

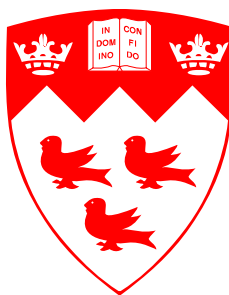
TARGET-BASED ADAPTATION USING THE DISCONTINUOUS PETROV-GALERKIN METHOD

Philip Zwanenburg
Bachelor of Engineering

Department of Mechanical Engineering
McGill University
Montreal, Quebec

July 20, 2016

A thesis submitted to
McGill University
in partial fulfillment of the requirements of the degree of
Doctor of Philosophy



© Philip Zwanenburg, 2016

DEDICATION

This thesis is dedicated to those who have fuelled my interest in numerical analysis through their genius, creativity and passion. **Include best graphic or logo**

ACKNOWLEDGEMENTS

ToBeDone

Don't forget NSERC+McGill Funding

ABSTRACT

ToBeDone

ABRÉGÉ

ToBeDone

TABLE OF CONTENTS

DEDICATION	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ABRÉGÉ	v
LIST OF TABLES	viii
LIST OF FIGURES	ix
1 Introduction	1
1.1 Motivation & Open Topics	1
1.1.1 Computational Complexity	3
1.1.2 Restrictive CFL Condition and Suboptimal Dissipation and Dispersion Properties	5
1.1.3 Treatment of Complex Geometry	7
1.1.4 Optimal Test Spaces	10
1.2 Thesis Overview	12
1.3 Accompanying Code	12
1.4 Testing	13
2 Governing Equations	16
2.1 Thermodynamics	16
2.2 Navier-Stokes Equations	16
2.3 Boundary Conditions	16
2.4 Non-Dimensionalization	16
3 Numerical Methods	17
3.1 Discontinuous Petrov-Galerkin Method	17

3.2	Adjoint Formulation	19
4	Computational Complexity Improvements	20
4.1	Sum Factorization	20
4.1.1	Representative Example	20
4.1.2	Timing Comparisons	25
4.2	Representation of Operators in Fourier Space	25
5	TESTING FOOTNOTES	26
	Appendix A	27
	Appendix B	28
	REFERENCES	29

LIST OF TABLES

<u>Table</u>		<u>page</u>
1-1	Errors and Convergence Orders - $PI_c = 2P$ (GL/Cools), $PI_{cf} = 2P$ (GL), $PG = P$, EFE = 1	13

LIST OF FIGURES

<u>Figure</u>		<u>page</u>
1-1	Supersonic Vortex Structured Mixed Mesh	14
1-2	Mach Number Distribution: $P = 4$, $PG = 4$, $PF = 5$, $PFr = 8$, $PIs = 8$, $PIc = 11$, $PIsf = 8$, $PIcf = 12$, $EFE = 1$	15

CHAPTER 1

Introduction

1.1 Motivation & Open Topics

The use of computational fluid dynamics (CFD) tools for the numerical analysis of fluid flows has significantly reduced costs associated with aerodynamic design over the past several decades. As computing systems have become increasingly powerful, there has been a corresponding advance in the equations employed for flow simulation (initially beginning with potential equations and now using the full Navier-Stokes equations) as well as in the resolution of complex flow phenomena.

Finite volume methods currently represent the industry and, to a great extent, academic standard for the solution of partial differential equations (PDEs) in the aerospace community. This is in large part due to their robustness in the presence of steep gradients in the flow as well as their ability to model geometrically complex objects as a result of the possibly unstructured nature of the volumes. However, finite volume schemes are inherently second order accurate, making their usage inefficient when flow solutions are smooth. In these cases, it is advantageous to use higher-order functions for the representation of the solution, resulting in solution convergence at greater rates than second order as well as necessitating fewer degrees of freedom (DOF) to obtain similar levels of solution resolution. One would thus like to employ spectral methods because of their exponential convergence properties, however

these methods are unsuitable in the presence of complex geometry. Pseudo-spectral methods (which can be thought of as a spectral method being used within each control volume of a finite volume scheme) have become the most popular choice when attempting to address the concern of employing high-order accurate (greater than second order) solution representation in the presence of complex geometry.

The discontinuous Galerkin (DG) method, initially proposed by Reed et al. [1] and subsequently analyzed for the solution of systems of conservation laws [2, 3, 4, 5, 6], has become the most popular choice of high-order scheme in the CFD community. Despite its widespread usage, there are still several major issues related to the standard DG method:

- High computational complexity with increasing order of accuracy;
- Restrictive Courant-Friedrichs-Lewy (CFL) conditions for explicit time stepping and suboptimal dissipation and dispersion characteristics for wave propagation;
- Difficulty in generating meshes for complex geometrical objects as well as in converting low-order meshes provided by standard mesh generators to high-order meshes;
- Improper test space for best approximation in the energy norm when applied to hyperbolic PDEs.

The last of these issues provides the motivation for the thesis, however, many difficulties arising from the other listed problems have already been encountered simply

while setting up the framework to begin the novel research. With particular regard to the representation of complex curved geometry, it has been observed that improper geometry treatment can contribute significant error to a flow simulation, potentially eliminating any advantages obtained from the use of better test spaces, for example.

1.1.1 Computational Complexity

Provide links to sections where related results are provided if available

Significant progress has been made with regard to the improvement of the DG method's computational complexity, notably through the exploitation of sum factorization techniques, originally proposed by Orszag [7] and now employed in the tensor-product Spectral element method (SEM) of Kopriva [8] and for general elements using collapsed tensor-product spaces by Karniadakis et al. [9]. When explicit methods are used, use of the sum factorization technique reduces the growth in computational complexity from $O(N^{2d})$ to $O(N^{d+1})$ where d is the dimension of the problem and $N = P + 1$ where P is the order of the method. Savings for implicit schemes are even greater and further, the appropriate decomposition of element bases into corner, edge, facet, and volume modes allows for the ability to statically condense out the volume modes, significantly reducing the growth rate of globally coupled DOF as the order of the solution is increased in global matrix inversion stages. Sum factorization ideas have traditionally mainly been used for relatively high solution polynomial orders ($P \geq 8$) (reference for this, maybe sherwin) due to the lack of improvement over standard operators at lower orders, a minor goal of this work was

to demonstrate that the algorithm is also relevant in the polynomial range considered here ($0 \leq P \leq 4$). Refer to section confirming the break even point. Compare with sparse standard (collocated), discuss cost of application of inverse mass matrix (uncollocated), discuss cost of linear solve (implicit, uncollocated).

Another technique pursued for computational complexity minimization is the use of collocated schemes. Collocated schemes, notably the nodal DG scheme [10, 11], minimize interpolation and integration costs by collocating solution and flux interpolation nodes with cubature nodes. While this culminates in satisfactory results for linear PDEs solved on straight-sided elements, the use of higher-order accurate cubature rules is advocated in curved elements or when solving nonlinear systems of equations, destroying the collocation property of the scheme. It has also been demonstrated that simplex element “alpha-optimized” nodes [12] associated with the nodal DG scheme, were unsuitable when solving nonlinear PDEs due to significant aliasing errors [13]. This in turn spurred research into the derivation of triangular and tetrahedral element nodes which possess good interpolation properties and have an associated cubature rule, recovering at least volume collocation; this has been pursued by several authors [13, 14, 15]. However, cubature rules associated with these nodes are weak in the sense that they can not even exactly integrate linear PDEs on linear elements when cast in the variational weak DG framework which might degrade their accuracy and high-order convergence properties; the conclusions of Bassi et al. [16] suggest that this occurs for tensor-product elements.

