Journal of Computational Physics 230 (2011) 8028–8056

Contents lists available at SciVerse ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Adaptive hierarchic transformations for dynamically *p*-enriched slope-limiting over discontinuous Galerkin systems of generalized equations

C. Michoski [a,*], C. Mirabito [a], C. Dawson [a], D. Wirasaet [b], E.J. Kubatko [c], J.J. Westerink [b]

[a] *Institute for Computational Engineering and Sciences (ICES), Computational Hydraulics Group (CHG), University of Texas, Austin, TX 78712, United States*
[b] *Computational Hydraulics Laboratory, Department of Civil Engineering and Geological Sciences, University of Notre Dame, Notre Dame, IN 46556, United States*
[c] *Department of Civil and Environmental Engineering and Geodetic Science, The Ohio State University, Columbus, OH 43210, United States*

ARTICLE INFO

Article history:
Received 11 December 2010
Received in revised form 24 May 2011
Accepted 4 July 2011
Available online 22 July 2011

Keywords:
Discontinuous Galerkin
Finite element
RKDG
Strong stability preserving (SSP)
Total variation diminishing (TVD)
Adaptive slope limiting
Shock capturing
Dynamic *p*-adaptivity
Dynamic *p*-enrichment
Error analysis
Advective transport
Hyperbolic PDE

ABSTRACT

We study a family of generalized slope limiters in two dimensions for Runge–Kutta discontinuous Galerkin (RKDG) solutions of advection-diffusion systems. We analyze the numerical behavior of these limiters applied to a pair of model problems, comparing the error of the approximate solutions, and discuss each limiter's advantages and disadvantages. We then introduce a series of coupled *p*-enrichment schemes that may be used as standalone dynamic *p*-enrichment strategies, or may be augmented via any in the family of variable-in-*p* slope limiters presented.

Published by Elsevier Inc.

1. Introduction

Generally when solving advection-diffusion equations – which are not strictly diffusion dominated – by way of, for example, finite element or finite volume techniques, one observes the presence of spurious oscillations in the solution space often brought about by the existence of shocks in the space of approximate solutions as well as from the presence of sharp and/or discontinuous profiles in the physical domain itself. Such ill-behaved approximate solutions have led to the development of numerous methods designed with the intent to consistently stabilize and "limit" the solution in order to deal with these oscillations, as they are seen to arise quite frequently in common scientific applications. For example, slope limiters are known to be of central importance in storm surge modeling [8,32] in order to obtain, for example, well-behaved solutions in the presence of complicated free–boundary conditions along adapting shorelines. Limiting regimes are also of substantial importance in quantum hydrodynamic systems [29,4] and surface wave models [26] where they are used to reduce the oscillations caused by mathematical dispersion terms (*i.e.* nonlinear third order spatial derivative terms) that pervade, for

* Corresponding author. Tel.: +1 512 471 7584.
E-mail address: michoski@ices.utexas.edu (C. Michoski).

0021-9991/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.jcp.2011.07.009

example, tunneling solutions. In fact, slope limiters are of fundamental importance in most applications in standard fluid dynamics, being employed commonly in compressible Navier–Stokes [31], Eulers [42], and magnetofluid [39,17] applications, not to mention the important role limiters play in the study of radiative transfer [15] and kinetic theory [18]; just to note a handful.

From a numerical perspective, it is clear that one should desire that even shock dominated solutions, like both their smooth and non-limited counterparts, converge in p as $p \nearrow p_{\max}$. However, such convergence is fundamentally coupled to the behavior of the error accumulation with respect to one's chosen slope limiting methodology, which, it turns out, must operate over a larger number of degrees of freedom, respectively, as p increases. For example, in a hierarchical basis (as shown explicitly below) the degrees of freedom grow nonlinearly as a function of p and each degree of freedom ends up carrying information of potentially pathological (or undesirable) overshoots and undershoots which have developed over the native (or non-limited) solution space. It turns out that this complication introduces a substantial technical difficulty in practice, which many papers on numerical shock capturing [16,14,25,28,1,27,6,7,12] tend to avoid addressing directly. Most noteworthy is the observation that slopelimiters tend to limit the coefficients in their chosen basis independently of each other, in the sense that each component is adjusted based on information about the surrounding solution on a relatively local submanifold of the total domain. It then follows directly that the application of the limiter grows nonlinearly in each timestep as a function of p. Since a limiter *de jure* introduces error into the FEM solution space each time it operates on the FEM solution, more applications of it (iteratively) to the solution space should, as a general rule, lead to greater error accumulation assuming the first application always introduces approximately the same amount of error. In fact, this is what we observe in each of our limiters *de facto*. However, we offer an alternative approach to this problem below which is both highly efficient and consistent with the more general setting of *hp*-adaptivity.

It perhaps comes as no surprise that the same type of complications do not arise with respect to the mesh size h. That is, the convergence in h as $h \searrow h_{\min}$ tends to arise as a natural consequence of the usual h convergence, where convergence seems essentially guaranteed in most reasonable limiting regimes, while the order of convergence most certainly is not [27]. This issue raises another subtle technical difficulty which we will not address directly in this paper, though we will mention its importance in the proper context.

Another important technicality pertaining to computational efficiency arises with respect to the well-known Courant–Friedrichs–Lewy (CFL) condition. In this setting the temporal discretization is (partially) bounded from above by the spatial discretization. That is, in order to reach a higher order accuracy at a fixed value of h one must project onto a higher order polynomial basis in p, thus reducing the admissible timestep Δt of the scheme – which obeys an inverse relation by virtue of the CFL condition: $\Delta t \propto 1/p$ as discussed in [38].

Since this p-dependence on the solution accuracy runs counter to the CFL restriction in a practical computational sense, substantial effort has been invested in developing "smart schemes" which in some way are able to "sense" the appropriate place (e.g. $\boldsymbol{x} \in \Omega$) within the solution domain to enrich the polynomial order p, while keeping other areas either unaffected or adaptively de-enriching areas of "less importance." The ultimate goal of these schemes is to attempt to substantially improve the computational efficiency of the numerical scheme without ceding notable accuracy in the solution. In fact, it is generally theoretically true that when one couples adaptive h-refinement to p-enrichment (i.e. *hp*-adaptivity) an exponential improvement in the convergence scaling of the solution may be obtained [10]. However, dynamic adaptive h-refinement is beyond the current scope of this paper and will be addressed elsewhere.

On the other hand several different schemes have been developed for dynamic p-enrichment of solutions (independent of h-refinement), though many suffer the added complexity of being extremely system (PDE) dependent. The advantage of system dependent regimes is that such schemes often display very close coupling to the physics of the solution (e.g. energy methods as discussed in [30]). The disadvantage is, of course, that the scheme is very system dependent and hence whenever a variable is added or changed the entire scheme must be recalculated; which is particularly troublesome for systems of equations which are not mathematically well-posed. Other schemes rely on – in the FEM setting for example – the generalized features of numerical variational solutions and as a consequence often depend strongly on a relatively large array of user defined constants. These schemes are obviously quite attractive from a meta-application perspective, where being able to deal with generalizable systems displaying complicated initial-boundary data generates great allure in itself. In this paper we focus on the latter class of solutions, as we are interested in schemes which may apply to a large and generalized class of PDEs, without being bound, *ab initio*, to any one particular system of equations.

Nevertheless, in the present paper we restrict ourselves to the class of discontinuous Galerkin finite element methods, where the underlying basis is chosen such as to signify a ubiety of discontinuous solutions – that is, we turn our focus in this paper to shock-dominated solutions. In this setting we are interested in the situation where continuously adaptive p-enrichment is coupled to an adapting-in-p slope limiting regime. We view this setting as very attractive, since the discontinuity sensors for p-adaptation schemes are well established [33,41] to be good sensors for slope limiting methodologies as well, where the p-enrichment leads to stability and efficiency of the scheme while the slope-limiting further stabilizes the presence of spurious oscillations emerging near pathological discontinuities as so approximated to order p.

The outline of this paper is as follows. In Section 2 we present our generalized setting, which can be summarized as: given an advection-diffusion system of equations, consider the initial free boundary value problem recast into the weak formulation and spatially discretized. We then take a temporal discretization via a RKSSP DG approach in which we obtain the form of our approximate solutions. Our formulation is general, while our examples focus on problems of hyperbolic transport saving the more general applications for the sequel to this paper. In Section 3 we introduce a number of slope limiters consistent

with any order p basis. The first is the vertex limiter regime of [25], the second the classical Barth–Jespersen limiter [6], and the third and fourth are minor adaptations of the former two limiters made with an eye towards improving the L^2– error convergence by adjusting a so-called "blind spot" present in the previous schemes at higher order p. The fifth approach is comprised of a family of hierarchical reconstruction approaches [1,27], while the sixth regime is a linear restriction method that can be viewed as a generalization of a limiter originally sketched in [7]. The final limiting regime we present is a mixed extension of the previous limiters referred to here as a hierarchic recombination approach. Section 4 then provides numerical experimentation using the schemes presented in Section 3 – namely a classical advective scalar transport problem, and a stationary solution to a closely related problem with highly singular initial data. We also show some convergence results on an analytic test case. Finally, in Section 5 we present the adaptive p-enrichment schemes, which are fully coupled to the slope limiters from Section 4 *ab initio*. These come in two basic types, the first for (*heuristically*) smooth solutions, and the second for solutions demonstrating (vaguely) "appreciable gradients."

2. Advection-diffusion systems in the DG formalism

We are interested in solutions to an initial-boundary value problem for a generalized advection-diffusion system of arbitrarily mixed hyperbolic-parabolic type in $\Omega \times (0,T)$, where $\Omega \subset \mathbb{R}^2$ with boundary $\partial\Omega$, such that the system satisfies:

$$\boldsymbol{U}_t + \boldsymbol{F}_x - \boldsymbol{G}_x = \boldsymbol{g}, \quad \text{given initial conditions} \quad \boldsymbol{U}_{|t=0} = \boldsymbol{U}_0, \tag{1}$$

and generalized componentwise Robin boundary values

$$a_i U_i + \nabla_x U_{i,x}(b_i \cdot \boldsymbol{n} + c_i \cdot \boldsymbol{\tau}) - f_i = 0, \quad \text{on } \partial\Omega. \tag{2}$$

That is, the system is comprised of a generalized m-dimensional state vector $\boldsymbol{U} = \boldsymbol{U}(t,\boldsymbol{x}) = (U_1,\ldots,U_m)$, an advective flux matrix $\boldsymbol{F} = \boldsymbol{F}(\boldsymbol{U})$, a viscous flux matrix $\boldsymbol{G} = \boldsymbol{G}(\boldsymbol{U},\boldsymbol{U}_x)$, and a source term $\boldsymbol{g} = \boldsymbol{g}(t,\boldsymbol{x}) = (g_1,\ldots,g_m)$, where $\boldsymbol{x} \in \mathbb{R}^2$ and $t \in (0,T)$. The vectors $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{f}$ are comprised of the m functions, $a_i = a_i(t,\boldsymbol{x})$, $b_i = b_i(t,\boldsymbol{x})$, $c_i = c_i(t,\boldsymbol{x})$ and $f_i = f_i(t,\boldsymbol{x})$ for $i = 1,\ldots,m$, where $\boldsymbol{n}$ denotes the unit outward pointing normal and $\boldsymbol{\tau}$ the unit tangent vector.

In addition, because we are interested in approximate numerical solutions of the form of [2,3] restricted in part to the family of methods for elliptic equations, we rewrite (1) as a coupled system in terms of an auxiliary variable $\boldsymbol{\Sigma}$, such that

$$\boldsymbol{U}_t + \boldsymbol{F}_x - \boldsymbol{G}_x = \boldsymbol{g}, \quad \text{and} \quad \boldsymbol{\Sigma} = \boldsymbol{U}_x, \tag{3}$$

where we have substituted in the viscous flux matrix the auxiliary term, so that $\boldsymbol{G} = \boldsymbol{G}(\boldsymbol{U},\boldsymbol{\Sigma})$.

For notational completeness we adopt the following discretization scheme motivated by [31,13]. Take an open $\Omega \subset \mathbb{R}^2$ with boundary $\partial\Omega$, given $T > 0$ such that $\mathcal{Q}_T = ((0,T) \times \Omega)$. Let $\mathcal{T}_h$ denote the partition of the closure of the polygonal triangulation of Ω, which we denote Ω_h, into a finite number of polygonal elements denoted Ω_e, such that $\mathcal{T}_h = \{\Omega_{e_1}, \Omega_{e_2}, \ldots, \Omega_{e_{ne}}\}$, for $ne \in \mathbb{N}$ the number of elements in Ω_h. In this work we define the mesh diameter h to satisfy $h = \min_{ij}(d_{ij})$ for the distance function $d_{ij} = d(\boldsymbol{x}_i,\boldsymbol{x}_j)$ and elementwise edge vertices $\boldsymbol{x}_i, \boldsymbol{x}_j \in \partial\Omega_e$ when the mesh is structured and regular. For unstructured meshes we mean the average value of h over the mesh.

Now, let Γ_{ij} denote the edge shared by two neighboring elements Ω_{e_i} and Ω_{e_j}, and for $i \in I \subset \mathbb{Z}^+ = \{1,2,\ldots\}$ define the indexing set $r(i) = \{j \in I : \Omega_{e_j}$ is a neighbor of $\Omega_{e_i}\}$. Let us denote all Ω_{e_i} containing the boundary $\partial\Omega_h$ by S_j and letting $I_B \subset \mathbb{Z}^- = \{-1,-2,\ldots\}$ define $s(i) = \{j \in I_B : S_j$ is an edge of $\Omega_{e_i}\}$ such that $\Gamma_{ij} = S_j$ for $\Omega_{e_i} \in \Omega_h$ when $S_j \in \partial\Omega_{e_i}, j \in I_B$. Then for $\Xi_i = r(i) \cup s(i)$, we have

$$\partial\Omega_{e_i} = \bigcup_{j \in \Xi(i)} \Gamma_{ij}, \quad \text{and} \quad \partial\Omega_{e_i} \cap \partial\Omega_h = \bigcup_{j \in s(i)} \Gamma_{ij}.$$

We are interested in obtaining an approximate solution to $\boldsymbol{U}$ at time t on the finite dimensional space of discontinuous piecewise polynomial functions over Ω restricted to $\mathcal{T}_h$, given as

$$S_h^p(\Omega_h,\mathcal{T}_h) = \left\{ v : v_{|\Omega_{e_i}} \in \mathcal{P}^p(\Omega_{e_i}) \ \forall \Omega_{e_i} \in \mathcal{T}_h \right\}$$

for $\mathcal{P}^p(\Omega_{e_i})$ the space of degree $\leqslant p$ polynomials over Ω_{e_i}.

Choosing a set of degree p polynomial basis functions $N_\ell \in \mathcal{P}^p(\Omega_{e_i})$ for $\ell = 1,\ldots,n_p$ corresponding to the degree of freedom, we can denote the state vector at time t over Ω_{e_i}, by

$$\boldsymbol{U}_h(t,\boldsymbol{x}) = \sum_{\ell=1}^{n_p} \boldsymbol{U}_\ell^i(t) N_\ell^i(\boldsymbol{x}), \ \forall x \in \Omega_{e_i}, \tag{4}$$

where the N_ℓ^i's are the finite element shape functions in the DG setting, and the $\boldsymbol{U}_\ell^i$'s correspond to the unknowns. We characterize the finite dimensional test functions

$$\boldsymbol{v}_h, \boldsymbol{\omega}_h \in W^{k,q}(\Omega_h,\mathcal{T}_h), \quad \text{by} \quad \boldsymbol{v}_h(x) = \sum_{\ell=1}^{n_p} \boldsymbol{v}_\ell^i N_\ell^i(x) \quad \text{and} \quad \boldsymbol{\omega}_h(x) = \sum_{\ell=1}^{n_p} \boldsymbol{\omega}_\ell^i N_\ell^i(x)$$

where $\boldsymbol{v}_\ell^i$ and $\boldsymbol{\omega}_\ell^i$ are the coordinates of the test functions in each Ω_{e_i}, and with the broken Sobolev space over the partition $\mathcal{T}_h$ defined by

$$W^{k,q}(\Omega_h,\mathcal{T}_h)=\left\{w: w_{|\Omega_{e_i}}\in W^{k,q}(\Omega_{e_i})\ \forall\Omega_{e_i}\in\mathcal{T}_h\right\}.$$

Thus, for $\boldsymbol{U}$ a classical solution to (3), multiplying by $\boldsymbol{v}_h$ or $\boldsymbol{\omega}_h$ and integrating elementwise by parts yields the coupled system:

$$\begin{aligned}&\frac{d}{dt}\int_{\Omega_{e_i}}\boldsymbol{U}\cdot\boldsymbol{v}_h dx+\int_{\Omega_{e_i}}(\boldsymbol{F}\cdot\boldsymbol{v}_h)_x dx-\int_{\Omega_{e_i}}\boldsymbol{F}:\boldsymbol{v}_x^h dx-\int_{\Omega_{e_i}}(\boldsymbol{G}\cdot\boldsymbol{v}_h)_x dx+\int_{\Omega_{e_i}}\boldsymbol{G}:\boldsymbol{v}_x^h dx=\int_{\Omega_{e_i}}\boldsymbol{v}_h\cdot\boldsymbol{g}dx,\\&\int_{\Omega_{e_i}}\boldsymbol{\Sigma}\cdot\boldsymbol{\omega}_h dx-\int_{\Omega_{e_i}}(\boldsymbol{U}\cdot\boldsymbol{\omega}_h)_x dx+\int_{\Omega_{e_i}}\boldsymbol{U}:\boldsymbol{\omega}_x^h dx=0,\end{aligned}\tag{5}$$

where $(:)$ denotes the scalar product.

