity solution, given by Eq. (35). Plots of the error norms vs grid

spacing can be observed in Figures 11 and 12 and tabulated data can be found in Tables 6 and 7. Note that in this case, the grid spacing was defined as  $h = 1/N$ .

From these results, it can be observed that the collapsed-edge formulation on triangle maintains the expected order of accuracy in the cases tested in both  $E_{L2}$  and  $E_{H1}$ . Additionally, the error magnitudes achieved, with respect to the grid spacing, is very similar for both methods.

Triangles, $P = 2$			Triangles, $P = 3$			Quadrilaterals, $P = 2$			Quadrilaterals, $P = 3$		
$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order	$N$	$E_{L2}$	Order
2	6.852e-5	—	2	6.747e-7	—	2	8.747e-5	—	2	6.503e-7	—
4	5.849e-6	3.55	4	2.560e-8	4.72	4	6.314e-6	3.79	4	3.099e-8	4.39
8	6.917e-7	3.08	8	1.433e-9	4.16	8	7.922e-7	2.99	8	1.943e-9	4.00
16	8.249e-8	3.07	—	—	—	16	1.041e-7	2.93	—	—	—

Table 6.  $E_{L2}$  and Order of Accuracy results for Couette flow problem

Triangles, $P = 2$			Triangles, $P = 3$			Quadrilaterals, $P = 2$			Quadrilaterals, $P = 3$		
$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order	$N$	$E_{H1}$	Order
2	7.456e-4	—	2	1.263e-5	—	2	8.028e-4	—	2	1.063e-5	—
4	1.549e-4	2.27	4	1.076e-6	3.55	4	1.595e-4	2.33	4	1.138e-6	3.22
8	3.612e-5	2.10	8	1.155e-7	3.22	8	3.853e-5	2.05	8	1.421e-7	3.00
16	8.411e-6	2.10	—	—	—	16	9.198e-6	2.07	—	—	—

Table 7.  $E_{H1}$  and Order of Accuracy results for Couette flow problem

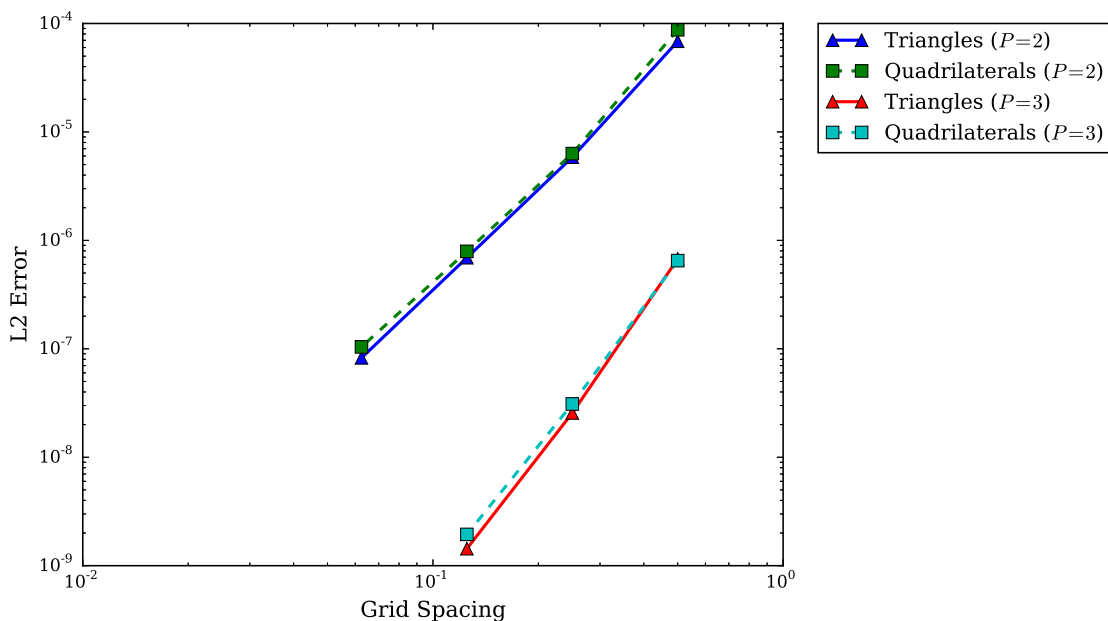


Figure 11.  $E_{L2}$  vs grid spacing for Couette flow problem

## C. 2D Viscous Flow over an NACA0012 airfoil

### 1. Test Case Description

As noted in the previous section, the collapsed edge methodology can be extended to higher order element representations. Additionally, since the developed scheme treats triangular elements as quadrilaterals, the scheme can naturally operate on mixed grids of quadrilaterals and triangles with no additional modification.