A final avenue which has been pursued for the minimization of computational cost associated with high-order methods is the use of pre-integrated DG schemes [17]

or DG-type schemes cast in differential form, notably the Energy Stable Flux Reconstruction (ESFR) schemes introduced in §1.1.2. In these versions of the DG scheme, integral operations associated with the weak formulation of the DG scheme are performed in a preprocessing stage, eliminating many of the cubature operations associated with the original scheme. It is shown in [link to scheme derivation section](#) that the majority of operators can be pre-computed in this fashion for both weak and strong forms of DG-type schemes and that it would be naive not to do so.

1.1.2 Restrictive CFL Condition and Suboptimal Dissipation and Dispersion Properties

As there is some level of subcell resolution in high-order schemes, the maximum CFL number decreases with the order of approximation of the solution on a fixed mesh. This results in restrictive time step sizes when solving problems explicitly. The recently introduced Flux Reconstruction (FR) schemes, initially proposed for tensor-product elements [18] and subsequently generalized to simplex elements [19], were originally presented as a unifying framework for DG-type schemes where stable time step limits greatly in excess of the standard DG scheme were achieved. However, restrictions on the type of flux reconstruction allowable while maintaining linear stability of the various schemes proposed by Huynh [18] were initially unclear. Appropriate choices for energy stability resulted in the derivation of the ESFR schemes, energy stable for linear advection on tensor-product elements [20] and for advection-diffusion on tensor-product elements [21]; the extension to simplex elements has also been made [22, 23, 24].

Within the ESFR framework, with different schemes parametrized simply by two parameters c and k (which can be associated with a scaling of the highest mode of the numerical flux correction), investigations were performed to find the values of c and k achieving the maximum stable time steps for linear advection-diffusion problems. These values were denoted as c_+ and k_+ , allowing for time steps approximately twice as large as in the original DG formulation while maintaining the optimal order of convergence [21, 22, 23, 24]. A similar analysis relating to filtering of the numerical flux in the standard DG formulation also demonstrated that the stable time step could be increased while maintaining the optimal order of convergence [25]. More recent work has focused on optimizing within an ESFR framework to find optimal flux reconstruction operators for the minimization of dissipation and dispersion [26]. Interestingly, it was found that there existed an FR operator, not fitting within the ESFR framework, which had significantly improved performance as compared to the optimal ESFR operator for the minimization of dissipation and dispersion. While the authors were unable to conclude that this method was energy stable, the extension of the energy stability envelope may make this possible [27].

As the formulation of the ESFR schemes is extremely similar to that of the original DG scheme, it was natural that comparisons between the two would be made and it was eventually determined that ESFR schemes could be interpreted as modally filtered DG schemes. This was initially demonstrated in 1D and for straight-sided tensor product extensions [28] and subsequently generalized to straight-sided simplex elements [24]. A related investigation proving the equivalence of certain ESFR schemes with DG variants on both straight-sided and curved tensor-product elements

was also performed [29, 30]. More recently, equivalence between all ESFR schemes and modally filtered DG schemes on curved tensor-product and simplex elements was demonstrated, with the important conclusion that ESFR schemes can be formulated as a weak DG scheme where discontinuous edge flux is substituted for numerical edge flux correction [31]. Given the theoretical justifications for the numerical flux and the significant computational effort expended to compute it, this interpretation puts the modification of its contribution to the residual into question, despite increasing the stable time step size. It was further demonstrated that ESFR schemes are inherently less efficient than DG schemes in weak form when solving problems implicitly.

1.1.3 Treatment of Complex Geometry

The proper treatment of complex geometry in high-order DG-type methods has been shown to be crucial. In the seminal work on the topic, it was demonstrated that curved geometry had to be represented in an isoparametric space of order P for all solution polynomial orders $P \geq 2$ and in a superparametric space of second order in the case of first order solution representation in order to maintain the optimal convergence properties of these schemes, in two dimensions (2D) [32]. It was noted that low-order geometry representation results in deterioration of solution quality as the order of the scheme is increased, due to rarefaction waves being formed at vertices of polygonal mesh surfaces [33].

As the majority of mesh generators only provide linear meshes, a necessary inclusion in any high-order spectral element code is the projection of an initially straight-sided mesh to the curved geometry. This projection is commonly achieved through

the use of transfinite blending function interpolation, first proposed by Gordon et al. for tensor-product elements [34] and subsequently generalized to simplex elements [35, 36, 37, 38, 39, 40]. Assuming that corner nodes are located on the curved geometry, the process proceeds by sequentially projecting straight edge and facet nodes to the curved geometry followed by the application of a blending operation which appropriately displaces volume nodes (not touching the boundary). Ensuring that the error in the curved geometry representation converges at the same rate as the solution and the need for parametrization of arbitrary 3D curved surfaces represent two major challenges.

While errors due to improper geometry representation may only begin to manifest themselves at very fine levels of solution resolution, for high-order methods, these levels of accuracy are achievable. Further, if these geometric errors result in decreasing convergence rates of high-order methods, then the increase in the computational complexity would be incurred with no additional benefit, rendering these schemes useless. While optimal convergence orders for nonlinear hyperbolic cases have been demonstrated on curved geometry by many authors in 2D, the geometric complexity of the test cases used for these demonstrations has been minimal, generally with the possibility of employing analytic surface parametrization or using C^∞ boundary functions. Curved surface representation is most commonly achieved using nodal geometry interpolation through a set of surface geometry nodes obtained from an arc length surface parametrization. To the author's knowledge, optimal convergence orders have never been published for any 3D simulation of nonlinear hyperbolic systems of equations over curved geometry. In 3D, additional geometric concerns must also

be taken into account such as the enforcement of certain metric identities in order to ensure “free-stream preservation” (i.e. avoid violations of conservation resulting only from curved geometry treatment) [41].

It is important to note that the discussion above assumed that a valid linear mesh could initially be generated for the complex geometry to be modelled in the simulation. In fact, it has recently been estimated that approximately 80% of overall analysis time in the aerospace industry, among others, is devoted to (linear) mesh generation [42], resulting in serious challenges when attempting to interface between simulated aerodynamic shape optimized profiles and the Computer Aided Design (CAD) model, for example. This motivated the formulation of isogeometric analysis where the solution is represented using the same basis as the CAD geometry, allowing for perfect geometric representation at any level of mesh refinement and seamless interfacing with CAD modelling [42]. The competitiveness of this new approach with existing methods has already been demonstrated in numerous benchmark test cases in both structural and fluid mechanics. The Non-uniform rational basis spline (NURBS) enhanced finite element method (NEFEM) in which the standard DG method is employed in the majority of the solution domain while isogeometric analysis is employed in curved elements has also recently been shown to provide superior results to standard DG discretizations on all test cases considered [43, 44]. A further significant advantage of isogeometric analysis is that no interfacing with the CAD geometry would be required while performing mesh optimization during scheme execution, making these optimization algorithms suitable for the simulation of flow

over vastly more complex geometric objects than those generally used for academic purposes.

1.1.4 Optimal Test Spaces

The Galerkin method was originally developed for the solution of PDEs in structural mechanics in which it can be shown to provide optimal results based on the symmetry and positive-definiteness of the associated variational problem. This optimal approximation property is lost however in the context of hyperbolic PDEs. Thus, robustness issues associated with high-order methods generally result from the development of large oscillations in the presence of steep solution gradients (shocks or boundary layers) on under-resolved meshes. The intuitive remedy for this occurrence of locally refining the discontinuous space used to represent the solution is extremely effective, however, it is generally difficult to know a-priori where solution refinement is required and thus it is desirable to be able to robustly compute solutions on relatively coarse meshes in order to obtain a residual based indicator to guide the process; obtaining solutions on under-resolved meshes often requires some form of stabilization.

This issue was first addressed in the context of continuous finite element solutions through the use of residual-based stabilization in the form of the Streamline Upwind Petrov Galerkin (SUPG) formulation, in which the stabilization was interpreted as a variation of the test space [45]. It was subsequently shown that the SUPG formulation was in essence complementing the solution space with information related to the

fine-scale (unresolvable by the current mesh) Green's function in the context of the variational multiscale method [46, 47].