Now, let $\boldsymbol{n}_{ij}$ be the unit outward normal to $\partial\Omega_{e_i}$ on Γ_{ij}, and let $v_{|\Gamma_{ij}}$ and $v_{|\Gamma_{ji}}$ denote the values of v on Γ_{ij} considered from the interior and the exterior of Ω_{e_i}, respectively. Then by choosing componentwise approximations in (5) by substituting in (4), we arrive with the approximate form of the first term of (5) given by,

$$\frac{d}{dt}\int_{\Omega_{e_i}}\boldsymbol{U}_h\cdot\boldsymbol{v}_h dx\approx\frac{d}{dt}\int_{\Omega_{e_i}}\boldsymbol{U}\cdot\boldsymbol{v}_h dx,\tag{6}$$

the second term using an inviscid numerical flux $\boldsymbol{\Phi}_i$, by

$$\widetilde{\boldsymbol{\Phi}}_i(\boldsymbol{U}_h|_{\Gamma_{ij}},\boldsymbol{U}_h|_{\Gamma_{ji}},\boldsymbol{v}_h)=\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\boldsymbol{\Phi}(\boldsymbol{U}_h|_{\Gamma_{ij}},\boldsymbol{U}_h|_{\Gamma_{ji}},\boldsymbol{n}_{ij})\cdot\boldsymbol{v}_h|_{\Gamma_{ij}}d\Xi\approx\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\sum_{l=1}^{2}(\boldsymbol{F})_l\cdot(n_{ij})_l\boldsymbol{v}_h|_{\Gamma_{ij}}d\Xi,\tag{7}$$

and the third term in (5) by,

$$\boldsymbol{\Theta}_i(\boldsymbol{U}_h,\boldsymbol{v}_h)=\int_{\Omega_{e_i}}\boldsymbol{F}_h:\boldsymbol{v}_x^h dx\approx\int_{\Omega_{e_i}}\boldsymbol{F}:\boldsymbol{v}_x^h dx.\tag{8}$$

Next we approximate the boundary viscous term of (5) using a generalized viscous flux $\widehat{\mathcal{G}}$ such that,

$$\mathcal{G}_i(\boldsymbol{\Sigma}_h,\boldsymbol{U}_h,\boldsymbol{v}_h)=\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\widehat{\mathcal{G}}(\boldsymbol{\Sigma}_h|_{\Gamma_{ij}},\boldsymbol{\Sigma}_h|_{\Gamma_{ji}},\boldsymbol{U}_h|_{\Gamma_{ij}},\boldsymbol{U}_h|_{\Gamma_{ji}},\boldsymbol{n}_{ij})\cdot\boldsymbol{v}_h|_{\Gamma_{ij}}d\Xi\approx\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\sum_{l=1}^{2}(\boldsymbol{G})_l\cdot(n_{ij})_l\boldsymbol{v}_h|_{\Gamma_{ij}}d\Xi,\tag{9}$$

while the second viscous term is approximated by:

$$\mathcal{N}_i(\boldsymbol{\Sigma}_h,\boldsymbol{U}_h,\boldsymbol{v}_h)=\int_{\Omega_{e_i}}\boldsymbol{G}_h:\boldsymbol{v}_x^h dx\approx\int_{\Omega_{e_i}}\boldsymbol{G}:\boldsymbol{v}_x^h dx.\tag{10}$$

Finally the source $\boldsymbol{g}$ term of (5) is given to satisfy

$$\mathcal{H}_i(\boldsymbol{v}_h,\boldsymbol{x}_h,t)=\int_{\Omega_{e_i}}\boldsymbol{v}_h\cdot\boldsymbol{g}_h dx\approx\int_{\Omega_{e_i}}\boldsymbol{v}_h\cdot\boldsymbol{g}dx.\tag{11}$$

For the auxiliary equation in (5) we expand it such that the approximate solution satisfies:

$$\mathcal{Q}_i(\widehat{\boldsymbol{U}},\boldsymbol{\Sigma}_h,\boldsymbol{U}_h,\boldsymbol{\omega}_h,\boldsymbol{\omega}_x^h)=\int_{\Omega_{e_i}}\boldsymbol{\Sigma}_h\cdot\boldsymbol{\omega}_h dx+\int_{\Omega_{e_i}}\boldsymbol{U}_h:\boldsymbol{\omega}_x^h dx-\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\widehat{\boldsymbol{U}}(\boldsymbol{U}_h|_{\Gamma_{ij}},\boldsymbol{U}_h|_{\Gamma_{ji}},\boldsymbol{\omega}_h|_{\Gamma_{ij}},\boldsymbol{n}_{ij})d\Xi=0,\tag{12}$$

where,

$$\sum_{i\in I}\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\widehat{\boldsymbol{U}}(\boldsymbol{U}_h|_{\Gamma_{ij}},\boldsymbol{U}_h|_{\Gamma_{ji}},\boldsymbol{\omega}_h|_{\Gamma_{ij}},\boldsymbol{n}_{ij})d\Xi\approx\sum_{i\in I}\sum_{j\in\Xi(i)}\int_{\Gamma_{ij}}\sum_{l=1}^{2}(\boldsymbol{U})_l\cdot(n_{ij})_l\boldsymbol{\omega}_h|_{\Gamma_{ij}}d\Xi$$

given $\widehat{\boldsymbol{U}}$ a generalized numerical flux, and where

$$\int_{\Omega_{e_i}}\boldsymbol{\Sigma}_h\cdot\boldsymbol{\omega}_h dx\approx\int_{\Omega_{e_i}}\boldsymbol{\Sigma}\cdot\boldsymbol{\omega}_h dx,\quad\text{and}\quad\int_{\Omega_{e_i}}\boldsymbol{U}_h:\boldsymbol{\omega}_x^h dx\approx\int_{\Omega_{e_i}}\boldsymbol{U}:\boldsymbol{\omega}_x^h dx.$$

Combining the above approximations and setting, $\mathcal{X}=\sum_{\Omega_{e_i}\in\mathcal{T}_h}\mathcal{X}_i$, while denoting the inner product

$$(\boldsymbol{a}_h^n,\boldsymbol{b}_h)_{\Omega_{\mathcal{G}}}=\sum_{\Omega_{e_i}\in\mathcal{T}_h}\int_{\Omega_{e_i}}\boldsymbol{a}_h^n\cdot\boldsymbol{b}_h dx,$$

we arrive at our approximate solution to (3) as the pair of functions $(\boldsymbol{U}_h,\boldsymbol{\Sigma}_h)$ for all $t\in(0,T)$ satisfying:

The Discontinuous Galerkin formulation

$$
\begin{aligned}
&(a)\quad \boldsymbol{U}_h \in C^1((0,T);S_h^p), \quad \boldsymbol{\Sigma}_h \in S_h^p,\\
&(b)\quad \tfrac{d}{dt}(\boldsymbol{U}_h,\boldsymbol{v}_h)_{\Omega_{\mathcal{G}}} + \widetilde{\boldsymbol{\Phi}}(\boldsymbol{U}_h,\boldsymbol{v}_h) - \boldsymbol{\Theta}(\boldsymbol{U}_h,\boldsymbol{v}_h) - \mathcal{G}(\boldsymbol{\Sigma}_h,\boldsymbol{U}_h,\boldsymbol{v}_h) + \mathcal{N}(\boldsymbol{\Sigma}_h,\boldsymbol{U}_h,\boldsymbol{v}_h) = \mathcal{H}(\boldsymbol{v}_h,\boldsymbol{x}_h,t),\\
&(c)\quad \mathcal{Q}(\widehat{\boldsymbol{U}},\boldsymbol{\Sigma}_h,\boldsymbol{U}_h,\boldsymbol{\omega}_h,\boldsymbol{\omega}_x^h) = 0,\\
&(d)\quad \boldsymbol{U}_h(0) = \Pi_h \boldsymbol{U}_0,
\end{aligned}
\tag{13}
$$

where Π_h is a projection operator onto the space of discontinuous piecewise polynomials S_h^p, and where below we always utilize a standard L^2-projection, given for a function $\boldsymbol{f}_0 \in L^2(\Omega_{e_i})$ such that our approximate projection $\boldsymbol{f}_{0,h} \in L^2(\Omega_{e_i})$ is obtained by solving, $\int_{\Omega_{e_i}} \boldsymbol{f}_{0,h}\boldsymbol{v}_h dx = \int_{\Omega_{e_i}} \boldsymbol{f}_0 \boldsymbol{v}_h dx$. We provide several explicit simplified examples of this generalized formalism below, though in the followup paper we address models motivated by more complicated dynamics (e.g. see [20,24,8,22,21]) that employ the full system of (13) including multicomponent reaction-advection-diffusion and free boundary conditions, etc.

The discretization in time follows now directly from (13), where we employ a family of SSP (strong stability preserving, or often "total variation diminishing (TVD)") Runge–Kutta schemes as discussed in [36,37]. That is, for the generalized SSP Runge–Kutta scheme we rewrite (13*b*) in the form: $\mathbf{M}\boldsymbol{U}_t = \mathbf{R}$, where $\boldsymbol{U} = (\boldsymbol{U}_1,\ldots,\boldsymbol{U}_p)$ for each element from (4), where $\mathbf{R} = \mathbf{R}(\boldsymbol{U},\boldsymbol{\Sigma})$ is the advection-diffusion contribution along with the source term, and where $\mathbf{M}$ is the usual mass matrix. Then the generalized s stage of order γ SSP Runge–Kutta method (denoted SSP(s,γ) or RKSSP(s,γ)) may be written to satisfy:

$$
\begin{aligned}
&\boldsymbol{U}^{(0)} = \boldsymbol{U}^n,\\
&\boldsymbol{U}^{(i)} = \sum_{r=0}^{i-1}\left(\alpha_{ir}\boldsymbol{U}^r + \Delta t\beta_{ir}\mathbf{M}^{-1}\mathbf{R}^r\right), \quad \text{for } i = 1,\ldots,s\\
&\boldsymbol{U}^{n+1} = \boldsymbol{U}^{(s)},
\end{aligned}
\tag{14}
$$

where $\mathbf{R}^r = \mathbf{R}(\boldsymbol{U}^r,\boldsymbol{\Sigma}^r,\boldsymbol{x},t^n + \delta_r\Delta t)$ and the solution at the nth timestep is given as $\boldsymbol{U}^n = \boldsymbol{U}_{|t=t^n}$ and at the nth plus first timestep by $\boldsymbol{U}^{n+1} = \boldsymbol{U}_{|t=t^{n+1}}$, with $t^{n+1} = t^n + \Delta t$. The α_{ir} and β_{ir} are the coefficients arising from the Butcher Tableau, and the third argument in $\mathbf{R}^r$ corresponds to the time-lag complication arising in the constraints of the TVD formalism. That is $\delta_r = \sum_{l=0}^{r-1}\mu_{rl}$, where $\mu_{ir} = \beta_{ir} + \sum_{l=r+1}^{i-1}\mu_{lr}\alpha_{il}$, where we have taken that $\alpha_{ir} \geqslant 0$ satisfying $\sum_{r=0}^{i-1}\alpha_{ir} = 1$.

It is often possible to optimize the generalized SSP schemes of (14) by restricting to an optimization class of stage exceeding order SSP Runge–Kutta time discretizations of [21] as long as $p \leqslant 3$. This class of SSP Runge–Kutta schemes has the advantage of optimizing the polynomial order p of the approximate solution $\boldsymbol{U}_h$ with respect to the r stage of the SSP Runge–Kutta scheme (incidentally satisfying SSP$(r,p+1)$) in order to minimize the effect of the rigid constraint introduced by the CFL condition on the timestep Δt. The limitation on p (i.e. requiring $p \leqslant 3$) is generally more restrictive than we encounter here, and thus, as will become apparent below, in the context of dynamic p-enriched slope limited solutions we are generally unable to exploit these optimization schemes directly.

3. A dynamic-in-*p* family of slope limiters

3.1. A transformation of basis

Finite element approximate solutions are recovered with respect to any number of different finite element bases (e.g. Legendre polynomials, Lagrange polynomials, Labotto polynomials, Jacobi polynomials, Gegenbauer polynomials, Chebyshev polynomials, Bernstein polynomials, Gram–Schmidt polynomials, NURBS, *T*-splines, Wachspress functions, etc.). As a consequence of this, it is often advantageous to develop a strategy to transform into a specific basis in order to limit the solution, and then to transform back into the native bases to perform the remainder of the calculations. This occurs because some slope limiting regimes use fundamental properties of a certain choice of basis in order to develop a limiting strategy. We provide an explicit example of this procedure below, in the case of transforming between the Dubiner basis and the Taylor basis; or as we denote it below: by way of the invertible Dubiner–Taylor transform $\mathcal{L}$. We also not here that for the sake of providing explicit calculations, we restrict below to triangular meshes, though the formalism can be easily extended to a more general framework.

Take a solution vector $\boldsymbol{U}$ with approximate form $\boldsymbol{U}_h \approx \boldsymbol{U}$ as given by (4), and project it onto the degree p Dubiner basis such that:

$$
\boldsymbol{U}_h(\boldsymbol{x},t)_{|\Omega_e} = \sum_{0<i+j\leqslant p} \boldsymbol{U}_{ij}(t)\phi_{ij}(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \Omega_e,
\tag{15}
$$

where the $\phi_{ij}(\boldsymbol{x})$ are the Dubiner basis functions for each degree of freedom in the solution vector.

It is our aim to take this approximate solution $\boldsymbol{U}_h$ and limit it with respect to the k-th order Taylor basis via, for example, the vertex slope limiter of [25] and the hierarchical reconstruction of [1,27], etc. Now, the Taylor basis in two dimensions is given to arbitrary differential order $k \geqslant (i+j)$ by the Taylor polynomial centered at c via:

$$\boldsymbol{U}_h(x,y) = \boldsymbol{U}_h|_c + \sum_{0<i+j\leqslant k} \frac{(x-x_c)^i(y-y_c)^j}{i!j!}\left(\frac{\partial^{i+j}\boldsymbol{U}_h}{\partial x^i \partial y^j}\right)\Bigg|_c, \tag{16}$$

where x_c and y_c are explicitly chosen as the values at the centroid $c = (x_c, y_c)$ of each finite element Ω_e in the physical space Ω – that is, each Ω_{e_i} taking coordinates $\boldsymbol{x} \in \Omega$ – where it is clear that $i + j \geqslant 1$ in the sum denotes the differential order of the basis expansion (i.e. the indices satisfy $i, j \in \mathbb{N}$).

Now, for cell averages satisfying $\overline{\boldsymbol{U}} = |\Omega_e|^{-1}\int_{\Omega_e} \boldsymbol{U}_h d\boldsymbol{x}$, the average of (16) may be simply written by

$$\overline{\boldsymbol{U}}_h(x,y) = \boldsymbol{U}_h|_c + \sum_{0<i+j\leqslant k} \overline{\left(\frac{(x-x_c)^i(y-y_c)^j}{i!j!}\right)\left(\frac{\partial^{i+j}\boldsymbol{U}_h}{\partial x^i \partial y^j}\right)}\Bigg|_c \tag{17}$$

such that subtracting (17) from (16) formally yields:

$$\boldsymbol{U}_h = \overline{\boldsymbol{U}}_h + \sum_{0<i+j\leqslant k} \left(\frac{(x-x_c)^i(y-y_c)^j}{i!j!} - \overline{\frac{(x-x_c)^i(y-y_c)^j}{i!j!}}\right)\left(\frac{\partial^{i+j}\boldsymbol{U}_h}{\partial x^i \partial y^j}\right)\Bigg|_c. \tag{18}$$

Additional analysis (also see [28]) has shown empirically that the conditioning of the system in the Taylor basis (with respect to, for example, the invertibility of the Taylor mass matrix) is improved by rescaling with respect to the cell averages over the local bounds, given by $\psi\Delta x = (x_{\max} - x_{\min})$ and $\psi\Delta y = (y_{\max} - y_{\min})$ where $\psi = p$ for $p > 2$, and $\psi = 2$ for $p \leqslant 2$. It is useful to note here that in the master element representation these scalings are merely a pair of constants, while in the physical element representation they will in general vary elementwise.

Then we are interested in implementing a locally renormalized Taylor basis prescribed with respect to the physical space Ω given componentwise via the explicit formulation:

$$\varphi_{ij}(x,y) = \left(\frac{(x-x_c)^i}{i!\Delta x^i}\right)\left(\frac{(y-y_c)^j}{j!\Delta y^j}\right) - \overline{\left(\frac{(x-x_c)^i}{i!\Delta x^i}\right)\left(\frac{(y-y_c)^j}{j!\Delta y^j}\right)}, \tag{19}$$

where again cell averages are chosen to satisfy,

$$\overline{\left(\frac{(x-x_c)^i}{i!\Delta x^i}\right)\left(\frac{(y-y_c)^j}{j!\Delta y^j}\right)} = \frac{1}{|\Omega_e|}\int_{\Omega_e}\left(\frac{(x-x_c)^i}{i!\Delta x^i}\right)\left(\frac{(y-y_c)^j}{j!\Delta y^j}\right)dxdy.$$

Notice also that the constant terms of (18) vanish with respect to the barycenter c, which is just to say that the value of the centroid is by definition the cell average. Moreover, note that the renormalization vanishes for linear terms, since the average value is achieved at the centroid c (see [25] for more examples at order $p \leqslant 2$).

Now we see that (18) satisfies in vector form that:

$$\boldsymbol{U}_h = \overline{\boldsymbol{U}}_h\varphi_{00} + \sum_{0<i+j\leqslant k}\varphi_{ij}\left\{\left(\frac{\partial^{i+j}\boldsymbol{U}_h}{\partial x^i \partial y^j}\right)\Bigg|_c \Delta x^i \Delta y^j\right\}, \tag{20}$$

where we have denoted our effective Taylor basis $\varphi_{ij} \in \mathbb{R}[\Omega]$, such that $\varphi_{ij} = \varphi_{ij}(\boldsymbol{x})$ in the polynomial ring $\mathbb{R}[\Omega]$ such that $\boldsymbol{x} \in \Omega$. By the polynomial ring $\mathbb{R}[\Omega]$ we simply mean the set of all polynomials with coefficients in $\mathbb{R}$ centered at a particular $\boldsymbol{x} \in \Omega$. The bracketed terms in (20) here represent our effective scaled coefficients, and from here forward the scaling parameters will generally be suppressed for notational simplicity.

We will further make use of the fact that (20) may be viewed as the k-jet over $\mathbb{R}^2$. That is, for $\Omega \subset \mathbb{R}^2$ and components of the approximate solution vector $\boldsymbol{U}_h$ the Taylor basis functions φ_{ij} comprise the abstract indeterminates of the k-jet $(J_c^k\boldsymbol{U}_h)(\varphi_{ij})$ centered at c, in that by definition

$$\boldsymbol{U}_h|_{\Omega_{e_j}} := (J_c^k\boldsymbol{U}_h)(\varphi_{ij}), \tag{21}$$

such that our approximate solutions are elements of the abstract jet space $\boldsymbol{U}_h|_{\Omega_{e_j}} \in J_c^k(\mathbb{R}^2, \Omega)$. The jet space $J_c^k(\mathbb{R}^2, \Omega)$ is simply defined as the set of equivalence classes of k-jets which agree to order k and map between the Cartesian plane and an element of Ω, as clearly our approximate solutions in the Taylor (polynomial) basis do. By the set of equivalence classes of k-jets which agree to order k, we mean for any two solutions $\boldsymbol{V}_h|_{\Omega_{e_j}}$ and $\boldsymbol{U}_h|_{\Omega_{e_j}}$ in the Taylor basis restricted to Ω_{e_j} – that is k-jets – the equivalence relation $\boldsymbol{U}_h|_{\Omega_{e_j}} - \boldsymbol{V}_h|_{\Omega_{e_j}} \sim 0$ holds to order k.