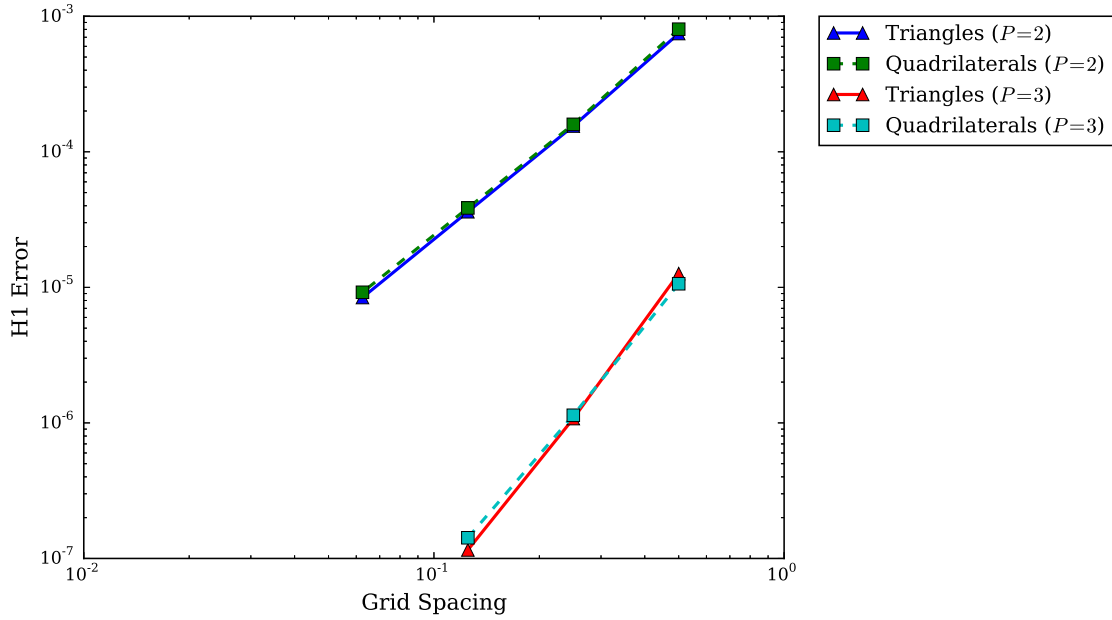


Figure 12.  $E_{H1}$  vs grid spacing for Couette flow problem

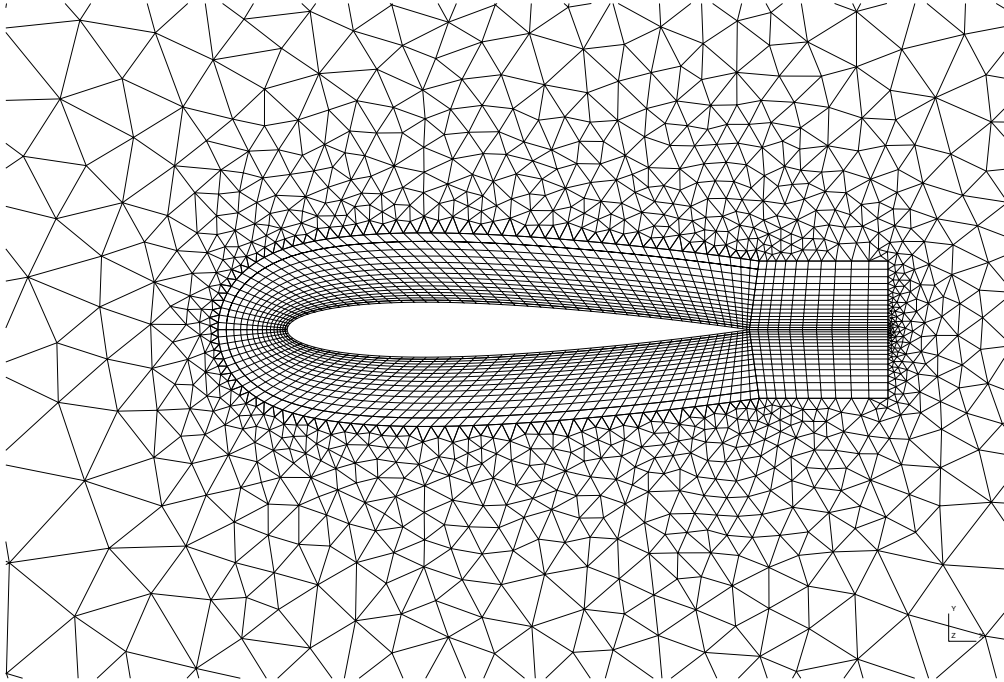
To demonstrate this potential as well as show the efficacy of the scheme in solving problems on a realistic geometry, a test case solving the Navier-Stokes equations for viscous flow around a NACA0012 was carried out. The freestream Mach number was  $M = 0.5$ , the Reynolds number of the flow,  $Re = 5000$  and the angle of attack was set as  $\alpha = 0^\circ$ . The mixed mesh used for this test case was comprised of second-order quadrilateral and triangular elements, represented with uncollapsed and collapsed eight-node serendipity element representations respectively. There were 1,773 quadrilateral elements and 3,796 triangular elements, with 5,569 elements in total. The mesh near the airfoil surface can be observed in Figure 13. The NACA0012 airfoil geometry was generated using a modified formula with a sharp trailing edge which can be found in Ref. 9. A no-slip adiabatic wall boundary condition was applied at the airfoil surface with a subsonic characteristic boundary condition applied at the farfield boundary. For this case, the dynamic viscosity,  $\mu$ , was set using Sutherland’s law. This simulation was marched forward in time using an explicit standard fourth-order Runge-Kutta timestepping scheme until a periodic steady-state solution was achieved. At this Reynolds number, the flow separates near the trailing edge of the airfoil, causing the development of a small circulation region in the near wake region, followed by the development of an unsteady vortex sheet further downstream<sup>12,13</sup>. For this case, polynomials with  $P = 3$  were used to represent the solution.