Following the motivation for pursuing test spaces other than the approximation spaces used in DG methods, the discontinuous Petrov-Galerkin (DPG) method with optimal test functions was introduced, where test functions are determined such that they minimize the residual in an appropriate energy setting [48, 49]; in other words, test spaces are chosen for good stability properties as opposed to good approximation properties. Further advantages of the method include the treatment of numerical traces and fluxes as the globally coupled unknowns, resulting in the static condensation discussed above, but also precluding the need for Riemann solvers. The use of optimal test spaces also results in an internally preconditioned algebraic least squares system [50], which is symmetric and positive definite, allowing for the use of efficient nonlinear iterative solvers. While the DPG method offers optimally stable spatial discretization with respect to the chosen residual norm without extra stabilization parameters, even in the presence of steep gradients, it has been observed that exact solutions under Newton linearization may contain large oscillations and that artificial viscosity might be needed not as a stabilization but as a regularization mechanism through which such oscillations can be suppressed [51]. Despite this concern, beginning from an extremely coarse solution, the DPG method has already been shown to have the ability to resolve physical features through automatic adaptation, without the use of artificial diffusion or shock capturing terms in the presence of large gradients [50].

Likely add to this section after further exploring DPG: Comparison of upwinded test space with adjoint for goal-oriented adaptive refinement; investigation into comparison of dissipation of Riemann flux vs DPG.

1.2 Thesis Overview

ToBeModified

Implementation of sum factorization for faster algorithms. Note the lack of improvement when simply using sparse operators for the collocated scheme but likely show results for DPG where the scheme is inherently uncollocated and the standard operator is fully dense. Make some nice figures to demonstrate. Also show convergence order results using collocated nodes for simplex elements with suboptimal orders.

Investigation of ESFR schemes (comparison with other methods). Possibly some dissipation/dispersion results for DG vs DPG.

Potential comparison of blending vs ToBeCurved. Investigation into proper treatment of metric terms and cubature order (provide supporting mathematical analysis if possible). Comparison with NURBS geometry (time permitting).

List DPG related work when completed.

1.3 Accompanying Code

All of the results presented below have been prepared using an open-source c code available on [github](#). Throughout the thesis, specific functions are referred to with the hope of encouraging the reader to consult the code if further implementation

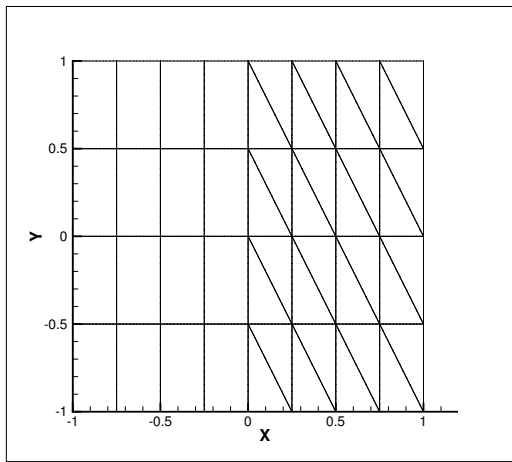
details are of interest; in these cases, the function names are highlighted in blue as in

main.c

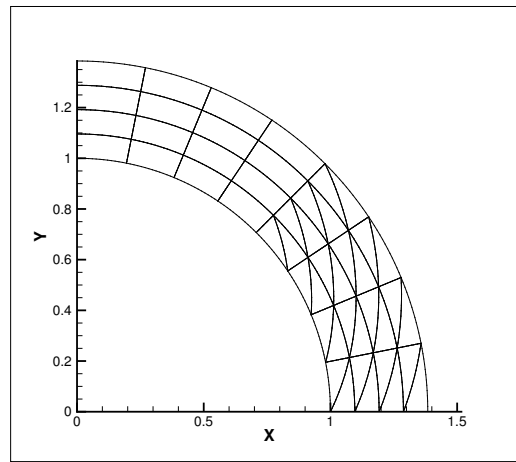
1.4 Testing

Table 1–1: Errors and Convergence Orders - $PI_c = 2P$ (GL/Cools), $PI_{cf} = 2P$ (GL), $PG = P$, EFE = 1

Polynomial	Mesh Size	L_2 Error			Conv. Order		
		ρ	p	s	ρ	p	s
P1	7.91e-02	1.66e-01	2.42e-01	4.52e-02	-	-	-
	3.95e-02	4.93e-02	6.58e-02	1.58e-02	1.75	1.88	1.52
	1.98e-02	1.63e-02	2.08e-02	5.63e-03	1.60	1.66	1.49
	9.88e-03	5.56e-03	6.90e-03	2.00e-03	1.55	1.59	1.49
P2	1.09e-01	1.75e-03	1.78e-03	1.55e-03	-	-	-
	5.46e-02	2.20e-04	2.06e-04	2.38e-04	2.99	3.11	2.70
	2.73e-02	2.75e-05	2.47e-05	3.22e-05	3.00	3.06	2.89
	1.36e-02	3.43e-06	2.99e-06	4.15e-06	3.01	3.05	2.96
	6.82e-03	4.29e-07	3.69e-07	5.25e-07	3.00	3.02	2.98
P3	8.33e-02	6.68e-04	8.98e-04	6.25e-05	-	-	-
	4.17e-02	7.09e-05	9.54e-05	4.82e-06	3.24	3.24	3.70
	2.08e-02	7.25e-06	9.89e-06	4.05e-07	3.29	3.27	3.57
	1.04e-02	7.64e-07	1.06e-06	2.76e-08	3.24	3.22	3.87
	5.21e-03	7.23e-08	1.03e-07	1.76e-09	3.40	3.36	3.97
P4	6.74e-02	6.33e-06	7.24e-06	4.53e-06	-	-	-
	3.37e-02	2.58e-07	2.68e-07	2.39e-07	4.62	4.76	4.24
	1.69e-02	8.27e-09	7.60e-09	8.92e-09	4.96	5.14	4.74
	8.43e-03	2.56e-10	2.22e-10	2.91e-10	5.01	5.10	4.94



(a) Mesh before Mapping



(b) Mapped Mesh

Figure 1-1: Supersonic Vortex Structured Mixed Mesh

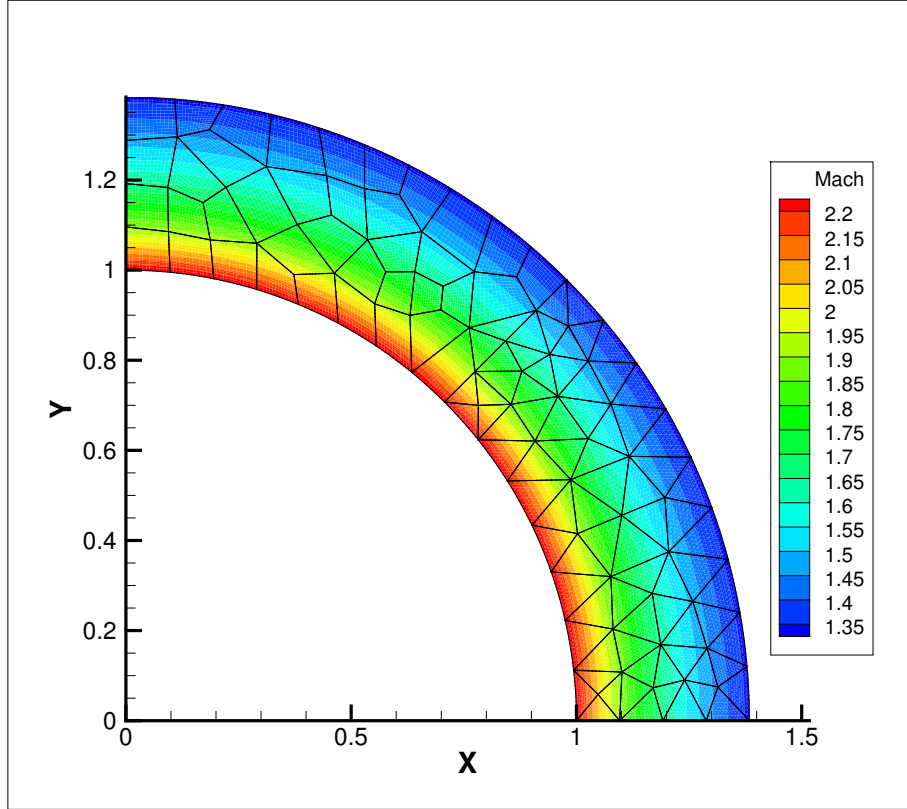


Figure 1-2: Mach Number Distribution: $P = 4$, $PG = 4$, $PF = 5$, $PFr = 8$, $PIs = 8$, $PIc = 11$, $PIsf = 8$, $PIcf = 12$, $EFE = 1$