In this sense, an effective slope limiter may be viewed as a stabilization rescaling of the jet by the k coefficients $\alpha^{(i+j)}$ (as derived in Section 4), such that the slope limited approximate solution vector $\boldsymbol{U}_h^v$ is formally the same as the stabilized k-jet centered at c; that is $\boldsymbol{U}_h^v|_{\Omega_{e_j}} := (J_c^k\boldsymbol{\alpha}\boldsymbol{U}_h)(\varphi_{ij})$ where both the approximate solution and the corresponding limited approximate solution are each, respectively, elements of the same abstract jet space $\boldsymbol{U}_h|_{\Omega_{e_j}}, \boldsymbol{U}_h^v|_{\Omega_{e_j}} \in J_c^k(\mathbb{R}^2, \Omega)$ when letting the equivalence relation $\sim$ be approximate $\sim_h$ (i.e. approximate with respect to the solution order accuracy but with vanishing asymptotics).

Now, in order to work between the Taylor basis representation φ_{ij} and the Dubiner basis representation ϕ_{ij}, we must construct a transformation between the physical element space Ω and the master element space $\mathcal{M}$, as well as a transformation

between the two (abstract) polynomial bases. Below we make these mappings explicit, and refer to them collectively in this work as the Dubiner–Taylor transform, which is given by the invertible mapping $\mathcal{L} : \mathbb{R}[\mathcal{M}] \to J_c^k(\mathbb{R}^2, \Omega)$.

First consider the usual Dubiner basis functions in the master element space componentwise $\phi_{ij} \in \mathbb{R}[\mathcal{M}]$ for $\phi_{ij} = \phi_{ij}(\boldsymbol{x})$, and $\mathbb{R}[\mathcal{M}]$ the polynomial ring in coordinates $\boldsymbol{x} \in \mathcal{M}$ given by:

$$\phi_{ij} = P_i^{0,0}(\psi_1)\left(\frac{1-\psi_2}{2}\right)^i P_j^{2i+1,0}(\psi_2), \tag{22}$$

using p-th order Jacobi polynomials with weights α, β, such that $P_p^{\alpha,\beta}(\cdot)$ is evaluated with respect to the coordinates $\boldsymbol{x} = (\xi, \eta)$ of the master triangle element, where the master element quadrilateral transformation in the Dubiner mapping provides that: $\psi_1 = \left(\frac{2(1+\xi)}{(1-\eta)} - 1\right)$ and $\psi_2 = \eta$, such that $\psi_1 = \psi_1(\boldsymbol{x})$ and $\psi_2 = \psi_2(\boldsymbol{x})$.

Now, consider the two state vectors, $\boldsymbol{\phi} = (\phi_{00}, \phi_{10}, \ldots, \phi_{cd})^T$ and $\boldsymbol{\varphi} = (\varphi_{00}, \varphi_{10}, \ldots, \varphi_{cd})^T$, where in the lexicographic ordering (described in detail in Section 3.2) we have $c + d \leqslant p$. Now, we may transform between the master and physical element representations of our components $\varphi = \varphi(x,y)$ and $\phi = \phi(\xi,\eta)$ using the following affine mapping:

$$x = -\frac{1}{2}\{\xi(x_1 - x_2) + \eta(x_1 - x_3) - x_2 - x_3\}, \quad y = -\frac{1}{2}\{\xi(y_1 - y_2) + \eta(y_1 - y_3) - y_2 - y_3\}, \tag{23}$$

with inverse given by

$$\begin{aligned} \xi &= \chi\left\{(y_3 - y_1)\left(x - \frac{1}{2}(x_2 + x_3)\right) + (x_1 - x_3)\left(y - \frac{1}{2}(y_2 + y_3)\right)\right\}, \\ \eta &= \chi\left\{(y_1 - y_2)\left(x - \frac{1}{2}(x_2 + x_3)\right) + (x_2 - x_1)\left(y - \frac{1}{2}(y_2 + y_3)\right)\right\}. \end{aligned} \tag{24}$$

Here $\{(x_1,y_1),(x_2,y_2),(x_3,y_3)\}$ are the vertices of the triangles in the physical space, and the area χ^{-1} of the physical element Ω_e is given from the cross product of two of the triangle edge vectors, via the usual formula

$$\chi = 2(x_2y_3 - x_3y_2 + x_3y_1 - x_1y_3 + x_1y_2 - x_2y_1)^{-1}.$$

Then by substitution of (23) and (24), we may easily construct the invertible mapping $\mathbf{S} : J_c^k(\mathbb{R}^2, \Omega) \to J_c^k(\mathbb{R}^2, \mathcal{M})$, such that $\boldsymbol{\varsigma} = \mathbf{S}(\boldsymbol{\varphi})$ represents the Taylor basis in the master element space $\mathcal{M}$. That is, to construct $\mathbf{S}$ explicitly we take the constant first order transformation rules for the derivatives in the base coordinates, given by

$$\partial_x\xi = \chi(y_3 - y_1), \quad \partial_y\xi = \chi(x_1 - x_3), \quad \partial_x\eta = \chi(y_1 - y_2), \quad \partial_y\eta = \chi(x_2 - x_1), \tag{25}$$

in the master element representation $\widehat{\Omega}_{e_i} \in \mathcal{M}$, and

$$\partial_\xi x = (x_2 - x_1)/2, \quad \partial_\xi y = (y_2 - y_1)/2, \quad \partial_\eta x = (x_1 - x_3)/2, \quad \partial_\eta y = (y_1 - y_3)/2 \tag{26}$$

in the physical element representation $\Omega_{e_i} \in \Omega$.

Thus provided the coordinate pair (ξ,η) in the master element representation $\widehat{\Omega}_{e_i} \in \mathcal{M}$ we may use (23) evaluated at the element quadrature points ℓ to fully determine $\mathbf{S}$, where the evaluation at the quadrature points allows for explicit computation of the integral averages in the Taylor basis components (19), or, more explicitly, where we compute:

$$\overline{\left(\frac{(x-x_c)^i}{i!\Delta x^i}\right)\left(\frac{(y-y_c)^j}{j!\Delta y^j}\right)} \approx \frac{1}{|\Omega_e|}\sum_\ell w_\ell \left(\frac{(x_\ell - x_c)^i}{i!\Delta x_\ell^i}\right)\left(\frac{(y_\ell - y_c)^j}{j!\Delta y_\ell^j}\right)|\det \mathbf{J}|$$

for w_ℓ the quadrature weights and the determinant of the Jacobian matrix $\mathbf{J}$ satisfying $|\det \mathbf{J}| = \left|\frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi}\right|$.

All that remains then is to find the coefficient matrix which constructs the change of polynomial basis mapping $\mathbf{N} : \mathbb{R}[\mathcal{M}] \to J_c^k(\mathbb{R}^2, \mathcal{M})$, such that we may write the components of the transformed Taylor basis ς_{ij}, given by terms $T_{ij}\varsigma_{ij}$, with respect to the components of the master element frame Dubiner basis ϕ_{ij}, given by terms $D_{ij}\phi_{ij}$; or such that we recover the matrices

$$\mathbf{T} = \mathbf{N}(\boldsymbol{\phi}), \quad \text{and vice versa} \quad \mathbf{D} = \mathbf{N}^{-1}(\boldsymbol{\varsigma}). \tag{27}$$

But in light of (15) and (18) it follows that for the κ-th component of the m-th size solution vector $\boldsymbol{U}_h$ in $\boldsymbol{\phi}$ we may solve for the Taylor coefficients T_{ij} using the system:

$$\begin{pmatrix} \int_{\widehat{\Omega}_{e_i}} \varsigma_{00} U d\eta d\xi \\ \int_{\widehat{\Omega}_{e_i}} \varsigma_{10} U d\eta d\xi \\ \vdots \\ \int_{\widehat{\Omega}_{e_i}} \varsigma_{cd} U d\eta d\xi \end{pmatrix} = \begin{pmatrix} \int_{\widehat{\Omega}_{e_i}} \varsigma_{00}^2 d\eta d\xi & \int_{\widehat{\Omega}_{e_i}} \varsigma_{00}\varsigma_{10} d\eta d\xi & \cdots & \int_{\widehat{\Omega}_{e_i}} \varsigma_{00}\varsigma_{cd} d\eta d\xi \\ \int_{\widehat{\Omega}_{e_i}} \varsigma_{00}\varsigma_{10} d\eta d\xi & \int_{\widehat{\Omega}_{e_i}} \varsigma_{10}^2 d\eta d\xi & \cdots & \int_{\widehat{\Omega}_{e_i}} \varsigma_{10}\varsigma_{cd} d\eta d\xi \\ \vdots & \vdots & \ddots & \vdots \\ \int_{\widehat{\Omega}_{e_i}} \varsigma_{00}\varsigma_{cd} d\eta d\xi & \int_{\widehat{\Omega}_{e_i}} \varsigma_{10}\varsigma_{cd} d\eta d\xi & \cdots & \int_{\widehat{\Omega}_{e_i}} \varsigma_{cd}^2 d\eta d\xi \end{pmatrix} \begin{pmatrix} T_{00} \\ T_{11} \\ \vdots \\ T_{cd} \end{pmatrix}, \tag{28}$$

for the κ-th component of $\boldsymbol{U}_h$. Note that for the convenience of the reader, we suppress the component index κ here and below, though it should be understood that the slope limiting operations are generally performed componentwise over the elements of the solution state vector.

Now, extending (28) over all the components, the Taylor mass matrix tensor $\mathbf{M}_{\varsigma}$ on the right and the inner product matrix $\mathbf{P}_{\varsigma}$ on the left serve to define the desired transformation:

$$\mathbf{N}(\boldsymbol{\phi}) = \mathbf{M}_{\varsigma}^{-1} \circ \mathbf{P}_{\varsigma}.$$

Its inverse is simply given by forming the Dubiner mass matrix tensor $\mathbf{M}_{\phi}$ and the inner product matrix in $\boldsymbol{\phi}$ denoted $\mathbf{P}_{\phi}$, such that:

$$\mathbf{N}(\boldsymbol{\varsigma})^{-1} = \mathbf{M}_{\phi}^{-1} \circ \mathbf{P}_{\phi}.$$

Then we have fully constructed the invertible Dubiner–Taylor transform $\mathcal{L} : \mathbb{R}[\mathcal{M}] \to J_c^k(\mathbb{R}^2, \Omega)$ as satisfying

$$\mathcal{L}(\boldsymbol{\phi}) = \mathbf{S}^{-1} \circ \mathbf{N} = \mathbf{S}^{-1} \circ \mathbf{M}_{\varsigma}^{-1} \circ \mathbf{P}_{\varsigma} = \mathbf{T} \circ \boldsymbol{\varphi}. \tag{29}$$

with inverse satisfying :

$$\mathcal{L}^{-1}(\boldsymbol{\varphi}) = \mathbf{N}(\boldsymbol{\varsigma})^{-1} \circ \mathbf{S}(\boldsymbol{\varphi}) = \mathbf{M}_{\phi}^{-1} \circ \mathbf{P}_{\phi} \circ \mathbf{S}(\boldsymbol{\varphi}) = \mathbf{D} \circ \boldsymbol{\phi}.$$

3.2. The formal vertex-based hierarchical limiters

We now formally construct the generalized vertex-based slope limiter based off the Barth–Jespersen limiter [25,6]. In this context we define a neighborhood as comprised of those elements that share a common vertex $\boldsymbol{x}_i$, indexed with respect to every vertex of each finite element cell Ω_{e_j}. More clearly, we define the *focal neighborhood* $\Omega_f = \{\Omega_{e_j}\}_i$ (in the sense of the *foci* of geometric optics, as shown in Fig. 2) as the collection of elements such that $\boldsymbol{x}_i \in \Omega_{e_j}$ – where $\{\Omega_{e_j}\}_i$ includes the base element Ω_{e_i} – such that $i = 1, 2, 3$ over triangular elements.

We now note that one must choose a base space in which to implement this slope limiter (e.g. the physical Ω or master $\mathcal{M}$ element spaces, etc.). A fairly common choice (*viz.* [28,25]) is to limit with respect to the full physical space Ω. However, in the context of the local DG formulation this choice is not always so clearly taken. That is, given our transformations from Section 3.1, it is clear that we may not require the full Dubiner–Taylor transform $\mathcal{L}$ but rather have the option to restrict to the master element space $\mathcal{M}$ by simply using the invertible map $\mathbf{N}$. More clearly, since local DG formulations often exploit computational efficiency by working over a master element representation $\mathcal{M}$, we are presented with a choice of composition maps to limit in the master or physical element spaces as shown in Fig. 1, and given either by $\mathbf{N}^{-1} \circ \mathcal{L}_{\mathcal{M}} \circ \mathbf{N}$ over $\mathcal{M}$, or by $\mathcal{L}^{-1} \circ \mathcal{L}_{\Omega} \circ \mathcal{L}$ over Ω. However, since (29) shows that $\mathcal{L}$ requires the extra algorithmic step of transforming back into the physical coordinate frame Ω, in the name of computational efficiency, we clearly prefer the former composition given the context of a relatively standard local DG method. However, when working in a global DG formulation where one elects, for example, a global linear solve, it may be more beneficial to limit with respect to Ω, which as shown in Fig. 1 may also be easily accomplished.

Now, we may define the explicit role of the vertex slope limiter as: a method of finding the *limiter matrix* $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_m)^T$ such that for the solution vector satisfying $\boldsymbol{U}_h = (U_1, \ldots, U_m)^T$, with m the number of unknowns in the system of equations, a vector defined by $\boldsymbol{\alpha} = (\alpha^{(0)}, \ldots, \alpha^{(k)})^T$ for each order derivative $i + j \leqslant k$, the limiter coefficients $\alpha^{(i+j)} \in [0,1]$ allow for a recasting of the renormalized solution in (20) componentwise in the vertex slope limited form with respect to a *focal stencil*, that is $\Omega_{f_i} \subset \Omega_f$ for a fixed vertex $\boldsymbol{x}_i$ (see Fig. 4 for more detail).

In fact, regardless of the initial location containing the state vector (i.e. with respect to $\mathcal{M}$ or with respect to Ω) by simply using our transformations $\mathbf{S}$ and $\mathbf{N}$ from Section 2 we can recast (20) in the master element space $\mathcal{M}$ such that componentwise we have the vertex slope limited approximate solution U^v which satisfies:

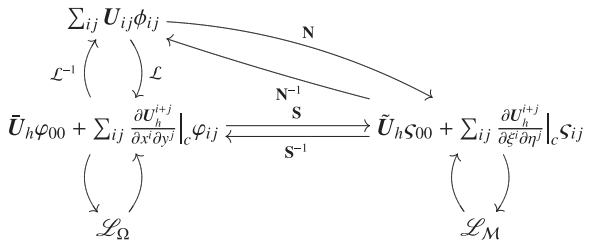

Fig. 1. We look at the maps $\mathbf{N} : \mathbb{R}[\mathcal{M}] \to J_c^k(\mathbb{R}^2, \mathcal{M}), \mathbf{S} : J_c^k(\mathbb{R}^2, \mathcal{P}) \to J_c^k(\mathbb{R}^2, \mathcal{M})$, and $\mathcal{L} : \mathbb{R}[\mathcal{M}] \to J_c^k(\mathbb{R}^2, \Omega)$, where $\mathcal{L}_{\Omega}$ and $\mathcal{L}_{\mathcal{M}}$ are the abstract operators that limit in either the physical element space Ω or the master element space $\mathcal{M}$.

$$U^{\mathrm{v}} = \overline{U}\varsigma_{00} + \sum_{0<i+j\leqslant k} \alpha^{(i+j)}\varsigma_{ij}\left(\frac{\partial^{i+j}U}{\partial\xi^i\partial\eta^j}\right)\Big|_c, \tag{30}$$

where $\overline{U}$ and U correspond to the approximate solution vector $\boldsymbol{U}_h$ transformed to the master element frame in the Taylor basis representation.

Now, notice that above there exists only one $\alpha^{(i+j)}$ for each top k-th order mixed derivative in ξ and η. In order to recover the $\alpha^{(i+j)}$'s in the polynomial basis expansion, we must decompose our solution Taylor expansion into mixed order linear reconstructions. To do this, we first order our Taylor polynomial into a hierarchical basis such that each monomial index $b = b(i,j)$ is provided using the lexicographic ordering with ordered lattice pairs (i,j) given by the sequence $(0,0) < (0,1) < (1,0) < (0,2) < (1,1) < \cdots = (i,j)$ corresponding to indices b, respectively; that is by the sequence $(1) < (2) < (3) < (4) < (5) < \cdots < (b) \cdots < (s)$ in the Taylor expansion. In fact, the monomial index in the hierarchy may be determined by the diophantine equation:

$$b = \frac{j}{2}(j+1) + ij + \frac{i}{2}(i+3) + 1. \tag{31}$$

Then we generate the hierarchical triangular sequence $s = s(p)$, where $p = p(i,j)$ satisfies $p = (i+j)$, such that s determines the upper bound on the degrees of freedom in the polynomial expansion,

$$s = \frac{1}{2}(p+1)(p+2), \quad \text{given inverse } g = g(s) \text{ such that } \quad g = \left\lfloor \frac{1}{2} + \sqrt{2s} \right\rfloor, \tag{32}$$

where $\lfloor\cdot\rfloor$ is the usual floor function. Note that in particular we may use $g = g(s)$ or $g = g(b)$ for $g(b) \in \mathfrak{l}$ corresponding to *level* $\mathfrak{l} \neq \mathfrak{l}_{top}$ (defined below) since by virtue of the mapping (32) both return the same value.