## 2. Results

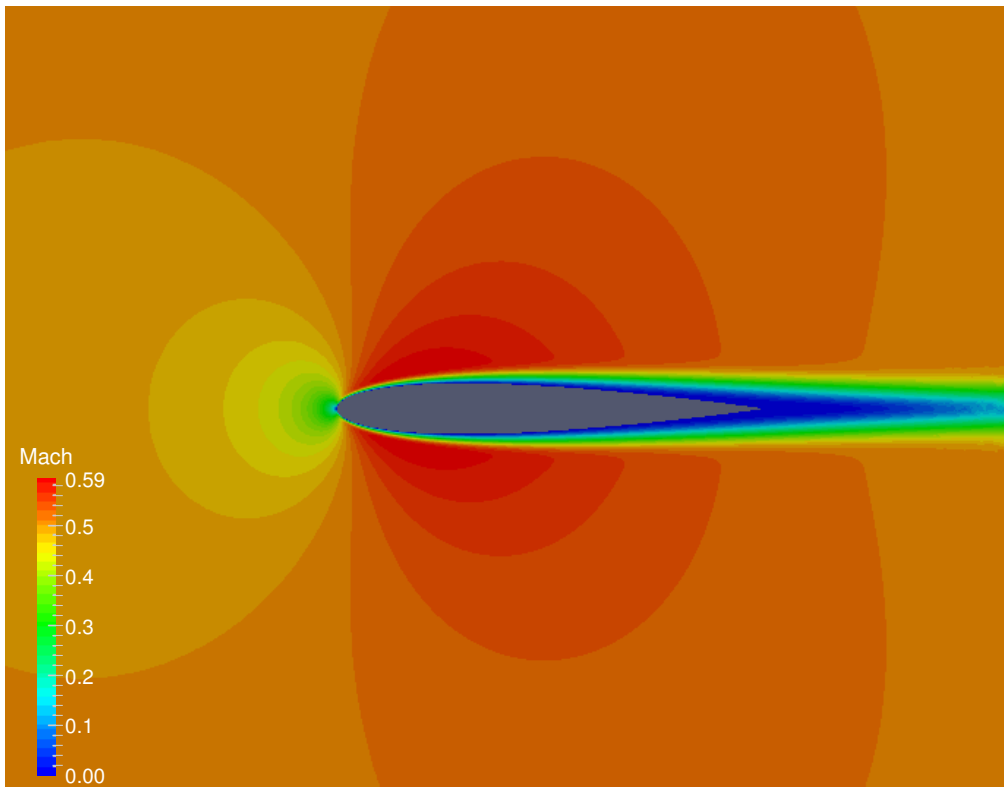
The predicted drag from this simulation, separated into pressure and viscous components can be found in Table 8, along with the results of similar studies. The results obtained on this single grid agree well with with previously published results, with improvement in accuracy to be gained from further mesh refinement or adaptation. This result indicates that the collapsed-edge formulation can be used effectively for these types of problems.