CHAPTER 2

Governing Equations

2.1 Thermodynamics

2.2 Navier-Stokes Equations

2.3 Boundary Conditions

2.4 Non-Dimensionalization

Under Consideration

CHAPTER 3

Numerical Methods

don't forget to add notation

The fundamental underlying numerical methods used for this work are presented below using a general first order PDE as a model. Emphasis is placed on the discrete representations for ease of transition between the mathematical and computational frameworks.

Add appendix on basis functions and derivatives.

3.1 Discontinuous Petrov-Galerkin Method

Consider the scalar 3D homogeneous conservation law,

$$\frac{\partial}{\partial t}u(\mathbf{x},t) + \nabla \cdot \mathbf{f}(u(\mathbf{x},t)) = 0, \quad t \geq 0, \quad \mathbf{x} := \begin{bmatrix} x & y & z \end{bmatrix} \in \Omega,$$
$$u(\mathbf{x},0) = u_0(\mathbf{x}), \mathbf{x} \in \Omega,$$

where $\mathbf{f}(u(\mathbf{x},t)) := \begin{bmatrix} f_1(u(\mathbf{x},t)) & f_2(u(\mathbf{x},t)) & f_3(u(\mathbf{x},t)) \end{bmatrix}$ represents the fluxes in the x , y and z directions. Note that row vector notation is used throughout. The finite dimensional computational space, denoted by Ω^h , is divided into M elements where

$$\Omega \simeq \Omega^h := \bigcup_{m=1}^M \Omega_m^h.$$

The global solution, $u(\mathbf{x}, t)$, is assumed to be represented by $u^h(\mathbf{x}, t)$, which is composed of piecewise polynomial approximations,

$$u(\mathbf{x}, t) \simeq u^h(\mathbf{x}, t) = \bigoplus_{m=1}^M u_m^h(\mathbf{x}, t).$$

On each element, the solution is represented by a combination of N_{vn_S} linearly independent modal (hierachical) or nodal (interpolating) polynomial basis functions, $\phi_{\mathbf{m}}(\mathbf{x})$, of maximal order P , and is generally expressed as

$$\mathbf{x} \in \Omega_m : u_m^h(\mathbf{x}, t) = \sum_{i=1}^{N_{vn_S}} \phi_{m,i}(\mathbf{x}) \hat{u}_{m,i}(t) := \phi_{\mathbf{m}}(\mathbf{x}) \hat{\mathbf{u}}_{\mathbf{m}}(t)^T.$$

For tensor-product (TP), simplex (SI) and pyramidal (PYR) formulations, $N_{vn_S} = (P+1)^d$, $\frac{(P+d)!}{d!P!}$ and $\frac{(P+2)!(2P+3)}{d!P!}$, respectively, where d indicates the problem dimension; wedge (WEDGE) formulations employ a combination of 2D SI and 1D TP basis functions. The elementwise residuals are given by

$$R_m^h(\mathbf{x}, t) = \frac{\partial}{\partial t} u_m^h(\mathbf{x}, t) + \nabla \cdot \mathbf{f}(u_m^h(\mathbf{x}, t)) \quad (3.1.1)$$

for all M elements. The residuals are required to be orthogonal to a set of polynomial test functions, $\psi_{\mathbf{m}}(\mathbf{x})$, which need not be the same as the basis functions. After integration by parts, the integral weak form of the DPG scheme is obtained,

$$\begin{aligned} \int_{\Omega_m} \boldsymbol{\psi}_m(\boldsymbol{x})^T \frac{\partial}{\partial t} u_m^h(\boldsymbol{x}, t) - \nabla \boldsymbol{\psi}_m(\boldsymbol{x})^T \cdot \boldsymbol{f}(u_m^h(\boldsymbol{x}, t)) d\Omega_m \\ + \int_{\Gamma_m} \boldsymbol{\psi}_m(\boldsymbol{x})^T \hat{\boldsymbol{n}}_m \cdot \boldsymbol{f}^*(u^h(\boldsymbol{x}, t)) d\Gamma_m = \mathbf{0}^T, \end{aligned}$$

where Γ_m represents the element surface, $\boldsymbol{f}^*(u^h(\boldsymbol{x}, t))$ denotes the numerical flux,

$$\boldsymbol{\psi}_m(\boldsymbol{x}) := \begin{bmatrix} \psi_{m,1}(\boldsymbol{x}) & \psi_{m,2}(\boldsymbol{x}) & \dots & \psi_{m,N_{vn_T}}(\boldsymbol{x}) \end{bmatrix}$$

holds the test functions for the element and

$$\hat{\boldsymbol{n}}_m = \begin{bmatrix} n^x & n^y & n^z \end{bmatrix}_m.$$

The transformation mapping the physical coordinates to the reference coordinates on each element is expressed as

3.2 Adjoint Formulation

CHAPTER 4

Computational Complexity Improvements

4.1 Sum Factorization

As discussed in §1.1.1, sum factorization asymptotically reduces computational complexity of tensor-product operator application from $O(N^{2d})$ to $O(N^{d+1})$ where d is the dimension of the problem and $N = P+1$ where P is the order of the method. This is achieved by exploiting the decomposition of multi-dimensional TP operators into their 1D components, whenever both the basis functions and interpolation/cubature nodes admit such a decomposition.

4.1.1 Representative Example

The algorithm is most generally illustrated with a 3D example. Consider the case of numerical integration of the first component of the flux divergence with respect to the DG test functions as in [refer to equation](#),

$$\phi_r(\mathbf{R}_{v_I})^T \mathbf{W}_{v_I} \phi(\mathbf{R}_{v_I}) \hat{\mathbf{f}}_1^T := \mathbf{D}_r^w \hat{\mathbf{f}}_1^T.$$