Then letting $U^{e_j}_{i,c,b}$ be the value of the b-th term in the polynomial basis of $\boldsymbol{U}_h$ at the centroid c of element $\widehat{\Omega}_{e_j}$ containing $\boldsymbol{x}_i = (\xi_i, \eta_i)$ in the master element representation, we define the maximum $U^{\max}_{i,b}$ and minimum $U^{\min}_{i,b}$ values for each unknown monomial at $\boldsymbol{x}_i$ over the *focal stencil* Ω_{f_i} situated with respect to the master element frame $\widehat{\Omega}_{f_i}$ as

$$U^{\max}_{i,b} = \max_{\widehat{\Omega}_{e_j}\in\widehat{\Omega}_{f_i}} \left\{U^{e_j}_{i,b,c}\right\} \quad \text{and} \quad U^{\min}_{i,b} = \min_{\widehat{\Omega}_{e_j}\in\widehat{\Omega}_{f_i}} \left\{U^{e_j}_{i,b,c}\right\}. \tag{33}$$

Now, we are able to define the $(i+j)$-th linear reconstructions $U^{(i+j)}_{b,i}$ over the vertices $\boldsymbol{x}_i$ of any element by taking derivations with respect to the monomial coefficients of (30). That is, the linear perturbation of the constant term is constructed such that,

$$U^{(1)}_{b,i} = \overline{U} + \frac{\partial U_i}{\partial\xi}\Big|_c(\xi_i - \xi_c) + \frac{\partial U_i}{\partial\eta}\Big|_c(\eta_i - \eta_c), \quad \text{for } s = 3. \tag{34}$$

Moreover, it is now direct to construct the higher order terms whereby setting

$$\mathcal{C}_b = \left(\frac{\partial^{i+j}U}{\partial\xi^i\partial\eta^j}\right)\Big|_c \quad \text{for } b(i,j) > 1,$$

such that for any mixed derivative order in the hierarchical basis – as a property of the lexicographic ordering – we can write:

$$U^{(i+j)}_{b,i} = \mathcal{C}_b + \mathcal{C}_{b+g}(\eta_i - \eta_c) + \mathcal{C}_{b+g+1}(\xi_i - \xi_c), \tag{35}$$

for any polynomial order k. Proceeding, we can now define the correction factors $\alpha^{(i+j)}_b$ for each element Ω_{e_l}, where the vertex-based condition is simply defined as

$$\alpha^{(i+j)}_b = \min_{\boldsymbol{x}_i\in\widehat{\Omega}_{e_l}} \begin{cases} \min\left\{1, \left(\frac{U^{\max}_{i,b} - U^{e_l}_{i,c,b}}{U^{(i+j)}_{b,i} - U^{e_l}_{i,c,b}}\right)\right\}, & \text{for } U^{(i+j)}_{b,i} > U^{e_l}_{i,c,b} \\ 1, & \text{for } U^{(i+j)}_{b,i} = U^{e_l}_{i,c,b} \\ \min\left\{1, \left(\frac{U^{\min}_{i,b} - U^{e_l}_{i,c,b}}{U^{(i+j)}_{b,i} - U^{e_l}_{i,c,b}}\right)\right\}, & \text{for } U^{(i+j)}_{b,i} < U^{e_l}_{i,c,b} \end{cases} \tag{36}$$

which, again, is determined separately for each monomial represented in the master frame.

These $\alpha^{(i+j)}_b$ determine a set of limiting constraints for every hierarchical monomial in the Taylor expansion, but as in [25], we minimize over derivatives of similar top order, such that we recover the components:

$$\alpha^{(i+j)}_{\mathfrak{l}(p)} = \min_{g(b)=\mathfrak{l}(p_0)} \alpha^{(i+j)}_b. \tag{37}$$

Notice that these limiting coefficients span the *level* $\mathfrak{l}(p_0)$, not the hierarchical index b corresponding to *level* $\mathfrak{l}(p)$ (where p and p_0 are fully explained below). That is, in the hierarchical basis the linear reconstructions from the perturbation at the *level* below (i.e. *level* $(\mathfrak{l}-1)$) are what effectively determine the limiting coefficient at *level* $\mathfrak{l}$ (e.g. the gradient terms). More

precisely, the *level* $\mathfrak{l} = \mathfrak{l}(p_0)$ is determined with respect to the sequence of integers starting at $p_0(p_0-1)/2+1$ and increasing by one until reaching $p_0(p_0+1)/2$. Then the *level* is defined by $\mathfrak{l} = \sup\{g(p_0(p_0-1)/2+1),\ldots,g(p_0(p_0+1)/2)\}$, where $p_0 = 1$ for the strictly linear case, and in general is a positive integer such that $p_0 \leqslant p$ and is fully determined by $g(b(i,j))$. In general however, the *level* $\mathfrak{l}(p)$ spans $\mathfrak{l} = \sup\{g(p(p+1)/2)+1,\ldots,g((p+1)(p+2)/2)\}$ such that the *level* below $\mathfrak{l}(p_0)$ simply corresponds to setting $p = p_0 - 1$.

Finally we limit the magnitude of the correction by the maximum value of every correction factor of greater than or equal order. In other words, we do not allow a higher order correction to demonstrate greater regularity than a lower order correction, and in fact empirical experimentation has found this to be a necessary constraint. That is, setting $q = (i+j)$ and $r = (i'+j')$ for i' and j' indices, then we determine an upper bound on the correction parameter by resetting:

$$\alpha^{(q)} := \max_{q\leqslant r,\mathfrak{l}\leqslant\mathfrak{l}_{\text{top}}} \alpha_{\mathfrak{l}}^{(r)}, \quad \forall q \geqslant 1,\ \forall r \geqslant q. \tag{38}$$

The top *level* $\mathfrak{l}_{\text{top}}$ simply corresponds to the *level* whose upper bound is determined by $g(s) = g(b)$. Also notice that the derivative order $(i+j)$ is fundamentally coupled to the *level* $\mathfrak{l}$, and so is in some ways redundant notation which we have used in order to emphasize this coupling.

It is also worth noting, that as a consequence of the above construction we are now easily able to implement an arbitrarily higher-order extension of the Barth–Jespersen limiter [25,6], where we may perform the exact steps as above, but simply exchange (33) with

$$U_{i,b}^{\max} = \max_{\widehat{\Omega}_{e_j}\in\widehat{\Omega}_{E_i}} \left\{U_{i,b,c}^{e_j}\right\} \quad \text{and} \quad U_{i,b}^{\min} = \min_{\widehat{\Omega}_{e_j}\in\widehat{\Omega}_{E_i}} \left\{U_{i,b,c}^{e_j}\right\}, \tag{39}$$

where $\widehat{\Omega}_{E_i}$ is the *edge stencil* of $\widehat{\Omega}_{e_j}$ at $\boldsymbol{x}_i$ in the master element representation – or the corresponding set of those physical elements sharing an edge with Ω_{e_j} at vertex $\boldsymbol{x}_i$ such that the base element $\Omega_{e_j} \in \Omega_{E_i}$ (see Fig. 2).

A schematic is provided in Fig. 3 which is meant to simplify the notation and unify the basic principles underlying both the vertex and Barth–Jespersen limiters (as well as the adapted vertex-based limiters of Section 3.2.1).

3.2.1. On adapted vertex-based limiters

Both the vertex limiter and the Barth–Jespersen limiter from Section 3.2 demonstrate a similar – though often times non-ideal – behavior. That is, notice that in both the definition of (33) and 39) that we have found a maximum or minimum with respect to a local neighborhood of the mesh. Hence, in either case, when we compute the limiting coefficients in (36) a local bound (e.g. (39)) is always achieved, even in the degenerate case of when $U_{i,b}^{\min} = U_{i,b}^{\max}$.

As a consequence of this, the numerator in the quotients of (36) vanish on elements admitting a local extremum, leading to persistent and excessive diffusivity (i.e. limiting $\alpha = 0$ at each such timestep) arising at all orders in each local extrema of the mesh, even when those extrema are neither spurious nor potentially unstable; and moreover, this behavior compounds

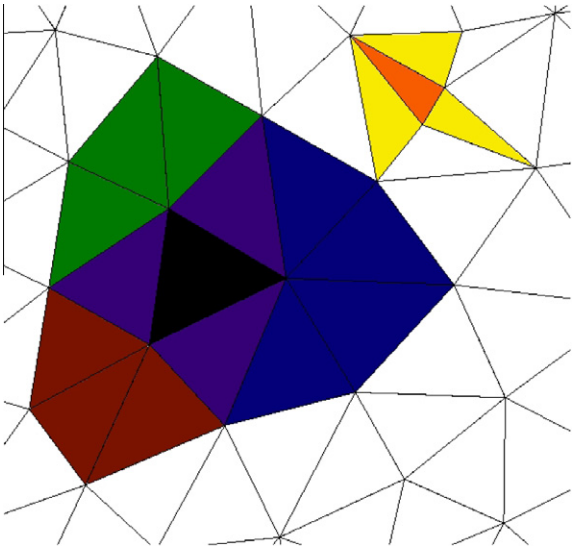

Fig. 2. Here we show the *focal neighborhood* Ω_f of a base element Ω_{e_i} filled in black. Green, red and blue are the three *focal neighborhood* groups based at vertices $\boldsymbol{x}_i$ of the black base cell, while purple are cells contained in more than one of the two *focal neighbor stencils* (incidentally comprising the *edge neighborhoood* of Ω_{e_i}). In a contrasting geometric locale, the orange base cell's *edge neighbors* Ω_{E_j} are each filled in yellow, comprising the *edge neighborhood* Ω_E. See Fig. 4 for details. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

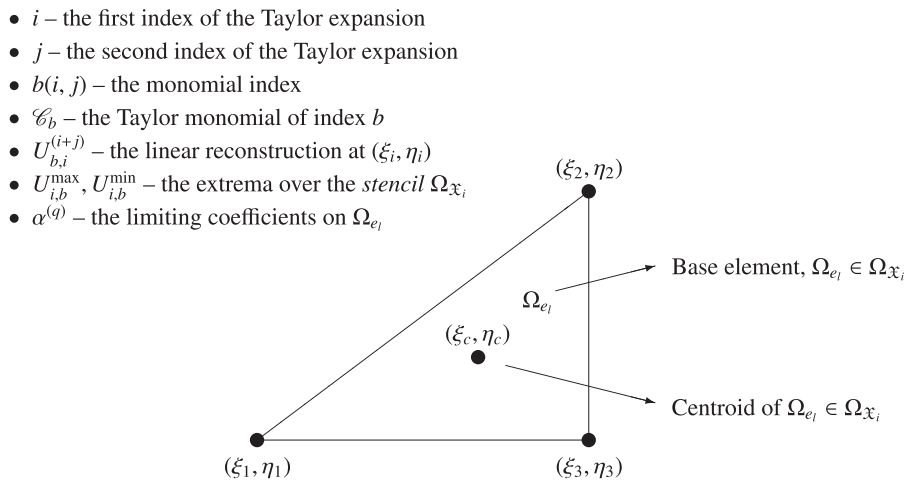

Fig. 3. Here we provide a key for the vertex and Barth–Jespersen limiters of Sections 3.2 and 3.2.1. Generally theses limiting procedures depend on the chosen *stencil* $\Omega_{\mathfrak{X}_i}$ and a local linear reconstruction of the solution in order to develop the limiting coefficients from (30).

in p since as p increases the number of degrees of freedom (i.e. monomials) in the solution which have local extrema also increases nonlinearly.

This behavior over values of local extrema can become quite dominant depending on the mesh geometry. In particular, since the vertex-based limiter has a larger local neighborhood (i.e. the *focal neighborhood*) than the Barth–Jespersen limiter, in principle it should provide more information from which to glean a more accurate approximate local reconstruction. However, due to this "diffusivity," the larger local neighborhood actually lends itself towards increasing the nonlocality of the diffusive effects of the neighborhood-wise extrema as $p \nearrow p_{\max}$, and hence in practice can actually precipitate greater diffusion in the vertex limiter than the native Barth–Jespersen limiter as p increases (up to the mesh geometry).

In order to reduce this so-called "blind diffusion" in both limiters we introduce a simple functional which attempts to treat a portion of this special case separately. That is, we simply replace (36) with:

$$\alpha_b^{(i+j)} = \min_{\mathbf{x}_i \in \Omega_{e_l}} \begin{cases} \min\left\{1, \left(\frac{U_{i,b}^{\max} - U_{i,c,b}^{e_l}}{U_{b,i}^{(i+j)} - U_{i,c,b}^{e_l}}\right)\right\}, & \text{for } U_{b,i}^{(i+j)} > U_{i,c,b}^{e_l} \\ \min\left\{f_{\max}, \left|\frac{U_{i,b}^{\max} - U_{i,b}^{\min}}{U_{b,i}^{(i+j)} - U_{i,c,b}^{e_l}}\right|\right\}, & \text{for } U_{i,c,b}^{e_l} = U_{i,b}^{\max} \\ 1, & \text{for } U_{b,i}^{(i+j)} = U_{i,c,b}^{e_l} \\ \min\left\{f_{\min}, \left|\frac{U_{i,b}^{\min} - U_{i,b}^{\max}}{U_{b,i}^{(i+j)} - U_{i,c,b}^{e_l}}\right|\right\}, & \text{for } U_{i,c,b}^{e_l} = U_{i,b}^{\min} \\ \min\left\{1, \left(\frac{U_{i,b}^{\min} - U_{i,c,b}^{e_l}}{U_{b,i}^{(i+j)} - U_{i,c,b}^{e_l}}\right)\right\}, & \text{for } U_{b,i}^{(i+j)} < U_{i,c,b}^{e_l} \end{cases} \tag{40}$$

where $f_{\max}, f_{\min} \in (0,1)$ are constants used to limit the rate at which the extrema diffuse (that is, reduce the rate at which error is introduced into the solution), and when $f_{\max} = f_{\min}$ we denote them by f_d.

We find when setting $f_d = 1$ we generally get a very moderate improvement in the limiting error behavior of both the vertex and Barth–Jespersen limiters. Nevertheless, clearly (40) has only accounted partially for the degenerate local extrema cases, in particular it still fails to properly account for the case of $U_{b,i}^{\min} = U_{b,i}^{\max}$, and the absolute value is used to account for the fact that the signs have not been separately controlled. We have developed strategies for adopting fixes for these issues into the limiter, but in general find even the augmented regimes to still demonstrate substantially more diffuse behavior than the restricted regime presented in Section 3.4, and so will suppress any further comment on the subject at present, simply noting that it is possible to improve upon the basic behavior of the limiter in p by developing selection strategies to deal with the many special cases which arise over solutions locally, and where alternatively one is often also able to improve the error behavior by tuning $f_{\max}$ and $f_{\min}$.

It should be additionally noted here that in [25] a mass lumping strategy is implemented with respect to the triangular meshes in order to prevent the formation of undershoots and overshoots caused in the presence of the non-orthogonal Taylor mass matrix. It was also demonstrated in [25] that this strategy can have a measurably beneficial effect on the error behavior for $p \leqslant 2$, and is thus clearly worth further examination at higher p. We will return to this issue briefly in Section 4 as a note of comparison between the implementational strategies.

3.3. The hierarchical reconstruction via MUSCL or ENO

We now consider the hierarchical reconstruction scheme presented in [1,27]. Formally in this setting we simply take derivatives of (16) in the master element frame, and work locally over the averages and differences of these differential reconstructions. The method is presented as a two step process, where we start in step 1 at the highest order derivatives and work down to the lowest, with the caveat that the linear and constant terms are dealt with separately in step 2.

Step 1. Starting at the top order coefficient k, a linearization of the $(k-1)$-st derivative of (16) is given by (35) in the Taylor basis for $i+j=k-1$. Here, however, we recover the entire higher order component including the nonlinear terms, so that we must employ our monomial index function $b(i,j)$ given in (31). That is, beginning at the top *level* $\mathrm{I}(k)$ for $i+j=k$ we define the linear part as satisfying:

$$\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}} := \mathcal{C}_{b(i,j)}, \quad \forall b \in \mathrm{I}(k) \ \wedge \ \forall \Omega_{e_l} \in \Omega_{\mathfrak{x}}, \tag{41}$$

where $\mathfrak{x}$ here and below may be f or E (i.e. the *focal* and the *edge neighborhoods*, respectively, as shown in both Figs. 2 and 4), and where here and below $\wedge$ is the logical conjunction operator and $\vee$ is the corresponding logical disjunction operator (see Figs. 5–7).

At the lower (nonlinear) *levels* (i.e. the *levels* I such that $\mathrm{I}(1) < \mathrm{I} < \mathrm{I}(k)$) by expansion – after recovering the i and j indices of the base b-th component – then setting $\tilde{b} = b(i+i', j+j')$ and integrating locally over each cell in the neighborhood, we have that:

$$\overline{U}^{(i+j)}_{b,\Omega_{e_l}} = \mathcal{C}_{b(i,j)} + \Omega^{-1}_{e_l} \int_{\Omega_{e_l}} \left\{ \sum_{i'+j'>0} \frac{1}{i'!j'!} \mathcal{C}_{\tilde{b}} (\eta - \eta_c)^{i'} (\xi - \xi_c)^{j'} \right\} d\eta d\xi, \quad \forall \tilde{b} \leqslant s \ \wedge \ \forall \Omega_{e_l} \in \Omega_{\mathfrak{x}}. \tag{42}$$

Likewise for each *level* I we integrate the higher order perturbative terms such that:

$$\overline{U}^{(i+j)}_{b_{slope},\Omega_{e_l}} = \Omega^{-1}_{e_l} \int_{\Omega_{e_l}} \left\{ \sum_{i'+j'>0} \frac{1}{i'!j'!} \mathcal{C}_{\tilde{b}} (\eta - \eta_c)^{i'} (\xi - \xi_c)^{j'} \right\} d\eta d\xi, \quad \forall \tilde{b} \leqslant s \ \wedge \ \forall \Omega_{e_l} \in \Omega_{\mathfrak{x}}. \tag{43}$$

It is then these two averages which serve to limit the *level* I components of the Taylor basis by way of the linear type average $\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}$ of the difference of (42) with (43) in each b:

$$\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}} := \left(\overline{U}^{(i+j)}_{b,\Omega_{e_l}} - \overline{U}^{(i+j)}_{b_{slope},\Omega_{e_l}} \right), \quad \forall \Omega_{e_l} \in \Omega_{\mathfrak{x}}. \tag{44}$$

Now the linear terms of (44) will be used to determine the *candidates* for the updated values of the base cell Ω_{base}; which is to say that the $(k-1)$-st component of the k-th order jet $(J^k_c \boldsymbol{\alpha} \boldsymbol{U}_h)(\varsigma_{ij})$ is limited by filtering a set of *candidates* through a family of

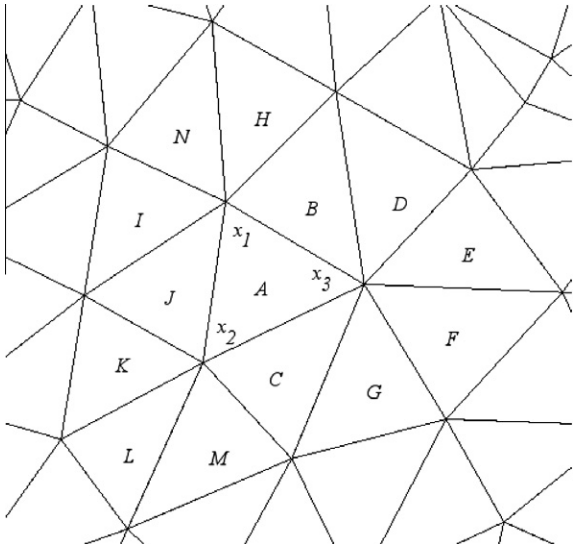

Fig. 4. Here we show the *focal stencil* Ω_{f_j} and the *edge stencil* Ω_{E_j} of a base element $\Omega_{e_j} = A$. The stencils are defined with respect to the base elements vertices $\boldsymbol{x}_i$ for $i = 1, 2, 3$, such that the *focal stencil* at $\boldsymbol{x}_1$ is $\Omega_{f_1} = \{J, I, N, H, B, A\}$, and likewise $\Omega_{f_2} = \{J, K, L, M, C, A\}$ and $\Omega_{f_3} = \{C, G, F, E, D, B, A\}$. Similarly the *edge stencils* are given by: $\Omega_{E_1} = \{J, B, A\}$, $\Omega_{E_2} = \{J, C, A\}$ and $\Omega_{E_3} = \{B, C, A\}$. Notice that the union of sets recovers the *focal neighborhood* ($\Omega_f = \cup_i \Omega_{f_i}$) and the *edge neighborhood* ($\Omega_E = \cup_i \Omega_{E_i}$), while the restriction of the symmetric difference of sets defines the *focal neighborhood group* ($\ominus_i \Omega_{f_i}|_{\Omega_{f_j}}$) and *edge neighborhood group* ($\ominus_i \Omega_{E_i}|_{\Omega_{E_j}}$) for any vertex j.