Scheme	Order of accuracy	Number of elements	$C_{d,p}$	$C_{d,v}$
Collapsed-edge DFR (current)	4	5569	0.02292	0.03394
VCJH-DG FR <sup>9</sup>	4	4650	0.02222	0.03348
Bassi-Rebay (BR1) DG <sup>12</sup>	4	1024	0.02208	0.03303
Cell-vertex finite volume <sup>14</sup>	2	16384	0.02270	0.03270
Cell-centered finite volume <sup>14</sup>	2	16384	0.02256	0.03301

Table 8. Pressure ( $C_{d,p}$ ) and viscous ( $C_{d,v}$ ) drag results for the NACA0012 test case ( $M = 0.5$ ,  $Re = 5000$ ,  $\alpha = 0^\circ$ ) from various sources



**Figure 13. Nearfield mesh for NACA0012 test case**



**Figure 14. Contours of Mach number for NACA0012 test case**



## V. Conclusion

In this paper, a direct extension the the tensor product based FR method on quadrilateral elements to triangles has been described. Numerical tests have indicated empirically that the developed methodology maintains the expected order of accuracy for linear and nonlinear problems and that the scheme can produce results of similar accuracy to existing methods. Moreover, the completed work on triangles identifies a framework that may be generalized to develop similar extensions of three-dimensional tensor product FR formulations on hexahedral elements to additional elements such as tetrahedra and pyramids. However, this study has also indicated that this method does suffer from a reduction in the stable timestep limit and efforts to address this issue will be key in the development of this method moving forward. As discussed previously, modification of the solution point locations within an element or coupling with an implicit timestepping method would be potential options in addressing this issue. The coupling of an implicit time stepping method to this scheme may prove to be more straightforward, due to the simplicity of the method relative to the existing FR procedure for triangular elements.

## Acknowledgments

The authors would like to acknowledge the support for this work provided by the Stanford Graduate Fellowship program.

## References

- <sup>1</sup>Huynh, H., "A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods," *AIAA paper*, Vol. 4079, 2007, pp. 2007.
- <sup>2</sup>Castonguay, P., Vincent, P. E., and Jameson, A., "A new class of high-order energy stable flux reconstruction schemes for triangular elements," *Journal of Scientific Computing*, Vol. 51, No. 1, 2012, pp. 224–256.
- <sup>3</sup>Williams, D. and Jameson, A., "Energy Stable Flux Reconstruction Schemes for Advection–Diffusion Problems on Tetrahedra," *Journal of Scientific Computing*, 2013, pp. 1–39.
- <sup>4</sup>Warburton, T., *Spectral/hp methods on polymorphic multidomains: Algorithms and applications*, Ph.D. thesis, Brown University, 1999.
- <sup>5</sup>Karniadakis, G. and Sherwin, S., *Spectral/hp element methods for computational fluid dynamics*, Oxford University Press, 2nd ed., 2005.
- <sup>6</sup>Romero, J. and Jameson, A., "A Simplified Formulation of the Flux Reconstruction Method," *Journal of Scientific Computing*, In Review.
- <sup>7</sup>Vincent, P. E., Castonguay, P., and Jameson, A., "A new class of high-order energy stable flux reconstruction schemes," *Journal of Scientific Computing*, Vol. 47, No. 1, 2011, pp. 50–72.
- <sup>8</sup>Huynh, H., "A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion," *AIAA paper*, Vol. 403, 2009, pp. 2009.
- <sup>9</sup>Williams, D., Castonguay, P., Vincent, P., and Jameson, A., "An extension of energy stable flux reconstruction to unsteady, non-linear, viscous problems on mixed grids," *20th AIAA Computational Fluid Dynamics Conference Proceedings*, 2011.
- <sup>10</sup>Hesthaven, J. and Warburton, T., *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, Vol. 54, Springer, 2007.
- <sup>11</sup>Cockburn, B. and Shu, C., "The local discontinuous Galerkin method for time-dependent convection-diffusion systems," *SIAM J. Numer. Anal.*, Vol. 35, 1998, pp. 2440–2463.
- <sup>12</sup>Bassi, F. and Rebay, S., "A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations," *Journal of Computational Physics*, Vol. 131, No. 2, 1997, pp. 267 – 279.
- <sup>13</sup>Gassner, G., Lörcher, F., and Munz, C., "A discontinuous Galerkin scheme based on a space-time expansion II. Viscous flow equations in multi dimensions," *Journal of Scientific Computing*, Vol. 34, No. 3, 2008, pp. 260–286.
- <sup>14</sup>Radespiel, R. and Swanson, R. C., "An Investigation of Cell Centered and Cell Vertex Multigrid schemes for the Navier-Stokes Equations," *AIAA Paper*, Vol. 89, No. 0453, 1989.

**This article has been cited by:**

1. Jerry E. Watkins, Joshua Romero, Antony Jameson Multi-GPU, Implicit Time Stepping for High-order Methods on Unstructured Grids . [[Citation](#)] [[PDF](#)] [[PDF Plus](#)]