The possibility of separation of the \mathbf{D}_r^w operator into its 1D components is most easily seen by expanding the matrix operations into summations. Using indices of b , t and c to represent basis functions, test functions, and cubature nodes respectively, and visualizing $\hat{\mathbf{f}}_1$ as a 3D array with $b_1 \times b_2 \times b_3$ components,

$$\begin{aligned}
& \phi_{t_1, t_2, t_{3r}}(\mathbf{R}_{v_I}) \mathbf{W}_{v_I} \phi(\mathbf{R}_{v_I}) \hat{\mathbf{f}}_1^T \\
&= \sum_{c_1} \sum_{c_2} \sum_{c_3} \phi_{t_1, t_2, t_{3r}}(r_{c_1}, s_{c_2}, t_{c_3}) w_{c_1} w_{c_2} w_{c_3} \phi(r_{c_1}, s_{c_2}, t_{c_3}) \hat{\mathbf{f}}_1^T \\
&= \sum_{c_1} \sum_{c_2} \sum_{c_3} \phi_{t_{1r}}^{1D}(r_{c_1}) \phi_{t_2}^{1D}(r_{c_2}) \phi_{t_3}^{1D}(r_{c_3}) w_{c_1} w_{c_2} w_{c_3} \sum_{b_1} \sum_{b_2} \sum_{b_3} \phi_{b_1}^{1D}(r_{c_1}) \phi_{b_2}^{1D}(r_{c_2}) \phi_{b_3}^{1D}(r_{c_3}) \hat{f}_{1_{b_1, b_2, b_3}} \\
&= \sum_{c_3} \sum_{b_3} \phi_{t_3}^{1D}(r_{c_3}) w_{c_3} \phi_{b_3}^{1D}(r_{c_3}) \sum_{c_2} \sum_{b_2} \phi_{t_2}^{1D}(r_{c_2}) w_{c_2} \phi_{b_2}^{1D}(r_{c_2}) \sum_{c_1} \sum_{b_1} \phi_{t_{1r}}^{1D}(r_{c_1}) w_{c_1} \phi_{b_1}^{1D}(r_{c_1}) \hat{f}_{1_{b_1, b_2, b_3}},
\end{aligned} \tag{4.1.1}$$

where summations are over the full range of each index. Assuming that the flux in reference space and test functions both use a P_1 basis, $\phi_r(\mathbf{R})$, and using P_2 cubature nodes, \mathbf{R}_{v_I} , the 1D interpolation and differentiation operators can be defined as

$$\begin{aligned}
\mathbf{I}^{w, 1D} &:= \phi^{1D}(\mathbf{r}_{v_I})^T \mathbf{W}_{v_I}^{1D} \phi^{1D}(\mathbf{r}_{v_I}) \\
&= \begin{bmatrix} \phi_0^{1D}(r_{v_I,0}) & \phi_0^{1D}(r_{v_I,1}) & \phi_0^{1D}(r_{v_I,2}) \\ \phi_1^{1D}(r_{v_I,0}) & \phi_1^{1D}(r_{v_I,1}) & \phi_1^{1D}(r_{v_I,2}) \end{bmatrix} \begin{bmatrix} w_0^{1D} & 0 & 0 \\ 0 & w_1^{1D} & 0 \\ 0 & 0 & w_2^{1D} \end{bmatrix} \begin{bmatrix} \phi_0^{1D}(r_{v_I,0}) & \phi_1^{1D}(r_{v_I,0}) \\ \phi_0^{1D}(r_{v_I,1}) & \phi_1^{1D}(r_{v_I,1}) \\ \phi_0^{1D}(r_{v_I,2}) & \phi_1^{1D}(r_{v_I,2}) \end{bmatrix}
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{D}^{w, 1D} &:= \phi_r^{1D}(\mathbf{r}_{v_I})^T \mathbf{W}_{v_I}^{1D} \phi^{1D}(\mathbf{r}_{v_I}) \\
&= \begin{bmatrix} \phi_{r,0}^{1D}(r_{v_I,0}) & \phi_{r,0}^{1D}(r_{v_I,1}) & \phi_{r,0}^{1D}(r_{v_I,2}) \\ \phi_{r,1}^{1D}(r_{v_I,0}) & \phi_{r,1}^{1D}(r_{v_I,1}) & \phi_{r,1}^{1D}(r_{v_I,2}) \end{bmatrix} \begin{bmatrix} w_0^{1D} & 0 & 0 \\ 0 & w_1^{1D} & 0 \\ 0 & 0 & w_2^{1D} \end{bmatrix} \begin{bmatrix} \phi_0^{1D}(r_{v_I,0}) & \phi_1^{1D}(r_{v_I,0}) \\ \phi_0^{1D}(r_{v_I,1}) & \phi_1^{1D}(r_{v_I,1}) \\ \phi_0^{1D}(r_{v_I,2}) & \phi_1^{1D}(r_{v_I,2}) \end{bmatrix}.
\end{aligned}$$

The standard operation can then be decomposed as

$$\begin{aligned}
\mathbf{D}_r^w \hat{\mathbf{f}}_1^T := & \begin{bmatrix} i_{00} & 0 & 0 & 0 & i_{01} & 0 & 0 & 0 \\ 0 & i_{00} & 0 & 0 & 0 & i_{01} & 0 & 0 \\ 0 & 0 & i_{00} & 0 & 0 & 0 & i_{01} & 0 \\ 0 & 0 & 0 & i_{00} & 0 & 0 & 0 & i_{01} \\ i_{10} & 0 & 0 & 0 & i_{11} & 0 & 0 & 0 \\ 0 & i_{10} & 0 & 0 & 0 & i_{11} & 0 & 0 \\ 0 & 0 & i_{10} & 0 & 0 & 0 & i_{11} & 0 \\ 0 & 0 & 0 & i_{10} & 0 & 0 & 0 & i_{11} \end{bmatrix} \begin{bmatrix} i_{00} & 0 & i_{01} & 0 & 0 & 0 & 0 & 0 \\ 0 & i_{00} & 0 & i_{01} & 0 & 0 & 0 & 0 \\ i_{10} & 0 & i_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & i_{10} & 0 & i_{11} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & i_{00} & 0 & i_{01} & 0 \\ 0 & 0 & 0 & 0 & 0 & i_{00} & 0 & i_{01} \\ 0 & 0 & 0 & 0 & i_{10} & 0 & i_{11} & 0 \\ 0 & 0 & 0 & 0 & 0 & i_{10} & 0 & i_{11} \end{bmatrix} \\
& \begin{bmatrix} d_{00} & d_{01} & 0 & 0 & 0 & 0 & 0 & 0 \\ d_{10} & d_{11} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & d_{00} & d_{01} & 0 & 0 & 0 & 0 \\ 0 & 0 & d_{10} & d_{11} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{00} & d_{01} & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{10} & d_{11} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & d_{00} & d_{01} \\ 0 & 0 & 0 & 0 & 0 & 0 & d_{10} & d_{11} \end{bmatrix} \begin{bmatrix} \hat{f}_{1,000} \\ \hat{f}_{1,100} \\ \hat{f}_{1,010} \\ \hat{f}_{1,110} \\ \hat{f}_{1,001} \\ \hat{f}_{1,101} \\ \hat{f}_{1,011} \\ \hat{f}_{1,111} \end{bmatrix}.
\end{aligned}$$

Noting that the non-zero elements in the decomposed components of \mathbf{D}_r^w are built from repetition of the 1D operators and assuming that $\hat{\mathbf{f}}_1$ is stored continuously in memory based on sequential variation of indices r , s , then t , the application of the standard operator can be performed in the following steps:

1. Multiplication in the r direction by $\mathbf{D}^{w,1D}$, interpreting $\hat{\hat{\mathbf{f}}}_1$ as a $N \times N^2$ column-major matrix,

$$\hat{\hat{\mathbf{f}}}_1^{Int_1} = \begin{bmatrix} d_{00} & d_{01} \\ d_{10} & d_{11} \end{bmatrix} \begin{bmatrix} \hat{f}_{1,000} & \hat{f}_{1,010} & \hat{f}_{1,001} & \hat{f}_{1,011} \\ \hat{f}_{1,100} & \hat{f}_{1,110} & \hat{f}_{1,101} & \hat{f}_{1,111} \end{bmatrix}$$