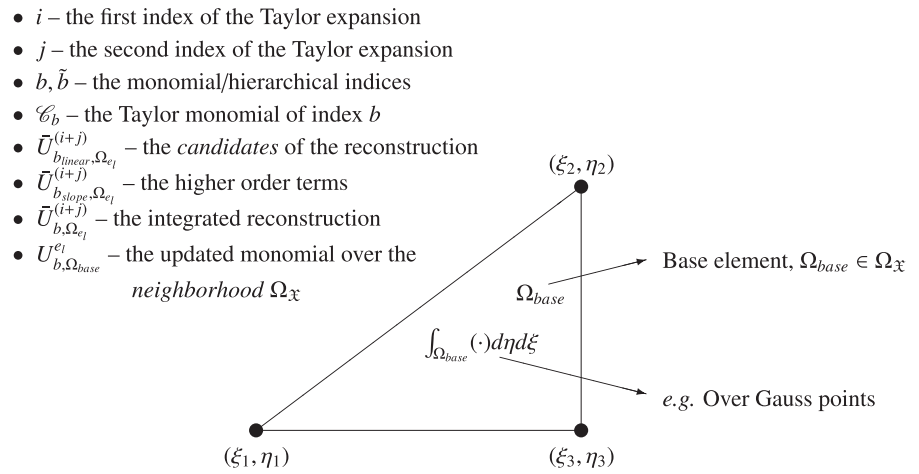

Fig. 5. Here we provide a key for the hierarchical reconstruction method developed in Section 3.3. The limiting procedure depends on the entire *neighborhood* Ω_x, the fully integrated solution, and a choice of minmod functions in order to reconstruct the limited form of the monomial coefficients on the base cell.

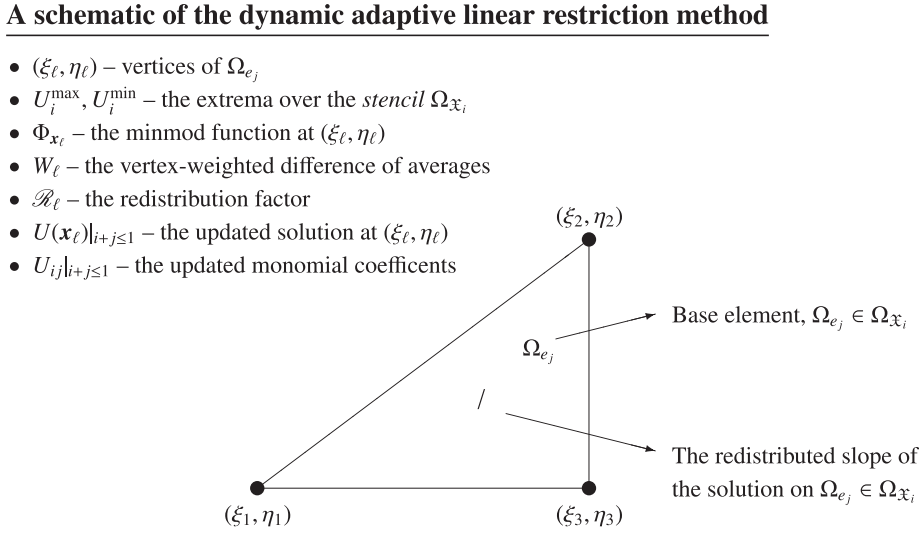

Fig. 6. Here we provide a key for the adaptive linear restriction method from Section 3.4. Again this limiting procedure depends on the *stencil* $\Omega_{\mathfrak{X}_i}$, the linear part of the full solution of order p, and on a redistribution strategy that can be thought of heuristically as depending on a "consistent redistribution of the slopes of the linear coefficients."

minmod functions, such that:

$$U^{e_l}_{b,\Omega_{base}} := \text{minmod}^{*}_{\forall \Omega_{e_l}\in\Omega_{\mathfrak{X}}}\left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right). \tag{45}$$

Notice that we may also choose to find candidates over restricted subsets of the full neighborhood $\Omega_{\mathfrak{X}}$ in order to try and more effectively *localize* our limiting. For example, we may choose to find the minmod function over the local *stencil* $\Omega_{\mathfrak{X}_i}$ centered about a vertex of the cell and then perform a different selection rule over that set of candidates; or, alternatively, we may compute the integral averages over the local *stencil* $\Omega_{\mathfrak{X}_i}$ in (41)–(43) and then perform a minmod with respect to the full neighborhood $\Omega_{\mathfrak{X}}$. We have implemented and tested a number of these different regimes, and consider each of them in this paper to live under the general heading of "hierarchical reconstruction schemes," though for the sake of brevity we focus only on (45) below.

Note that we perform Step 1 for each *level* $\mathfrak{l}(\jmath)$ where $\jmath < k$, and recursing down to the *level* corresponding to the $\mathfrak{l}$ associated to the quadratic components at $p = 2$; where first we limit the difference (44) across the *neighborhood* of a base element in

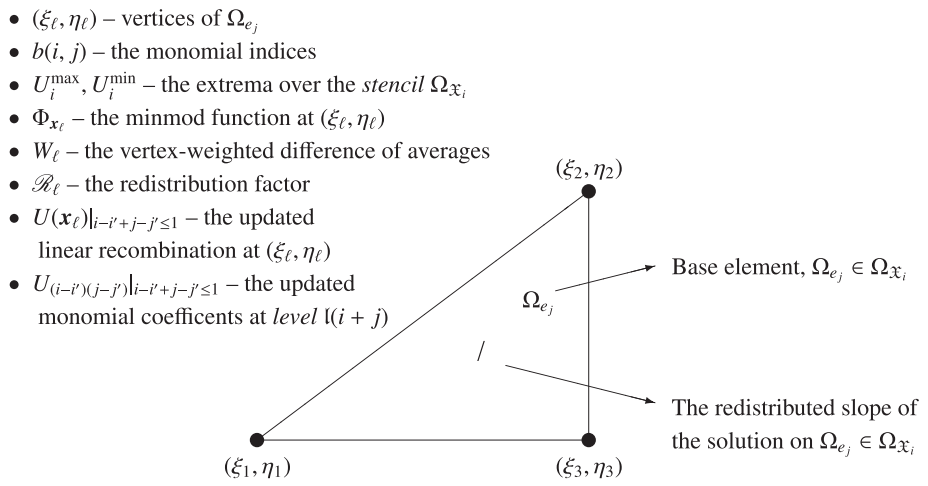

Fig. 7. Here we provide a key for the hierarchic linear recombination method of Section 3.5. This procedure depends on the chosen *stencil* $\Omega_{\mathfrak{X}_i}$, a collection of linear recombinations of restricted subsets of monomial coefficients from the total solution, and an application of the method developed in Section 3.4 to these linear recombinations in order to recover the limited solution.

order to reconstruct the values on the base cell proper. For these purposes, we employ the following set of $\text{minmod}^*_{\mathfrak{X}} = \Phi^*_{\mathfrak{X}}$ functions. The MUSCL reconstruction method relies on the function:

$$\Phi^{\text{m}}_{\mathfrak{X}}\left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right) = \begin{cases} \min\limits_i \left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right), & \text{if} \quad \overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}} > 0 \quad \forall \Omega_{e_l} \in \Omega_{\mathfrak{X}}, \\ \max\limits_i \left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right), & \text{if} \quad \overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}} < 0 \quad \forall \Omega_{e_l} \in \Omega_{\mathfrak{X}}, \\ 0, & \text{otherwise}, \end{cases}$$

while the ENO reconstruction is given by

$$\Phi^{\text{e}}_{\mathfrak{X}}\left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right) = \overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}} \quad \text{if} \quad \overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}} = \min_{\forall \Omega_{e_l} \in \Omega_{\mathfrak{X}}} \left|\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right|.$$

Additionally, following [27], the $\text{minmod}^*_{\mathfrak{X}}$ function may be set as a center bias scheme given by

$$\Phi^{\text{c}}_{\mathfrak{X}} = \Phi^{\text{m}}_{\mathfrak{X}}\left((1+\epsilon)\cdot \Phi^{\text{m}}_{\mathfrak{X}}\left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right), \frac{1}{r}\sum_{k=1}^{r} \overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_k}}\right), \tag{46}$$

or the weighted ENO scheme,

$$\Phi^{\text{e}_2}_{\mathfrak{X}} = \Phi^{\text{e}}_{\mathfrak{X}}\left((1+\epsilon)\cdot \Phi^{\text{e}}_{\Omega_{\mathfrak{X}}}\left(\overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_l}}\right), \frac{1}{r}\sum_{k=1}^{r} \overline{U}^{(i+j)}_{b_{\text{linear}},\Omega_{e_k}}\right), \tag{47}$$

where in either case r is the total number of neighboring cells of the base cell $\Omega_{e_{base}}$, ϵ is a user defined constant, and $(\cdot)$ in both (46) and (47) is merely standard multiplication. It is known that setting ϵ large helps to achieve the expected order of accuracy over triangular meshes.

Step 2. Now we address the case of how to limit the solution with respect to the linear $i+j=1$ and constant $i=j=0$ cases. For the linear case, we simply choose to limit with respect to a subset of limiting regimes, including those in Sections 3.2, 3.3, 3.4 and 3.5. We choose this, in particular, in order to electively replace the MUSCL and ENO schemes from Step 1, which are relatively speaking more diffuse in our experiments at *level* $\mathfrak{l}(1)$ than some other possible alternatives.

Finally, the constant terms at *level* $\mathfrak{l}(0)$ are simply set equal to the average value on their base cell, $\overline{U}_{\zeta_{00}}|_{\Omega_{base}}$ in order to enforce invariance of the cell averages. In other words, the constant terms remain unchanged.

We should also note that recent improvements have been made in the context of hierarchical reconstructuion tech-

niques. In particular, recent work has been done to extend the formalism above to include WENO-type linear reconstructions. That is, the WENO-type formalism of [44,35] has been extended to the context of hierarchical reconstruction based limiter regimes in [43] specifically in order to address the fact that the MUSCL and ENO type approaches have been shown to often fail to give the desired order of accuracy on triangular meshes. These techniques rely on the conditioning of a local ($\Omega_{\mathfrak{x}}$-restricted) recontruction matrix, and are beyond the scope of the present paper. We direct the interested reader to [43].

3.4. On a dynamically adaptive linear restriction

In this subsection we generalize and update a version of the BDS limiter that had its foundations initially seeded in [7] for linear polynomials over uniform structured meshes. We present the formal construction of a substantially more general form of this limiter, to act over an arbitrary order p basis by way of a linear restriction technique over unstructured triangular meshes. This limiter is developed with an eye towards p-enrichment schemes, and in particular hp-adaptive schemes, where in areas of high (jump) variability one generally wants to reduce the order of p while refining the mesh parameter h. In this section we first restrict back to the Dubiner basis $\phi_{ij} \in \mathbb{R}[\mathcal{M}]$, in part to compare to the same implementation carried out in the Taylor basis as a consequence of the formulation presented in Section 3.5, which for linears turns out to be equivalent.

Let us first restrict to the sub-quadratic terms of the basis for any order p, such that we are only concerned initially with the terms corresponding to $i+j \leqslant 1$. Then, similar to (33), setting $U_i^{e_j}$ as the constant piece of the Dubiner basis in $\boldsymbol{U}_h$ of the base element $\widehat{\Omega}_{e_j}$ containing $\boldsymbol{x}_i = (\xi_i, \eta_i)$ in the master element representation, we define the maximum $U_i^{\max}$ and minimum $U_i^{\min}$ values for each unknown at every $\boldsymbol{x}_i \in \widehat{\Omega}_{e_j}$ over the chosen *stencil* $\widehat{\Omega}_{\mathfrak{x}_i}$ as

$$U_i^{\max} = \max_{\forall \widehat{\Omega}_{e_j} \in \widehat{\Omega}_{\mathfrak{x}_i}} \left\{ U_i^{e_j} \right\} \quad \text{and} \quad U_i^{\min} = \min_{\forall \widehat{\Omega}_{e_j} \in \widehat{\Omega}_{\mathfrak{x}_i}} \left\{ U_i^{e_j} \right\}. \tag{48}$$

Next we take the full approximate solution restricted to its sub-quadratic part and evaluated at the three vertices of the cell, denoted by the three values $U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1}$ for $\ell = 1, 2, 3$ corresponding to the vertices, while $i+j$ corresponds to the polynomial order. Then at each vertex we employ the following minmod function $\Phi_{\boldsymbol{x}_\ell} = \Phi_{\boldsymbol{x}_\ell}(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1})$:

$$\Phi_{\boldsymbol{x}_\ell} = \max\left\{ \min\left\{ (U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1}, U_\ell^{\max} \right\}, U_\ell^{\min} \right\}, \tag{49}$$

where we subsequently reset the vertex value to $U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} := \Phi_{\boldsymbol{x}_\ell}(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1})$.

Proceeding, we estimate the average vertex value over the *stencil* to its value on the minmod'ed *neighborhood* by computing, $\mathrm{Avg}_\ell(U(\boldsymbol{x}_\ell)|_{i+\leqslant 1}) = \frac{1}{3}\sum_\ell U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1}$, and then we calculate a vertex-weighted difference between this average and $U_\ell^{e_j}$, which is given by:

$$\mathrm{W}_\ell = 3\left(\mathrm{Avg}_\ell(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1}) - U_\ell^{e_j} \right). \tag{50}$$

The restricted difference functions $\mathfrak{D}_\ell$ are then given with respect to each vertex $\boldsymbol{x}_\ell$,

$$\mathfrak{D}_\ell = \left(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} - U_\ell^{e_j} \right) \mathrm{sgn}\mathrm{W}_\ell \tag{51}$$

where $\mathrm{sgn}(\cdot)$ is the usual signum function except that $\mathrm{sgn}(0) := 1$. Then, if $\mathfrak{D}_\ell$ is positive, which means that either both the average and the approximate solution at the vertex are each larger than $U_\ell^{e_j}$, or similarly that they are both smaller than $U_\ell^{e_j}$, then we set:

$$\mathcal{D} = \max\left(1, \sum_{m=0}^{I} 1 \right), \quad \text{where } I = \sum_\ell \mathrm{sgn}\mathfrak{D}_\ell, \quad \text{for each } \boldsymbol{x}_\ell \text{ restricted such that } \mathfrak{D}_\ell > 0. \tag{52}$$

This allows us now to generate a vertex-wise redistribution factor $\mathcal{R}_\ell$ over each element, defined simply by setting

$$\mathcal{R}_\ell = \begin{cases} (W_\ell \mathrm{sgn}\, W_\ell)/\mathcal{D}, & \text{if } \mathfrak{D}_\ell > 0, \\ 0, & \text{otherwise}, \end{cases} \tag{53}$$

where the maximum allowed value $\mathcal{R}_\ell^{\max}$ is determined by:

$$\mathcal{R}_\ell^{\max} = \begin{cases} \left(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} - U_\ell^{\min} \right) & \text{if } \mathrm{sgn}W_\ell > 0, \\ \left(U_\ell^{\max} - U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} \right) & \text{otherwise}. \end{cases} \tag{54}$$

The approximate values at the vertices are then updated, where we make sure the maximum redistribution amount is not exceeded, $\mathcal{R}_\ell = \min(\mathcal{R}_\ell, \mathcal{R}_\ell^{\max})$. The redistributed vertex value is updated explicitly to satisfy:

$$U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} := U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} - \mathcal{R}_\ell \mathrm{sgn}W_\ell. \tag{55}$$

As an optional step, we add the ability to adapt our limiter to sense areas where substantial overshoots and/or undershoots have occurred, thus marking the presence of potential shock fronts. We check back to determine that if the redistribution at a specific vertex passes a given tolerance $\varepsilon \in \mathbb{R}^+$, then we either zero out the higher order terms if in a fixed order p solution, or we lower our polynomial order from p to p_{lim} (where p_{lim} may be $p-1$ or p_{min}, etc.) if in a p-adaptive context (which will be fully addressed in Section 5). That is, we define a restriction function $\Re = \Re(\mathcal{P}^k, U|_{i+j>1})$ that operates either on the restricted solution $U|_{i+j>1}$ or the local polynomial order $\mathcal{P}^k(\Omega_{e_l})$ over the entire cell:

$$\Re = \begin{cases} U|_{i+j>1} = 0 & \text{if } \left(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} - \Phi_{\boldsymbol{x}_\ell}| \leqslant \varepsilon\right) \wedge \mathcal{P}^{i+j>1}(\Omega_{e_l}), \\ \mathcal{P}^k(\Omega_{e_l}) \to \mathcal{P}^{k-1}(\Omega_{e_l}) & \text{if } \left(U(\boldsymbol{x}_\ell)|_{i+j\leqslant 1} - \Phi_{\boldsymbol{x}_\ell}| \leqslant \varepsilon\right) \wedge (p-\text{adaptive}) \wedge \mathcal{P}^{i+j>1}(\Omega_{e_l}) \end{cases}$$

if any of the vertex values exceed the tolerance. We also note that clearly ε should have an implicit dependence on h.

Finally we make sure that the difference is properly re-weighted for the next computation at the elements next vertex (if one exists) by determining the amount available to redistribute by computing: $W_\ell := (W_\ell - \mathcal{R}_\ell \text{sgn} W_\ell)$. This proceeds until no vertices are left to evaluate in the cell.

Thus we arrive with the sub-quadratic approximate solution, but what we need are the coefficients on ϕ_{10} and ϕ_{01} in the basis. To get these we must simply invert the following local constant matrix:

$$\begin{pmatrix} \phi_{00}(\boldsymbol{x}_1) & \phi_{10}(\boldsymbol{x}_2) & \phi_{01}(\boldsymbol{x}_3) \\ \phi_{00}(\boldsymbol{x}_1) & \phi_{10}(\boldsymbol{x}_2) & \phi_{01}(\boldsymbol{x}_3) \\ \phi_{00}(\boldsymbol{x}_1) & \phi_{10}(\boldsymbol{x}_2) & \phi_{01}(\boldsymbol{x}_3) \end{pmatrix} \begin{pmatrix} U_{00} \\ U_{10} \\ U_{01} \end{pmatrix} = \begin{pmatrix} U(\boldsymbol{x}_1)|_{i+j\leqslant 1} \\ U(\boldsymbol{x}_2)|_{i+j\leqslant 1} \\ U(\boldsymbol{x}_3)|_{i+j\leqslant 1} \end{pmatrix}, \tag{56}$$

which provides the unknowns.

3.5. The hierarchic linear recombination

Now, we develop a new slope limiting strategy based on the limiter presented in Section 3.4, but transformed into the Taylor basis $\varsigma_{ij} \in J^k_c(\mathbb{R}^2, \mathcal{M})$, and generalized over linear recombinations of linear reconstructions.

More clearly, we take our transformed solutions (30) such that in the Taylor basis we can extract the hierarchical basis at any *level* $\mathfrak{l}$, independently of cell vertices $\boldsymbol{x}_i$, by simply extracting for any hierarchical index b the set $\{\mathcal{C}_b, \mathcal{C}_{b+g}, \mathcal{C}_{b+g+1}\}$ from Section 3.2. Notice that this set is entirely determined by its indices i and j by way of $b(i,j)$. That is, we can simply denote $\{\mathcal{C}_b, \mathcal{C}_{b+g}, \mathcal{C}_{b+g+1}\}$ as the first three coefficients of the $(i+j)$-th derivative of $U^{\mathfrak{v}}_h$. As in Section 3.2 this provides our linear reconstruction, such that Eq. (35) becomes our effective sub-quadratic restriction of the $(i+j)$-th derivative of $U^{\mathfrak{v}}_h$ which we substitute into the formalism of Section 3.4. That is we set

$$U(\boldsymbol{x}_\ell)|_{i-i'+j-j'\leqslant 1} = \mathcal{C}_b + \mathcal{C}_{b+g}(\eta_i - \eta_c) + \mathcal{C}_{b+g+1}(\xi_i - \xi_c),$$

where i' and j' correspond to the sub-quadratic polynomial basis in the derivation of $U^{\mathfrak{v}}_h$ with coefficients at *level* $\mathfrak{l}(i+j)$; or, correspond to the coefficients of the linear recombination at *level* $\mathfrak{l}(i+j)$.

Then (48) is calculated, where we evaluate over every $\mathcal{C}_b$ in decreasing order. That is, for $b+g+1 \leqslant s$, we compute starting at the top $(k-1)$-st order derivative steps (48)–(56) from Section 3.4 with respect to each base coefficient b at that *level* $\mathfrak{l}(k-1)$. Then, due to the redundancy of representation for the mixed terms as discussed Section 3.3, we employ any of our minmod functions $\Phi^*_{\mathfrak{x}}$ from Section 3.3 (note that in the experiments below we always use the MUSCL minmod function). This is performed until we reach the *level* corresponding to $b=1$, at which point we perform the calculation one more time identically to that presented in Section 3.4 except in the Taylor basis.

Notice here that when the top order is linear, or when $p=k=1$ the strategy from Section 3.4 is equivalent to Section 3.5 up to a change of basis (for example in (56) the ϕ_{ij}'s become ς_{ij}'s), which provides for identical error behavior at $p=k=1$.

4. Slope limiting: numerical results

In this section we solve three example problems for an advected scalar quantity $\imath = \imath(t,\boldsymbol{x})$. All of our solutions have been run in parallel using an upwinding scheme for the choice of advective flux.

4.1. Convergence of solutions

The examples developed in Sections 4.2 and 4.3 both display discontinuities that have meaningful affects on the theoretical rates of convergence. Thus first we simply restrict to a smooth solution. That is, we use the same formalism of a scalar transport Eq. (57) developed in detail in Section 4.2, though in this case we change the initial conditions to a smooth Gaussian centered at the origin, given by $\imath_0 = a_0 e^{-(x^2+y^2)/25}$, where $a_0 = 1$ and the boundary condition is the standard transmissive condition on both $\imath_b$ and $\boldsymbol{u}_b$. This is a steady state Gaussian field that "rotates" about the origin by way of a pseudo-timestepping. The convergence results are shown in Table 1, Figs. 8 and 9.

Table 1
We show the convergence results for the h and p levels whose errors are bounded by machine precision after 64 timesteps. The L^2_{loc} projection error into the basis is also included, though, as is clear, these errors are often *below* machine double precision ($\sim 1.11 \times 10^{-16}$), and hence not particularly meaningful.

p	L^2/L^∞-error	L^2_{loc} projection error	$x = 1/h$
1	$1.76 \times 10^{-6}/5.31 \times 10^{-8}$	4.57×10^{-7}	64
2	$3.34 \times 10^{-8}/4.75 \times 10^{-10}$	1.63×10^{-11}	64
3	$6.24 \times 10^{-10}/3.19 \times 10^{-11}$	8.61×10^{-15}	64
4	$1.36 \times 10^{-11}/1.63 \times 10^{-12}$	8.22×10^{-19}	64
5	$4.14 \times 10^{-13}/6.54 \times 10^{-14}$	5.31×10^{-22}	64
1	$1.05 \times 10^{-5}/6.84 \times 10^{-7}$	7.12×10^{-6}	32
2	$3.94 \times 10^{-7}/1.10 \times 10^{-8}$	1.07×10^{-9}	32
3	$1.55 \times 10^{-8}/3.79 \times 10^{-10}$	2.16×10^{-12}	32
4	$5.35 \times 10^{-10}/2.49 \times 10^{-11}$	8.52×10^{-16}	32
5	$1.67 \times 10^{-11}/1.42 \times 10^{-12}$	2.12×10^{-18}	32
1	$5.24 \times 10^{-5}/6.17 \times 10^{-6}$	1.07×10^{-4}	16
2	$4.44 \times 10^{-6}/2.78 \times 10^{-7}$	6.85×10^{-8}	16
3	$3.56 \times 10^{-7}/2.31 \times 10^{-8}$	5.12×10^{-10}	16
4	$2.55 \times 10^{-8}/5.73 \times 10^{-10}$	8.78×10^{-13}	16
5	$1.50 \times 10^{-9}/4.36 \times 10^{-11}$	7.89×10^{-15}	16
1	$2.39 \times 10^{-4}/5.17 \times 10^{-5}$	1.30×10^{-3}	8
2	$4.73 \times 10^{-5}/6.89 \times 10^{-6}$	3.54×10^{-6}	8
3	$7.36 \times 10^{-6}/7.58 \times 10^{-7}$	9.55×10^{-8}	8
4	$1.21 \times 10^{-6}/6.41 \times 10^{-8}$	5.98×10^{-10}	8
5	$1.32 \times 10^{-7}/6.07 \times 10^{-9}$	2.19×10^{-11}	8

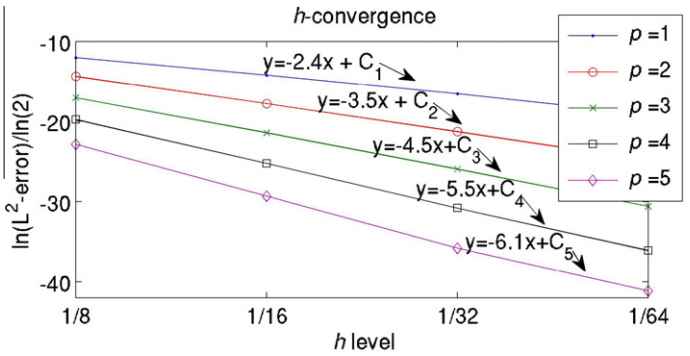

Fig. 8. The regression rates of convergence for the $p \in \{1,\ldots,5\}$ cases are given by the slope of a linear regression line taken from the data in Table 1.

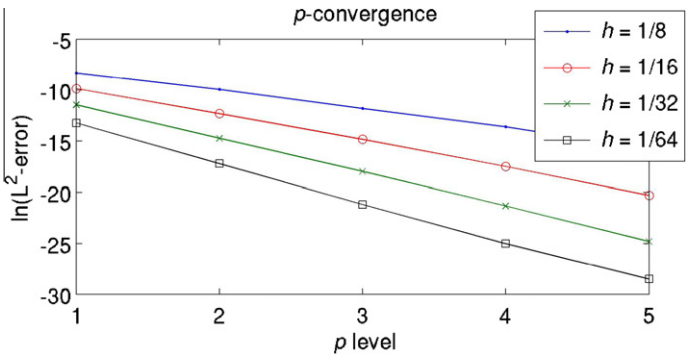

Fig. 9. The convergence in $p \in \{1,\ldots,5\}$ for the different mesh sizes, as taken from the data in Table 1.

4.2. The rotating half annular crest, cone, and hill solution

Here we solve a standard rotating landscape solution to a scalar transport equation. That is, consider the hyperbolic advection problem:

$$\partial_t \imath + \boldsymbol{u} \cdot \nabla_x \imath = 0, \tag{57}$$

with initial-boundary data given by

$$\imath_{|t=0} = \imath_0, \quad \text{and} \quad \imath_b = 0,$$

corresponding to vanishing boundary data, given a time-independent velocity vector field $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x})$ with the transported scalar quantity $\imath = \imath(t,\boldsymbol{x})$ in dimension two, such that $\boldsymbol{x} = (x,y)$ and $\boldsymbol{u} = (u,v)$.

We choose a simple square domain $\Omega = \left[-\frac{1}{2},\frac{1}{2}\right]^2$, with velocity field $\boldsymbol{u} = (y, -x)$. Then letting $\tau_{\mathcal{O}} = \pi/4$ and defining the auxiliary variables

$$\mathcal{O}_x = x\cos\tau_{\mathcal{O}} - y\sin\tau_{\mathcal{O}} \quad \text{and} \quad \mathcal{O}_y = y\cos\tau_{\mathcal{O}} + x\sin\tau_{\mathcal{O}},$$

we take initial data satisfying:

$$\imath_0 = \begin{cases} 1, & \text{if } A, \\ 1 - Ba^{-1}, & \text{if } B \leqslant a, \\ \frac{1}{4}(1+\cos\pi r), & \text{otherwise}, \end{cases} \tag{58}$$

where

$$A = (a_0 \leqslant B \leqslant a) \wedge (\mathcal{O}_x \leqslant a_1), \quad B = \sqrt{\left(\mathcal{O}_x - \frac{1}{4}\right)^2 + \mathcal{O}_y^2},$$

and

$$r = a^{-1}\min\left(a, \sqrt{\mathcal{O}_x^2 + (\mathcal{O}_y + 1/4)^2}\right),$$

taking $a = 0.18$, $a_0 = 0.025$ and $a_1 = -0.23$.

The exact solution may be determined by noticing that since for any $F(x,y)$, where $x = x(t)$ and $y = y(t)$, that

$$\frac{dF}{dt} = \partial_t F + \begin{pmatrix} x' \\ y' \end{pmatrix} \nabla F = 0,$$

which implies that for

$$\boldsymbol{u} = \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} y \\ -x \end{pmatrix}, \quad \text{we have the system} \quad x' = y \quad \text{and} \quad y' = -x,$$

that may be solved by recombining such that the solution to the second order ODE, $y'' + y = 0$ can be viewed as a generator of the rotation matrix R about the origin. That is, we obtain the clockwise transformation

$$R = \begin{pmatrix} \cos t & -\sin t \\ \sin t & \cos t \end{pmatrix}, \tag{59}$$

such that $R\boldsymbol{x}$ yields the exact solution.

For our numerical experiments, we follow a similar case to that presented in [25] setting our mesh width to $h = 1/128$ and $\Delta t = 1 \times 10^{-3}$ in keeping with the CFL condition on hyperbolic transport (e.g. see [38])). Let us briefly discuss the results shown in Figs. 10 and 11 and Table 2. We note that we have run all of our experiments on a regular structured triangular grid.(See Table 3)

In Fig. 10 we see the results for linears. The Durlofsky–Engquist–Osher [12] limiter, the vertex limiter [25] and the adapted vertex limiter seem to show qualitatively similar behaviors. The Barth–Jespersen limiter [6] is slightly more diffuse here at linears (where the adapted Barth–Jespersen shows only slight improvement over the native Barth–Jespersen limiter as well), while the BDS limiter [7] described in Section 3.4 shows by far the best L^2-error behavior and clearly maintains the best signature behavior of the solution everywhere but at the points of discontinuity, where these values are tightly redistributed. As previously suggested [25] the vertex limiter and the Barth–Jespersen limiter are both quite sensitive to mesh geometries, where the former is better suited in some sense to geometries with "sharp angles," and the latter (the Barth–Jespersen limiter) is well-suited for regular structured meshes (e.g. Delaunay triangulations). However, because of the so-called "blind diffusion" of both these regimes caused by local extrema – as discussed in Section 3.2.1 – this behavior is not entirely predictable or monotone with respect to mesh regularity, as we see below. The hierarchic linear recombination from Section 3.5, and the hierarchical reconstruction from Section 3.3 are both equivalent by construction to the BDS limiter at $p = 1$.

When $p > 1$ we see an immediate and substantial degradation in the limiting behavior of all the regimes, with the single exception of the linear restriction of the BDS limiter from Section 3.4. This is immediately prevalent at $p = 2$, where the hierarchic linear recombination method from Section 3.3 is the next best limiting regime and yet has an L^2-error more than four times that of the linear restriction. In fact, the hierarchical reconstruction method from Section 3.3 may be the most natural

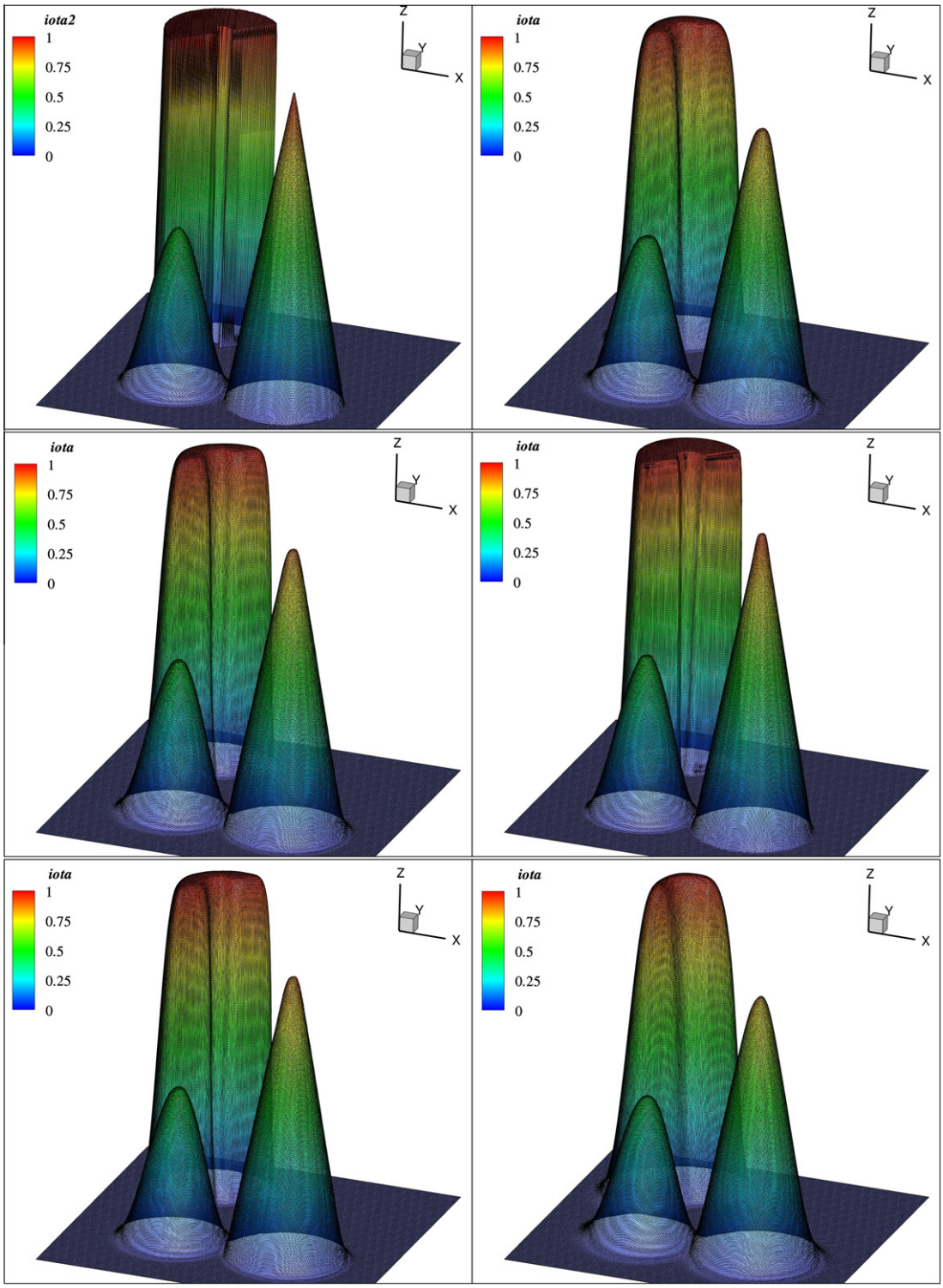

Fig. 10. Here we show the $p = 1$ results from Table 2 after one full revolution. The upper left is the exact L^2 projection at $p = 1$, the top right is the DEO limiter [12], the middle left is the vertex limiter [25,28], Section 3.2, the middle right the BDS limiter [7], Section 3.4, the bottom left the adapted vertex limiter Section 3.2.1, and the bottom right the BJ limiter [6], Section 3.2.