2. Transposition of $\hat{\hat{\mathbf{f}}}_1$, interpreted as a $N \times N^2$ column-major matrix, followed by multiplication by $\mathbf{I}^{w,1D}$ (in the s direction), followed by a transpose interpreting the result as a $N^2 \times N$ matrix,

$$\begin{aligned} \hat{\hat{\mathbf{f}}}_1^{Int_1} &= \begin{bmatrix} \hat{f}_{1,000}^{Int_1} & \hat{f}_{1,010}^{Int_1} & \hat{f}_{1,001}^{Int_1} & \hat{f}_{1,011}^{Int_1} \\ \hat{f}_{1,100}^{Int_1} & \hat{f}_{1,110}^{Int_1} & \hat{f}_{1,101}^{Int_1} & \hat{f}_{1,111}^{Int_1} \end{bmatrix} \rightarrow \begin{bmatrix} \hat{f}_{1,000}^{Int_1} & \hat{f}_{1,001}^{Int_1} & \hat{f}_{1,100}^{Int_1} & \hat{f}_{1,101}^{Int_1} \\ \hat{f}_{1,010}^{Int_1} & \hat{f}_{1,011}^{Int_1} & \hat{f}_{1,110}^{Int_1} & \hat{f}_{1,111}^{Int_1} \end{bmatrix} \\ \hat{\hat{\mathbf{f}}}_1^{Int_2} &= \begin{bmatrix} i_{00} & i_{01} \\ i_{10} & i_{11} \end{bmatrix} \begin{bmatrix} \hat{f}_{1,000}^{Int_1} & \hat{f}_{1,001}^{Int_1} & \hat{f}_{1,100}^{Int_1} & \hat{f}_{1,101}^{Int_1} \\ \hat{f}_{1,010}^{Int_1} & \hat{f}_{1,011}^{Int_1} & \hat{f}_{1,110}^{Int_1} & \hat{f}_{1,111}^{Int_1} \end{bmatrix} \\ \hat{\hat{\mathbf{f}}}_1^{Int_2} &= \begin{bmatrix} \hat{f}_{1,000}^{Int_2} & \hat{f}_{1,001}^{Int_2} & \hat{f}_{1,100}^{Int_2} & \hat{f}_{1,101}^{Int_2} \\ \hat{f}_{1,010}^{Int_2} & \hat{f}_{1,011}^{Int_2} & \hat{f}_{1,110}^{Int_2} & \hat{f}_{1,111}^{Int_2} \end{bmatrix} \rightarrow \begin{bmatrix} \hat{f}_{1,000}^{Int_2} & \hat{f}_{1,010}^{Int_2} & \hat{f}_{1,001}^{Int_2} & \hat{f}_{1,011}^{Int_2} \\ \hat{f}_{1,100}^{Int_2} & \hat{f}_{1,110}^{Int_2} & \hat{f}_{1,101}^{Int_2} & \hat{f}_{1,111}^{Int_2} \end{bmatrix} \end{aligned}$$

3. Transposition of $\hat{\hat{\mathbf{f}}}_1^{Int_2}$, interpreted as a $N^2 \times N$ column-major matrix, followed by multiplication by $\mathbf{I}^{w,1D}$ (now in the t direction), followed by a transpose interpreting the result as a $N \times N^2$ matrix,

$$\hat{\hat{\mathbf{f}}}_1^{Int_2} = \begin{bmatrix} \hat{f}_{1,000}^{Int_2} & \hat{f}_{1,010}^{Int_2} & \hat{f}_{1,001}^{Int_2} & \hat{f}_{1,011}^{Int_2} \\ \hat{f}_{1,100}^{Int_2} & \hat{f}_{1,110}^{Int_2} & \hat{f}_{1,101}^{Int_2} & \hat{f}_{1,111}^{Int_2} \end{bmatrix} \rightarrow \begin{bmatrix} \hat{f}_{1,000}^{Int_2} & \hat{f}_{1,100}^{Int_2} & \hat{f}_{1,010}^{Int_2} & \hat{f}_{1,110}^{Int_2} \\ \hat{f}_{1,001}^{Int_2} & \hat{f}_{1,101}^{Int_2} & \hat{f}_{1,011}^{Int_2} & \hat{f}_{1,111}^{Int_2} \end{bmatrix}$$

$$\hat{\mathbf{f}}_1^{fin} = \begin{bmatrix} i_{00} & i_{01} \\ i_{10} & i_{11} \end{bmatrix} \begin{bmatrix} \hat{f}_{1,000}^{Int_2} & \hat{f}_{1,100}^{Int_2} & \hat{f}_{1,010}^{Int_2} & \hat{f}_{1,110}^{Int_2} \\ \hat{f}_{1,001}^{Int_2} & \hat{f}_{1,101}^{Int_2} & \hat{f}_{1,011}^{Int_2} & \hat{f}_{1,111}^{Int_2} \end{bmatrix}$$

$$\hat{\mathbf{f}}_1^{fin} = \begin{bmatrix} \hat{f}_{1,000}^{fin} & \hat{f}_{1,100}^{fin} & \hat{f}_{1,010}^{fin} & \hat{f}_{1,110}^{fin} \\ \hat{f}_{1,001}^{fin} & \hat{f}_{1,101}^{fin} & \hat{f}_{1,011}^{fin} & \hat{f}_{1,111}^{fin} \end{bmatrix} \rightarrow \begin{bmatrix} \hat{f}_{1,000}^{fin} & \hat{f}_{1,010}^{fin} & \hat{f}_{1,001}^{fin} & \hat{f}_{1,011}^{fin} \\ \hat{f}_{1,100}^{fin} & \hat{f}_{1,110}^{fin} & \hat{f}_{1,101}^{fin} & \hat{f}_{1,111}^{fin} \end{bmatrix}.$$

Note that the redundant double transpose between applications of s and t operators can be combined into one but was not done in the interest of clarity. In the 2D TP case, the steps relating to the t direction are skipped and in the 3D case using WEDGE elements, the TRI operator is interpreted as the r operator, with no operator in the s direction, followed by the 1D TP operation in the t direction. Counting operations using general dimensions of N in each coordinate direction, it is clear for the application of the fully dense standard operator that the operation count is

$$N^3 \times (2N^3 - 1) = O(N^6) = O(N^{2d}), \quad (4.1.2)$$

while for the three 1D TP operators, omitting the cost of the transposes, the operation count is

$$3 \times N \times (2N - 1) \times N^2 = O(N^4) = O(N^{d+1}). \quad (4.1.3)$$

As this result is asymptotic, the advantages of the sum factorized approach are not apparent at low orders. Furthermore, for collocated schemes, TP interpolation operators are identity matrices and the standard differentiation operator becomes very

sparse. Using compressed sparse row (CSR) storage, application of standard operators remains competitive with sum factorized operators for all orders tested in this case. §4.1.2 presents results for timing comparisons using the different formulations.

don't forget to discuss implicit operators

4.1.2 Timing Comparisons

Add results

4.2 Representation of Operators in Fourier Space

CHAPTER 5

TESTING FOOTNOTES

We will now work on testing footnotes.¹ And now let's add some spurious text so that we can have a second² footnote.

¹ This is the first footnote.

² This is the second footnote. To test if the footnote is typeset single-spaced, we add meaningless text until we can see a second line.

Appendix A

Here is the text of an Appendix.