extension of the BDS limiter to order p, where the choice of linearization is the most direct application of the BDS scheme in the Taylor basis. But even here, where at $p = 2$ we have added only three more degrees of freedom to the polynomial hier-

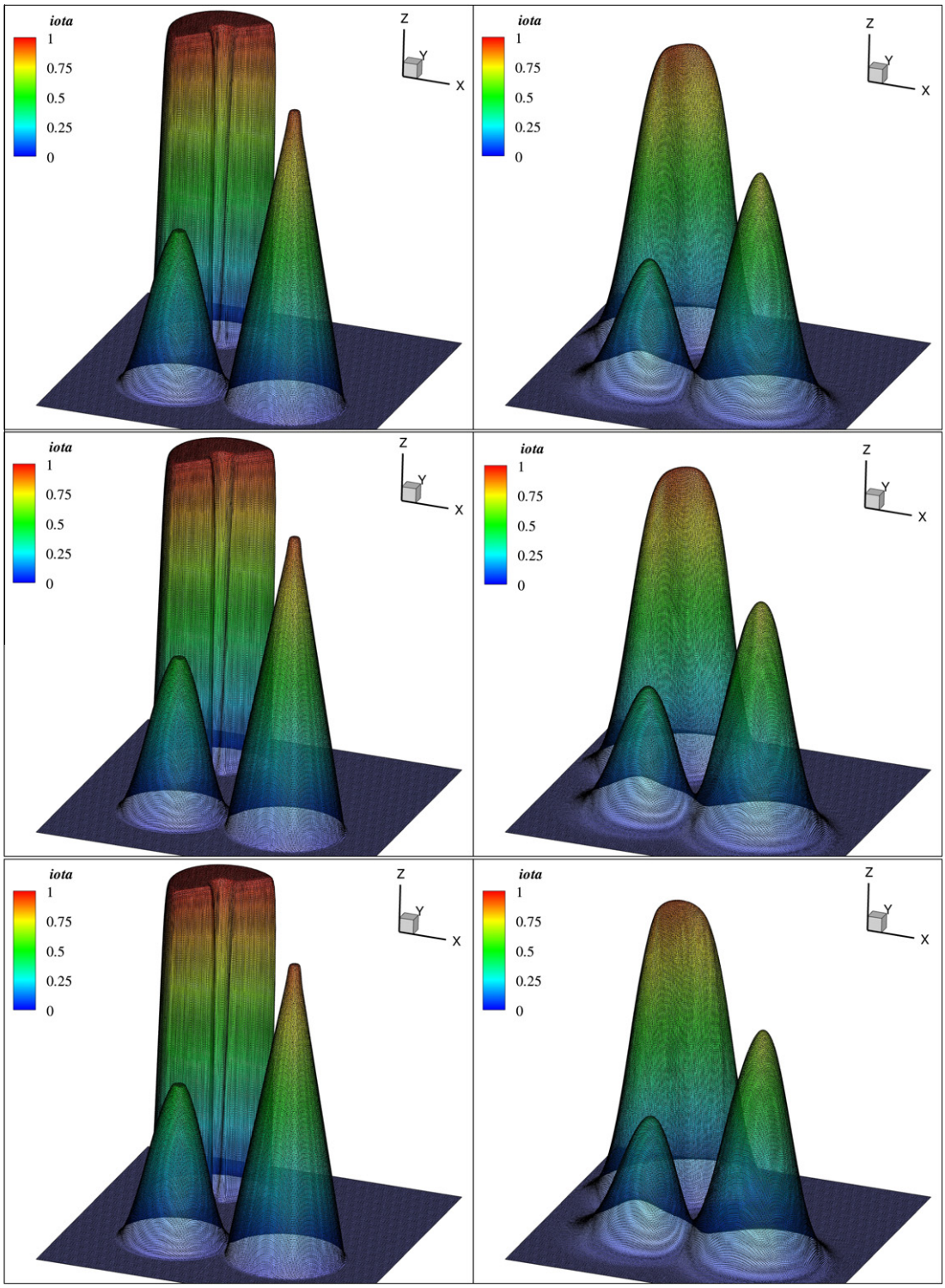

Fig. 11. Here we show the $p = 2$ to $p = 4$ results from Table 2 after one full revolution. The left column shows the linear restriction of the BDS limiter [7], Section 3.4 in descending order, while the right column shows the next best limiter, in descending order, i.e. at $p = 2$, $p = 3$ and $p = 4$ the hierarchical reconstruction$_{ENO}$ [1,27], Section 3.3.

archical basis, we see that performing the limiter on the linear reconstructions – which amounts to performing the limiting procedure on only two more components (i.e. the linear components which are limited with respect to their respective slopes) – shows a substantial loss locally in the sharpness of the resolution along the discontinuities.

Table 2
We give the L^2 and L^∞-errors of the approximate solutions after one full rotation with respect to (57), setting $h = 1/256$, $\Delta t = 1 \times 10^{-3}$ and using Runge–Kutta SSP (5,3). The error ratio for the solution with no limiter at $p = 1$ is $L^2/L^\infty = 2.55 \times 10^{-4}/0.61$, at $p = 2$ is $L^2/L^\infty = 2.28 \times 10^{-4}/0.44$, at $p = 3$ is $L^2/L^\infty = 1.71 \times 10^{-4}/0.35$, and for $p > 3$ is unstable. Though, as expected, the error in the stable unlimited solutions concentrate along the discontinuities demonstrating sharp ($\geqslant 10\%$ cell-wise in ι) overshoots and undershoots.

p	**Limiter type**	$\left(\frac{L^2\text{error}}{L^\infty\text{error}}\right)$	**Limiter type**	$\left(\frac{L^2\text{error}}{L^\infty\text{error}}\right)$
1	BJ limiter [6], Section 3.2	$\left(\frac{1.5\times10^{-3}}{0.73}\right)$	Adapted BJ, Section 3.2.1	$\left(\frac{1.5\times10^{-3}}{0.73}\right)$
1	DEO limiter [12]	$\left(\frac{1.1\times10^{-3}}{0.71}\right)$	BDS limiter [7], Section 3.4	$\left(\frac{5.9\times10^{-4}}{0.67}\right)$
1	Vertex [25,28], Section 3.2	$\left(\frac{1.1\times10^{-3}}{0.73}\right)$	Adapted vertex, Section 3.2.1	$\left(\frac{1.0\times10^{-3}}{0.72}\right)$
1	Recombination, Section 3.5	$\left(\frac{5.9\times10^{-4}}{0.67}\right)$	$\text{Reconstruction}_{\text{MUSCL}}$ [1,27], Section 3.3	$\left(\frac{5.9\times10^{-4}}{0.67}\right)$
2	BJ limiter [6], Section 3.2	$\left(\frac{2.3\times10^{-3}}{0.74}\right)$	Restriction [7], Section 3.4	$\left(\frac{5.0\times10^{-4}}{0.64}\right)$
2	Vertex [25,28], Section 3.2	$\left(\frac{2.3\times10^{-3}}{0.73}\right)$	Adapted vertex, Section 3.2.1	$\left(\frac{2.3\times10^{-3}}{0.74}\right)$
2	Recombination Section 3.5	$\left(\frac{2.2\times10^{-3}}{0.74}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{2.3\times10^{-3}}{0.73}\right)$
3	BJ limiter [6], Section 3.2	$\left(\frac{2.6\times10^{-3}}{0.72}\right)$	Restriction [7], Section 3.4	$\left(\frac{4.7\times10^{-4}}{0.73}\right)$
3	Vertex [25,28], Section 3.2	$\left(\frac{2.6\times10^{-3}}{0.72}\right)$	Adapted vertex Section 3.2.1	$\left(\frac{2.5\times10^{-3}}{0.73}\right)$
3	Recombination Section 3.5	$\left(\frac{2.6\times10^{-3}}{0.72}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{2.2\times10^{-3}}{0.75}\right)$
4	BJ limiter [6], Section 3.2	$\left(\frac{2.8\times10^{-3}}{0.72}\right)$	Restriction [7], Section 3.4	$\left(\frac{4.8\times10^{-4}}{0.69}\right)$
4	Vertex [25,28], Section 3.2	$\left(\frac{2.9\times10^{-3}}{0.72}\right)$	Adapted vertex Section 3.2.1	$\left(\frac{2.8\times10^{-3}}{0.72}\right)$
4	Recombination Section 3.5	$\left(\frac{2.9\times10^{-4}}{0.72}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{2.6\times10^{-3}}{0.73}\right)$

Table 3
We give the L^2 and L^∞-errors of the approximate solutions after T corresponding to a 1/4 rotation with respect to (60), setting $h = 1/128$, $\Delta t = 5 \times 10^{-4}$ and using Runge–Kutta SSP*(5,3)*. The error ratio for the solution with no limiter at $p = 1$ is $L^2/L^\infty = 3.9 \times 10^{-3}/0.40$, at $p = 2$ is $L^2/L^\infty = 2.9 \times 10^{-3}/0.34$, at $p = 3$ is $L^2/L^\infty = 2.4 \times 10^{-3}/0.23$, and for $p > 3$ is unstable. Again, as in Table 2, the unlimited solutions are dominated by local overshoots and undershoots along the discontinuities.

p	**Limiter type**	$\left(\frac{L^2\text{error}}{L^\infty\text{error}}\right)$	Limiter type	$\left(\frac{L^2\text{error}}{L^\infty\text{error}}\right)$
1	BJ limiter [6], Section 3.2	$\left(\frac{1.2\times10^{-2}}{0.49}\right)$	Vertex [25,28], Section 3.2	$\left(\frac{1.2\times10^{-2}}{0.51}\right)$
1	DEO limiter [12]	$\left(\frac{1.0\times10^{-2}}{0.47}\right)$	BDS limiter [7], Section 3.4	$\left(\frac{6.8\times10^{-3}}{0.40}\right)$
1	Recombination Section 3.5	$\left(\frac{6.8\times10^{-3}}{0.40}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{6.8\times10^{-3}}{0.40}\right)$
2	BJ limiter [6], Section 3.2	$\left(\frac{1.9\times10^{-2}}{0.50}\right)$	Restriction [7], Section 3.4	$\left(\frac{6.6\times10^{-3}}{0.38}\right)$
2	Vertex [25,28], Section 3.2	$\left(\frac{1.9\times10^{-2}}{0.50}\right)$	Adapted vertex Section 3.2.1	$\left(\frac{1.9\times10^{-2}}{0.50}\right)$
2	Recombination Section 3.5	$\left(\frac{1.9\times10^{-2}}{0.50}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{1.8\times10^{-2}}{0.52}\right)$
3	BJ limiter [6], Section 3.2	$\left(\frac{2.2\times10^{-2}}{0.50}\right)$	Restriction [7], Section 3.4	$\left(\frac{7.7\times10^{-3}}{0.39}\right)$
3	Vertex [25,28], Section 3.2	$\left(\frac{2.3\times10^{-2}}{0.50}\right)$	Adapted vertex Section 3.2.1	$\left(\frac{2.2\times10^{-2}}{0.50}\right)$
3	Recombination Section 3.5	$\left(\frac{2.3\times10^{-2}}{0.50}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{2.2\times10^{-2}}{0.51}\right)$
4	BJ limiter [6], Section 3.2	$\left(\frac{2.3\times10^{-2}}{0.50}\right)$	Restriction [7], Section 3.4	$\left(\frac{7.7\times10^{-3}}{0.38}\right)$
4	Vertex [25,28], Section 3.2	$\left(\frac{2.3\times10^{-2}}{0.50}\right)$	Adapted vertex Section 3.2.1	$\left(\frac{2.3\times10^{-2}}{0.50}\right)$
4	Recombination Section 3.5	$\left(\frac{2.3\times10^{-2}}{0.50}\right)$	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	$\left(\frac{2.2\times10^{-2}}{0.51}\right)$

The reason for this loss of resolution is not entirely mysterious or unexpected, though previous work [25] has demonstrated geometries where this degradation is not immediately observable at $p = 2$, and this behavior seems related to the mass lumping strategy previously discussed in Section 3.2.1 (which deserves closer analysis). Nevertheless, here we see that as p increases the number of applications of the limiter to the solution increases as a function of the degrees of freedom at the $(p-1)$-st degree (i.e. $(p-1)(p-2)/2$). In fact this is true for each of the limiting regimes, with the exception of the hierarchical reconstruction methods from Section 3.3, which actually perform yet another iteration of the limiter by employing one of the minmod functions at top order. However, the hierarchical reconstruction methods also seem to benefit from the fact that they utilize information coming from nonlinearities present in the solution at *every level* by linearizing with respect

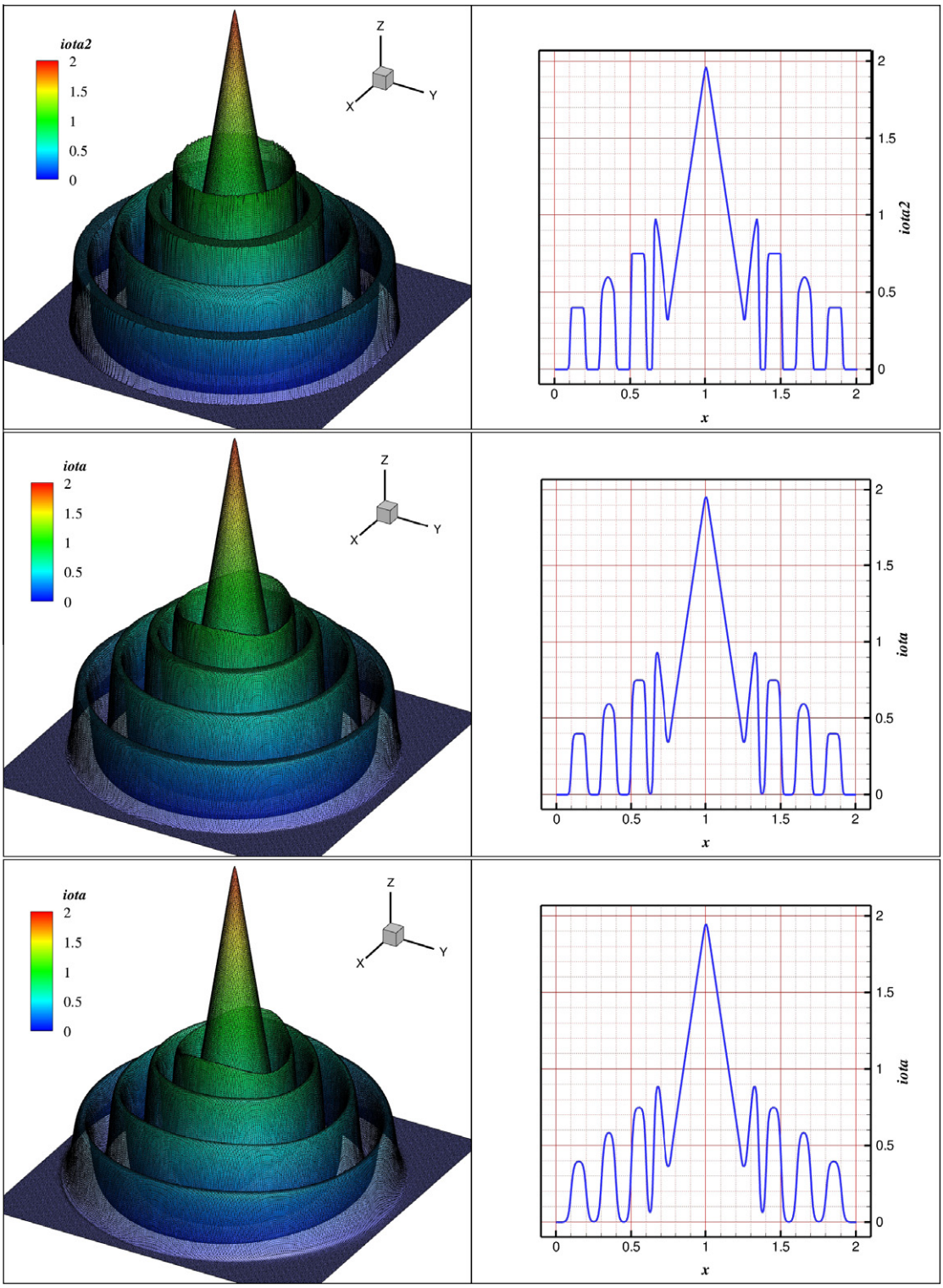

Fig. 12. Here at top we show the L^2-projection of the exact solution at $p = 7$, with the xz-plane slice on the right after 1/4 turn. The middle shows the $p = 1$ case of the linear restriction [7], Section 3.4, and the bottom shows the $p = 1$ DEO limiter [12].

to these nonlinearities (e.g. Eq. (44)) – in contrast to the vertex-based schemes which linearize about a single monomial component (35) of the expansion, and then utilize a regularizing constraint such as (38). It turns out that the addition of this

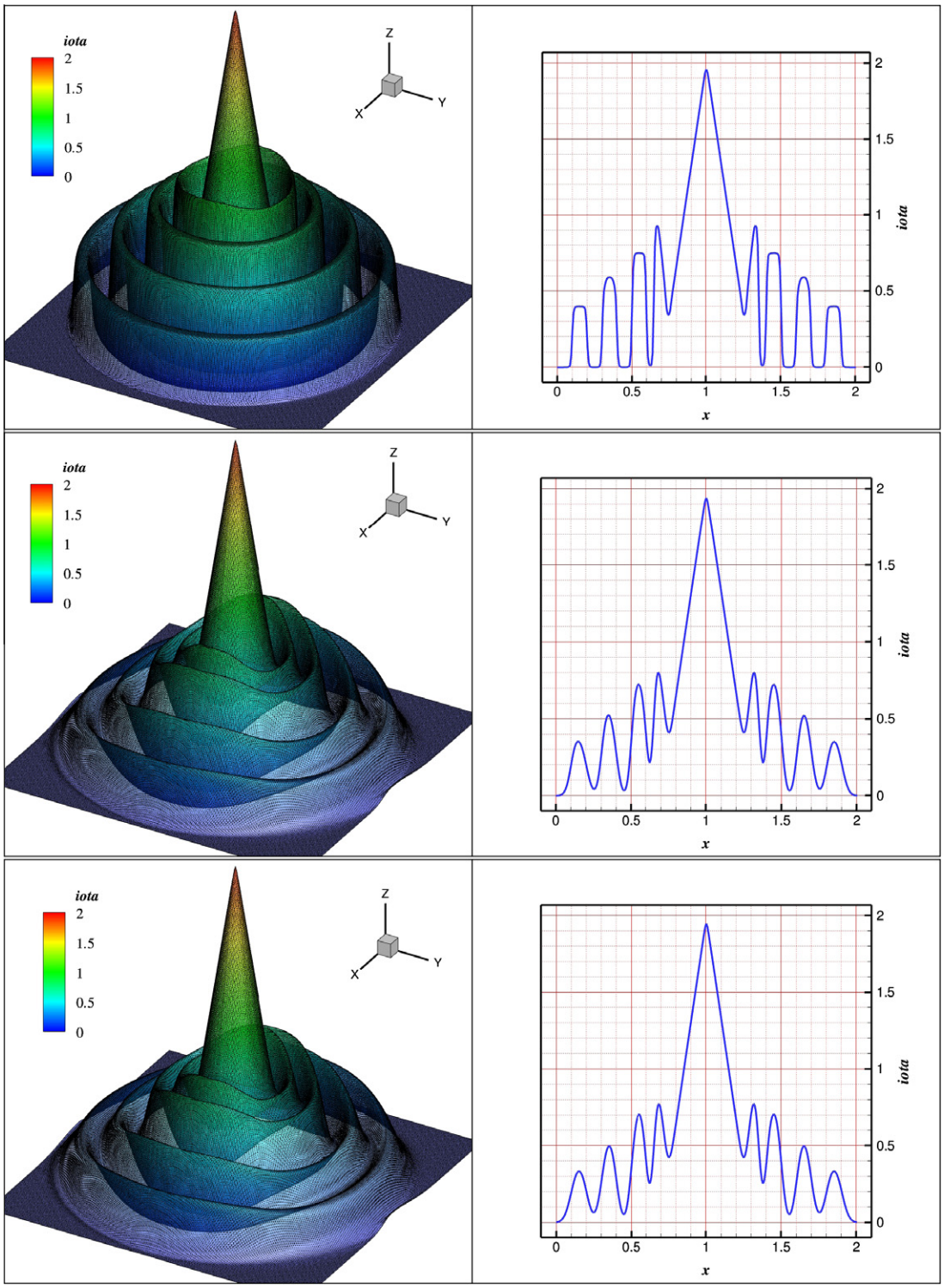

Fig. 13. At top we show the $p = 4$ linear restriction after 1/4 turn. The middle shows the $p = 3$ linear reconstruction [1,27], Section 3.3, and the bottom the $p = 2$ linear recombination Section 3.5.

nonlinear signature behavior at higher order seems to allow the hierarchical reconstruction methods to capture the profile more completely, even with the additional application of the limiting regime at each timestep.