Appendix B

Here is the text of a second, additional Appendix

REFERENCES

- [1] W. Reed, T. Hill, Triangular mesh methods for the neutron transport equation, 1973.
URL <http://www.osti.gov/scitech/servlets/purl/4491151>
- [2] B. Cockburn, C.-W. Shu, The Runge-Kutta local projection p1-discontinuous Galerkin finite element method for scalar conservation laws, *RAIRO Modél. Math. Anal. Numér* 25 (3) (1991) 337–361.
- [3] B. Cockburn, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. ii. general framework, *Mathematics of Computation* 52 (186) (1989) 411–435.
- [4] B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws iii: one-dimensional systems, *Journal of Computational Physics* 84 (1) (1989) 90–113.
- [5] B. Cockburn, S. Hou, C.-W. Shu, The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. iv. the multidimensional case, *Mathematics of Computation* 54 (190) (1990) 545–581.
- [6] B. Cockburn, C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws v: multidimensional systems, *Journal of Computational Physics* 141 (2) (1998) 199–224.
- [7] S. A. Orszag, Spectral methods for problems in complex geometries, *Journal of Computational Physics* 37 (1) (1980) 70 – 92.
doi:[http://dx.doi.org/10.1016/0021-9991\(80\)90005-4](http://dx.doi.org/10.1016/0021-9991(80)90005-4).
URL <http://www.sciencedirect.com/science/article/pii/0021999180900054>
- [8] D. Kopriva, Spectral element methods, in: *Implementing Spectral Methods for Partial Differential Equations*, Scientific Computation, Springer Netherlands, 2009, pp. 293–354. doi:[10.1007/978-90-481-2261-5_8](https://doi.org/10.1007/978-90-481-2261-5_8).
URL http://dx.doi.org/10.1007/978-90-481-2261-5_8

- [9] G. Karniadakis, S. Sherwin, Spectral/hp Element Methods for CFD, Numerical mathematics and scientific computation, Oxford University Press, 1999.
URL https://books.google.ca/books?id=R_ydFKtI59cC
- [10] J. Hesthaven, T. Warburton, Nodal high-order methods on unstructured grids: I. time-domain solution of Maxwell's equations, Journal of Computational Physics 181 (1) (2002) 186 – 221. doi:<http://dx.doi.org/10.1006/jcph.2002.7118>.
URL <http://www.sciencedirect.com/science/article/pii/S0021999102971184>
- [11] J. S. Hesthaven, T. Warburton, Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications, 1st Edition, Springer Publishing Company, Incorporated, 2007.
URL <http://www.springer.com/mathematics/computational+science+%26+engineering/book/978-0-387-72065-4>
- [12] T. Warburton, An explicit construction of interpolation nodes on the simplex, Journal of Engineering Mathematics 56 (3) (2006) 247–262. doi:[10.1007/s10665-006-9086-6](http://dx.doi.org/10.1007/s10665-006-9086-6).
URL <http://dx.doi.org/10.1007/s10665-006-9086-6>
- [13] F. Witherden, P. Vincent, An analysis of solution point coordinates for flux reconstruction schemes on triangular elements, Journal of Scientific Computing 61 (2) (2014) 398–423. doi:[10.1007/s10915-014-9832-2](http://dx.doi.org/10.1007/s10915-014-9832-2).
URL <http://dx.doi.org/10.1007/s10915-014-9832-2>
- [14] D. Williams, L. Shunn, A. Jameson, Symmetric quadrature rules for simplexes based on sphere close packed lattice arrangements, Journal of Computational and Applied Mathematics 266 (0) (2014) 18–38. doi:<http://dx.doi.org/10.1016/j.cam.2014.01.007>.
URL <http://www.sciencedirect.com/science/article/pii/S0377042714000211>
- [15] L. Shunn, F. Ham, Symmetric quadrature rules for tetrahedra based on a cubic close-packed lattice arrangement, Journal of Computational and Applied Mathematics 236 (17) (2012) 4348–4364. doi:<http://dx.doi.org/10.1016/j.cam.2012.03.032>.
URL <http://www.sciencedirect.com/science/article/pii/S0377042712001604>

- [16] F. Bassi, N. Franchina, A. Ghidoni, S. Rebay, A numerical investigation of a spectral-type nodal collocation discontinuous Galerkin approximation of the Euler and Navier-Stokes equations, *International Journal for Numerical Methods in Fluids* 71 (10) (2013) 1322–1339. doi:10.1002/flid.3713.
URL <http://dx.doi.org/10.1002/flid.3713>
- [17] H. L. Atkins, W. Chi-Shu, Quadrature-free implementation of discontinuous Galerkin method for hyperbolic equations, Tech. rep. (1996).
- [18] H. T. Huynh, A Flux Reconstruction Approach to High-Order Schemes Including Discontinuous Galerkin Methods, American Institute of Aeronautics and Astronautics, 2007. doi:10.2514/6.2007-4079.
URL <http://dx.doi.org/10.2514/6.2007-4079>
- [19] Z. Wang, H. Gao, A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids, *Journal of Computational Physics* 228 (21) (2009) 8161 – 8186. doi:<http://dx.doi.org/10.1016/j.jcp.2009.07.036>.
URL <http://www.sciencedirect.com/science/article/pii/S0021999109004239>
- [20] P. Vincent, P. Castonguay, A. Jameson, A new class of high-order energy stable flux reconstruction schemes, *Journal of Scientific Computing* 47 (1) (2011) 50–72. doi:10.1007/s10915-010-9420-z.
URL <http://dx.doi.org/10.1007/s10915-010-9420-z>
- [21] P. Castonguay, D. Williams, P. Vincent, A. Jameson, Energy stable flux reconstruction schemes for advection-diffusion problems, *Computer Methods in Applied Mechanics and Engineering* (2013) 400–417. doi:10.1016/j.cma.2013.08.012.
URL <http://www.sciencedirect.com/science/article/pii/S0045782513002156>
- [22] P. Castonguay, P. Vincent, A. Jameson, A new class of high-order energy stable flux reconstruction schemes for triangular elements, *Journal of Scientific Computing* 51 (1) (2012) 224–256. doi:10.1007/s10915-011-9505-3.
URL <http://dx.doi.org/10.1007/s10915-011-9505-3>
- [23] D. Williams, P. Castonguay, P. Vincent, A. Jameson, Energy stable flux reconstruction schemes for advection-diffusion problems on

- triangles, *Journal of Computational Physics* 250 (2013) 53 – 76. doi:<http://dx.doi.org/10.1016/j.jcp.2013.05.007>.
 URL <http://www.sciencedirect.com/science/article/pii/S0021999113003318>
- [24] D. Williams, A. Jameson, Energy stable flux reconstruction schemes for advection-diffusion problems on tetrahedra, *Journal of Scientific Computing* 59 (3) (2014) 721–759. doi:[10.1007/s10915-013-9780-2](https://doi.org/10.1007/s10915-013-9780-2).
 URL <http://dx.doi.org/10.1007/s10915-013-9780-2>
- [25] N. Chalmers, L. Krivodonova, R. Qin, Relaxing the CFL number of the discontinuous Galerkin method, *SIAM Journal on Scientific Computing* 36 (4) (2014) A2047–A2075. arXiv:<http://dx.doi.org/10.1137/130927504>, doi:[10.1137/130927504](https://doi.org/10.1137/130927504).
 URL <http://dx.doi.org/10.1137/130927504>
- [26] K. Asthana, A. Jameson, High-order flux reconstruction schemes with minimal dispersion and dissipation, *Journal of Scientific Computing* (2014) 1–32. doi:[10.1007/s10915-014-9882-5](https://doi.org/10.1007/s10915-014-9882-5).
 URL <http://dx.doi.org/10.1007/s10915-014-9882-5>
- [27] P. Vincent, A. Farrington, F. Witherden, A. Jameson, An extended range of stable-symmetric-conservative flux reconstruction correction functions, *Computer Methods in Applied Mechanics and Engineering* 296 (2015) 248 – 272. doi:<http://dx.doi.org/10.1016/j.cma.2015.07.023>.
 URL <http://www.sciencedirect.com/science/article/pii/S0045782515002418>
- [28] Y. Allaneau, A. Jameson, Connections between the filtered discontinuous Galerkin method and the flux reconstruction approach to high order discretizations, *Computer Methods in Applied Mechanics and Engineering* 200 (49–52) (2011) 3628–3636. doi:<http://dx.doi.org/10.1016/j.cma.2011.08.019>.
 URL <http://www.sciencedirect.com/science/article/pii/S004578251100274X>
- [29] D. De Grazia, G. Mengaldo, D. Moxey, P. E. Vincent, S. J. Sherwin, Connections between the discontinuous Galerkin method and high-order flux reconstruction schemes, *International Journal for Numerical Methods in Fluids* 75 (12) (2014) 860–877. doi:[10.1002/flid.3915](https://doi.org/10.1002/flid.3915).
 URL <http://dx.doi.org/10.1002/flid.3915>

- [30] G. Mengaldo, D. De Grazia, P. Vincent, S. Sherwin, On the connections between discontinuous Galerkin and flux reconstruction schemes: Extension to curvilinear meshes, *Journal of Scientific Computing* (2015) 1–21 doi:10.1007/s10915-015-0119-z.
URL <http://dx.doi.org/10.1007/s10915-015-0119-z>
- [31] P. Zwanenburg, S. Nadarajah, Equivalence between the energy stable flux reconstruction and filtered discontinuous Galerkin schemes, *Journal of Computational Physics* 306 (2016) 343 – 369. doi:http://dx.doi.org/10.1016/j.jcp.2015.11.036.
URL <http://www.sciencedirect.com/science/article/pii/S0021999115007767>
- [32] F. Bassi, S. Rebay, High-order accurate discontinuous finite element solution of the 2D Euler equations, *Journal of Computational Physics* 138 (2) (1997) 251 – 285. doi:http://dx.doi.org/10.1006/jcph.1997.5454.
URL <http://www.sciencedirect.com/science/article/pii/S0021999197954541>
- [33] L. Krivodonova, M. Berger, High-order accurate implementation of solid wall boundary conditions in curved geometries, *J. Comput. Phys.* 211 (2) (2006) 492–512. doi:10.1016/j.jcp.2005.05.029.
URL <http://dx.doi.org/10.1016/j.jcp.2005.05.029>
- [34] W. Gordon, C. Hall, Transfinite element methods: Blending-function interpolation over arbitrary curved element domains, *Numerische Mathematik* 21 (2) (1973) 109–129. doi:10.1007/BF01436298.
URL <http://dx.doi.org/10.1007/BF01436298>
- [35] G. M. Nielson, The side-vertex method for interpolation in triangles, *Journal of Approximation Theory* 25 (4) (1979) 318 – 336. doi:http://dx.doi.org/10.1016/0021-9045(79)90020-0.
URL <http://www.sciencedirect.com/science/article/pii/0021904579900200>
- [36] R. Haber, M. S. Shephard, J. F. Abel, R. H. Gallagher, D. P. Greenberg, A general two-dimensional, graphical finite element preprocessor utilizing discrete transfinite mappings, *International Journal for Numerical Methods in Engineering* 17 (7) (1981) 1015–1044. doi:10.1002/nme.1620170706.
URL <http://dx.doi.org/10.1002/nme.1620170706>

- [37] B. Szabo, I. Babuška, Finite Element Analysis, A Wiley-Interscience publication, Wiley, 1991.
URL <https://books.google.ca/books?id=JsCg-QWUT28C>
- [38] C. Lacombe, C. Bedard, Face-apex projectors for the interpolation function of a general tetrahedral mid-edge finite element, *Comput. Methods Appl. Mech. Eng.* 68 (2) (1988) 177–188. doi:10.1016/0045-7825(88)90114-4.
URL [http://dx.doi.org/10.1016/0045-7825\(88\)90114-4](http://dx.doi.org/10.1016/0045-7825(88)90114-4)
- [39] S. Dey, M. S. Shephard, J. E. Flaherty, Geometry representation issues associated with p-version finite element computations, *Computer Methods in Applied Mechanics and Engineering* 150 (1997) 39 – 55, symposium on Advances in Computational Mechanics. doi:[http://dx.doi.org/10.1016/S0045-7825\(97\)00103-5](http://dx.doi.org/10.1016/S0045-7825(97)00103-5).
URL <http://www.sciencedirect.com/science/article/pii/S0045782597001035>
- [40] Z. Xie, R. Sevilla, O. Hassan, K. Morgan, The generation of arbitrary order curved meshes for 3D finite element analysis, *Computational Mechanics* 51 (3) (2013) 361–374. doi:10.1007/s00466-012-0736-4.
URL <http://dx.doi.org/10.1007/s00466-012-0736-4>
- [41] D. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, *Journal of Scientific Computing* 26 (3) (2006) 301–327. doi:10.1007/s10915-005-9070-8.
URL <http://dx.doi.org/10.1007/s10915-005-9070-8>
- [42] T. Hughes, J. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Computer Methods in Applied Mechanics and Engineering* 194 (39–41) (2005) 4135 – 4195. doi:<http://dx.doi.org/10.1016/j.cma.2004.10.008>.
URL <http://www.sciencedirect.com/science/article/pii/S0045782504005171>
- [43] R. Sevilla, S. Fernandez-Mendez, A. Huerta, Nurbs-enhanced finite element method for Euler equations, *International Journal for Numerical Methods in Fluids* 57 (9) (2008) 1051–1069. doi:10.1002/fld.1711.
URL <http://dx.doi.org/10.1002/fld.1711>

- [44] R. Sevilla, S. Fernandez-Mendez, A. Huerta, Nurbs-enhanced finite element method (nefem), *Archives of Computational Methods in Engineering* 18 (4) (2011) 441–484. doi:10.1007/s11831-011-9066-5.
URL <http://dx.doi.org/10.1007/s11831-011-9066-5>
- [45] A. N. Brooks, T. J. Hughes, Streamline upwind/ Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations, *Computer Methods in Applied Mechanics and Engineering* 32 (1982) 199 – 259. doi:[http://dx.doi.org/10.1016/0045-7825\(82\)90071-8](http://dx.doi.org/10.1016/0045-7825(82)90071-8).
URL <http://www.sciencedirect.com/science/article/pii/0045782582900718>
- [46] T. J. Hughes, G. R. Feij  so, L. Mazzei, J.-B. Quincy, The variational multiscale method – A paradigm for computational mechanics, *Computer Methods in Applied Mechanics and Engineering* 166 (1998) 3 – 24, *advances in Stabilized Methods in Computational Mechanics*. doi:[http://dx.doi.org/10.1016/S0045-7825\(98\)00079-6](http://dx.doi.org/10.1016/S0045-7825(98)00079-6).
URL <http://www.sciencedirect.com/science/article/pii/S0045782598000796>
- [47] T. J. R. Hughes, G. Sangalli, Variational multiscale analysis: the fine-scale Green’s function, projection, optimization, localization, and stabilized methods, *SIAM Journal on Numerical Analysis* 45 (2) (2007) 539–557. arXiv:<http://dx.doi.org/10.1137/050645646>, doi:10.1137/050645646.
URL <http://dx.doi.org/10.1137/050645646>
- [48] L. Demkowicz, J. Gopalakrishnan, A class of discontinuous Petrov-Galerkin methods. part i: The transport equation, *Computer Methods in Applied Mechanics and Engineering* 199 (23–24) (2010) 1558 – 1572. doi:<http://dx.doi.org/10.1016/j.cma.2010.01.003>.
URL <http://www.sciencedirect.com/science/article/pii/S0045782510000125>
- [49] L. Demkowicz, J. Gopalakrishnan, A class of discontinuous Petrov-Galerkin methods. ii. optimal test functions, *Numerical Methods for Partial Differential Equations* 27 (1) (2011) 70–105. doi:10.1002/num.20640.
URL <http://dx.doi.org/10.1002/num.20640>

- [50] J. Chan, L. Demkowicz, R. Moser, A DPG method for steady viscous compressible flow, *Computers & Fluids* 98 (2014) 69 – 90, 12th USNCCM mini-symposium of High-Order Methods for Computational Fluid Dynamics - A special issue dedicated to the 80th birthday of Professor Antony Jameson. doi:<http://dx.doi.org/10.1016/j.compfluid.2014.02.024>.
URL <http://www.sciencedirect.com/science/article/pii/S0045793014000875>
- [51] J. Chan, A DPG method for convection-diffusion problems, . Ph.D. Thesis, University of Texas at Austin (2013).
URL <https://repositories.lib.utexas.edu/handle/2152/21417>