However, by far the most effective limiting regime for $p > 1$ is the linear restriction of the BDS limiter from Section 3.4, where in $\Re$ the ε has been set to 10^{-4}. Again, this result is not entirely unexpected, since slope limiting, as its name suggests, finds its roots in limiting the slopes of lines with respect to some linear basis [40]. That having been said, it then seems unlikely that one should be able to expect an improvement in the accuracy of a solution near a sharp front simply by applying the slope limiter more frequently to the linearization of its respective monomial components. Since, for example, if one assumes (fairly realistically) that the top order component has an approximately fixed order error which is introduced upon application of the limiter to the FEM solution, then each subsequent application of the limiter to the lower *level* $\mathfrak{l}$ components should only be able to increase the subsequent error introduced over all. In the hierarchical reconstruction methods of [1,27], on the other hand, the componentwise minmod function attenuates this effect somewhat, as does the fact that all of the limited higher order components serve to help limit the lower order components at every *level* $\mathfrak{l}$.

Before discussing this further, let us first confirm that this result is not simply a special case of (57) which demonstrates a pathological behavior with respect to (58). Below we take a solution with admits a number of additional types of singular submanifolds that help to further explicate each limiter's behavior.

4.3. Steady state convective torque

Now we show a steady state solution to Eq. (57), which effectively isolates the error present in the form of torque away from the steady (discontinuous) state in a rotating constant frame solution. Our goal here is to present a more difficult set of singular submanifolds $\mathcal{B}_i \subset \mathcal{B}$ present with respect to a steady state solution (where the solution here is thought of as the base manifold $\mathcal{B}$) in order to more completely isolate the error explicitly introduced by the limiting regimes over varying order p.

Here we work over the Cartesian domain $\Omega = [0,2] \times [-1,1]$, given the same boundary conditions from Section 4.1, and where the exact steady solution is characterized by a velocity field satisfying $\boldsymbol{u} = (y, 1-x)$ and a steady state scalar field $\imath$ given by:

$$\imath = \begin{cases} 2 - \frac{2}{3}r, & \text{if } r \leqslant a_1 \\ 2a_1(1+\cos[(r-a_2)\pi]), & \text{if } a_1 < r \leqslant 3.5a_3 \\ 3a_1, & \text{if } 4a_3 \leqslant r \leqslant 2a_1 \\ 3a_3(1+\cos[(r-a_2)\pi]), & 6a_3 \leqslant r \leqslant 7a_3 \\ a_1, & \text{if } 8a_3 \leqslant r \leqslant 9a_3 \\ 0, & \text{otherwise} \end{cases} \tag{60}$$

where

$$r = \sqrt{(x-1)^2 + (y)^2}, \quad a_1 = \frac{1}{4}, \quad a_2 = \frac{13}{3} \quad \text{and} \quad a_3 = \frac{1}{10}.$$

The solution $\mathcal{B}$ as shown in Fig. 12 is augmented from the relatively well-behaved circular convection case analyzed in [25]. Here we have similar outer rings (though substantially "thinned"), but have supplemented a pair of inner ring submanifolds that have a thickness of no more than a single point that similarly intersects an inner cone along a line of singular points, and with a very thin island outer ring. These initial conditions are not particularly well-behaved, as can be seen in Fig. 12, where even in the L^2-projected exact solution at $p = 7$ there are variations (jagged lines) at the mesh resolution along the lines of singular points. To compound this, we use a larger domain than that of [25], which effectively doubles the velocity of the pseudo-timestepping in the y-direction, providing for even more instability in the solution space.

Note that in Figs. 12 and 13, the asymmetry in the solution is merely due to that fact that we have only gone a quarter turn, thus the diffusive signature of each limiter has only been advected a quarter turn, and accumulates or dissipates according to the local behavior of the advective flux.

Now, notice that the adapted limiters from Section 3.2.1 are not well-suited to handle (60) at all. In fact (40) is, in particular, adapted to represent a case which almost always leads to problems, since it does not deal differentially with the special case of $U^{\max}_{i,b} = U^{\min}_{i,b}$, which in (40) up to the resolution h is the case for nearly every element in the domain, leading to an almost globally uniform "blind diffusion." In fact the adapted cases are almost identical to the native cases at low p – when not explicitly dealing with $U^{\max}_{i,b} = U^{\min}_{i,b}$ – even though the native vertex and Barth–Jespersen limiters do not recognize local extrema at all, while the adapted cases do recognize local extrema up to, but not including, the degenerate case of $U^{\max}_{i,b} = U^{\min}_{i,b}$. As p increases the repeated iterations of the limiter swamps this behavior in both the native and adapted limiters, and thus the solutions converge to the same value. It is possible that a mass lumping strategy might mitigate some of these affects (see [25] for more information on this technique).

Moreover, in this example (60) the Barth–Jespersen limiter is clearly initially more diffuse than the native vertex limiter, which is primarily due here to the fact that the singular submanifolds are chosen such that they – again up to the mesh resolution h – spatially oscillate on a local neighborhood which is larger than the characteristic length of the *edge neighborhood*, and so the *focal neighborhood* is a more appropriate area to "sense" in order to capture this semi-localized signature behavior. Moreover, the problem of local extrema as discussed in Section 3.2.1 is of lesser importance in this case, since up to the set of codimension one submanifolds of Ω_h in (60), the entire domain is characterized and dominated by extremely sharp profiles,

making the diffusion – which is potentially "blind" near smooth regions – more appropriate here. However, again as p increases this behavior gets swamped by the repeated iterations.

The linear restriction of the BDS limiter [7], Section 3.4 once again demonstrates the best limiting behavior as a function of increasing p, which again seems to emphasize the fact that limiting a solution for $p > 1$ must somehow account for the implicit nonlinearity present internal to the cell in a relatively explicit way; or, at least, a way which is fully functionally coupled to the entire solution as it exists everywhere on the local cell.

Nevertheless, the linear restriction still substantially outperforms all of the competing limiting regimes. There seems to be some indication here that, at least presently, one may expect that near areas dominated by shocks the best accuracy that one can hope for is linear accuracy, while still hoping to preserve physically important characteristics of the solution (e.g. positivity perserving, local conservation of mass, etc.). Of interest, is that this observation falls very neatly in line with the state of the art in *hp*-adaptive numerical schemes, where a general heuristic follows that for potentially discontinuous solutions, in areas of high cell-wise variability, the local order of p is only increased if inter-element jumps are small or bounded and controlled, and the internal cell-wise variation is strictly bounded above by the cell(s) (usually a subset of cells) containing the global maximum [10,30].

We explore this issue some in the subsequent section as it applies to p-enrichment, though we also note that at present we are not aware of any formal results which come anywhere near to formulating a theorem that subsumes this observational fact (which may in general prove to be only one part of the story). Nevertheless, such a result would be of substantial importance to the field, as would a counter example, which here could simply be the development of a fully p convergent slope limiting regime that limits at all *levels* I while still preserving the important physical features of the solution (and of course does so without relying on prior knowledge, such as the existence of an exact solution).

5. Adjoining the dynamic *p*-enrichment

Here we present a number of generalizable p-enrichment/de-enrichment schemes based on local data and apply them to the problems from Section 4. These p-enrichment schemes may be viewed as alternatives to, for example, the specific energy methods presented in [30] which rely upon the variational global entropy of the system of equations, and those discussed in [11,5,19], which, as in [30], try to maximally enrich the domain based on global solution behavior taken with respect to the available computational resources and either *a priori* or *a posteriori* estimates.

5.1. A general approach based on local data

We implement a dynamic p-enrichment scheme that utilizes a number of different methodologies in order to capture higher order structure in areas of "permissible variability." This scheme is built with respect to our collection of p-adaptive slope limiters from Section 3, such that we inherently arrive with a dynamically limited p-enriched solution.

The nuance of implementing such a scheme in the generalized formulation is that the solution must demonstrate a minimal smoothness condition in areas of p-enrichment, while in areas approaching discontinuity, p-enrichment must be suppressed in order to maintain stability (especially in the absence of a limiter). This issue is not a concern of course when one is able to make smoothness assumptions *a priori* about the entire solution space over $\Omega \times (0,T)$ (*viz.* the formalism of [9,23]), and has been shown to demonstrate very nice behavior especially in solution spaces which are not only smooth, but where in particular one would like to resolve stable areas of maximal variation (e.g. as are applicable in some storm surge model applications [23]).

Nevertheless, in the context of a slightly more generalized system of equations with, for example, a coupled hyperbolic equation (or possessing a hyperbolic character in a system of equations) such as (57), such assumptions cannot generally be made over the entire discrete solution space over $\Omega_h \times (0,T)$, since areas demonstrating strong local gradients $\nabla_x \boldsymbol{U}_h$ may indicate the presence or formation of numeric shock fronts (even given smooth initial data), in which case local p-enrichment has a destabilizing effect on the solution (that is, the weak approximation to a discontinuity becomes more ill-behaved with respect to increasing p).

Here we are concerned with dynamically p-adapted solutions to the generalized formulation of (13) and (14) in conjunction with the slope limiters presented in Section 3. We implement a very simple set of p-enrichment strategies, which as we will see, generally tend to undersample the variational space (e.g. in contrast to, for example, the *poor man's* or *poor man's greedy* algorithm of [11] which always adapts based on some percentage of a global relative bound). The reason for this simplification here is to reduce the number of varying parameters in the scheme, in order to isolate the stability of the solution with respect to the limiting schemes of Section 3. Hence, we simply set hard tolerances which do not depend on, for example, the available computational resources or global bounds on the solution.

Now, in order to additionally deal with both smooth and discontinuous initial-boundary data (as well as smooth and discontinuous solutions in $(0,T)$) we implement the following two distinct dynamic p-enrichment schemes – namely we designate them: Type I and Type II p-enrichment schemes. We also note that in this section all functions are defined with respect to the master element $\mathcal{M}$ representation.

The first type of enrichment scheme (i.e. Type I) applies to solutions in which smoothness may be assumed *a priori* over the entire domain $\Omega \times (0,T)$. That is, taking the approximate solution vector $\boldsymbol{U}_h$ we compute the auxiliary sensor over each i-th component of of the state variable $\boldsymbol{U}$ (having m components, as in Section 2):

$$\Pi_j^i = \left| \frac{\boldsymbol{U}_h^i|_{\omega_j} - \boldsymbol{U}_h^i|_c}{\chi_j} \right|, \tag{61}$$

where c is the centroid of element Ω_e and ω_j is the midpoint of the j-th edge of Ω_e, and the solution $\boldsymbol{U}_h$ is evaluated at these two points, respectively. For smooth solutions, the function χ_j may be set to either the distance $\chi_j = |\omega_j - c|$ as in [23], or the product $\chi_j = \omega_j c$ as in [9]. In either case, over each timestep n the following p-enrichment functional $\mathfrak{E}_{e_l} = \mathfrak{E}_{e_l}(\mathcal{P}^k(\Omega_{e_l}^n))$ is evaluated over each cell Ω_{e_l}:

Type I p-enrichment

$$\mathfrak{E}_{e_l} = \begin{cases} \mathcal{P}^{k+1}(\Omega_{e_l}^n) & \text{if } \left((\sup_i \sup_j \Pi_j^i \geqslant \epsilon) \wedge (k+1 \leqslant p_{\max}) \right) \vee (\tau_0 \geqslant t^w), \\ \mathcal{P}^{k-1}(\Omega_{e_l}^n) & \text{if } (\inf_i \sup_j \Pi_j^i < \epsilon) \wedge (k-1 \geqslant p_{\min}) \wedge (\tau_0 \geqslant t^w), \\ \mathcal{P}^{k}(\Omega_{e_l}^n) & \text{otherwise}, \end{cases} \tag{62}$$

where τ_0 is a counter that restricts the p enrichment/de-enrichment such that it may only occur every t^w timesteps, and where $k \in \{1,\ldots,p\}$.

For solutions demonstrating approximately nonzero local gradients $\nabla_x \boldsymbol{U}_h \not\approx 0$, wherein we might expect local discontinuities we must find an estimate of the local relative "smoothness" of $\boldsymbol{U}_h$. One way of doing this is by setting the auxiliary sensor equal to the following Van Leer minmod function across elements (as used in [9]):

$$\Pi_j^i = \text{minmod}(\boldsymbol{U}_h^i|_{v_j^+} - \boldsymbol{U}_h^i|_c, \boldsymbol{U}_h^i|_{v_j^-} - \boldsymbol{U}_h^i|_c), \tag{63}$$

where v_j is the j-th vertex of Ω_{e_l}. As $\Pi_i^j \to 0$ the solution becomes smoother, and one may subsequently employ (62).

A slightly simpler method of dealing with discontinuous solutions simply using local information is to define a local smoothness estimator (as discussed in [41,33]) such that we again may calculate an elementwise version of (61) depending only on the the interior of Ω_{e_l}, such that:

$$\Pi_i^{e_l} = \left(\frac{\|\boldsymbol{U}_h^i - \breve{\boldsymbol{U}}_h^i\|_{L^q(\Omega_{e_l})}}{\|\boldsymbol{U}_h^i\|_{L^q(\Omega_{e_l})}} \right), \tag{64}$$

for the L^q norms (except when $q = 2$ in which case we take the standard inner product, as used in our examples below), where $\breve{\boldsymbol{U}}_h$ is the elementwise projected solution $\mathcal{P}^{k-1}(\Omega_{e_l}^n)$, such that in our mixed version (62) becomes:

Type II p-enrichment

$$\mathfrak{E}_{e_l} = \begin{cases} \mathcal{P}^{k+1}(\Omega_{e_l}^n) & \text{if } (\sup_i \log_{10} \Pi_i^{e_l} \leqslant A) \wedge (k+1 \leqslant p_{\max}), \\ \mathcal{P}^{k-1}(\Omega_{e_l}^n) & \text{if } (\inf_i \log_{10} \Pi_i^{e_l} \geqslant A) \wedge (k-1 \geqslant p_{\min}) \wedge (\tau_0 \geqslant t^w), \\ \mathcal{P}^{k}(\Omega_{e_l}^n) & \text{otherwise}, \end{cases} \tag{65}$$

where the bound satisfies

$$A = \begin{cases} \log_{10} \tilde{c} k^{-q^2} + c, & \text{for } p > p_{\min} \\ \sup_i \log_{10} \Pi_i^{e_l}, & \text{otherwise} \end{cases} \tag{66}$$

such that $\tilde{c}, c \in \mathbb{R}^+$ are user defined constants, where $\tilde{c} \in (0, 10)$ is recommended (see for example [41]) for resolving discontinuities in the context of hp-adaptivity, and where we have found $c \in (-2, 2)$ optimal. The basic intuition that underpins the use of (64) is the observation that discontinuous basis functions are assumed to decay, for smooth solutions, at a rate comparable to that of the Fourier coefficients in a standard expansion of the solution – which clearly decay at a rate of $1/k^4$ for $q = 2$ (see [41,33,34]), to which we obtain an indicator of the relative local regularity of the solution, i.e. the faster the

Table 4
We give the L^2-errors of the approximate solutions after T corresponding to a 1/4 rotation with p-enrichment on (60), setting $h = 1/128$, $\Delta t = 5 \times 10^{-4}$ and using Runge–Kutta SSP (5,3).

p	Limiter type	Type I, L^2	Type II, L^2	ϵ	c,	$\bar{c}$	t^w	q
1–5	BJ limiter [6], Section 3.2	3.66×10^{-3}	2.18×10^{-3}	0.1	−1	0.1	0	2
1–5	Vertex [25,28], Section 3.2	3.10×10^{-3}	2.10×10^{-3}	0.1	−1	0.1	0	2
1–5	Restriction [7], Section 3.4	X	8.03×10^{-4}	0.1	−1	0.1	0	2
1–5	Recombination Section 3.5	2.17×10^{-3}	2.09×10^{-3}	0.1	−1	0.1	0	2
1–5	$\text{Reconstruction}_{\text{ENO}}$ [1,27], Section 3.3	1.98×10^{-3}	1.91×10^{-3}	0.1	−1	0.1	0	2

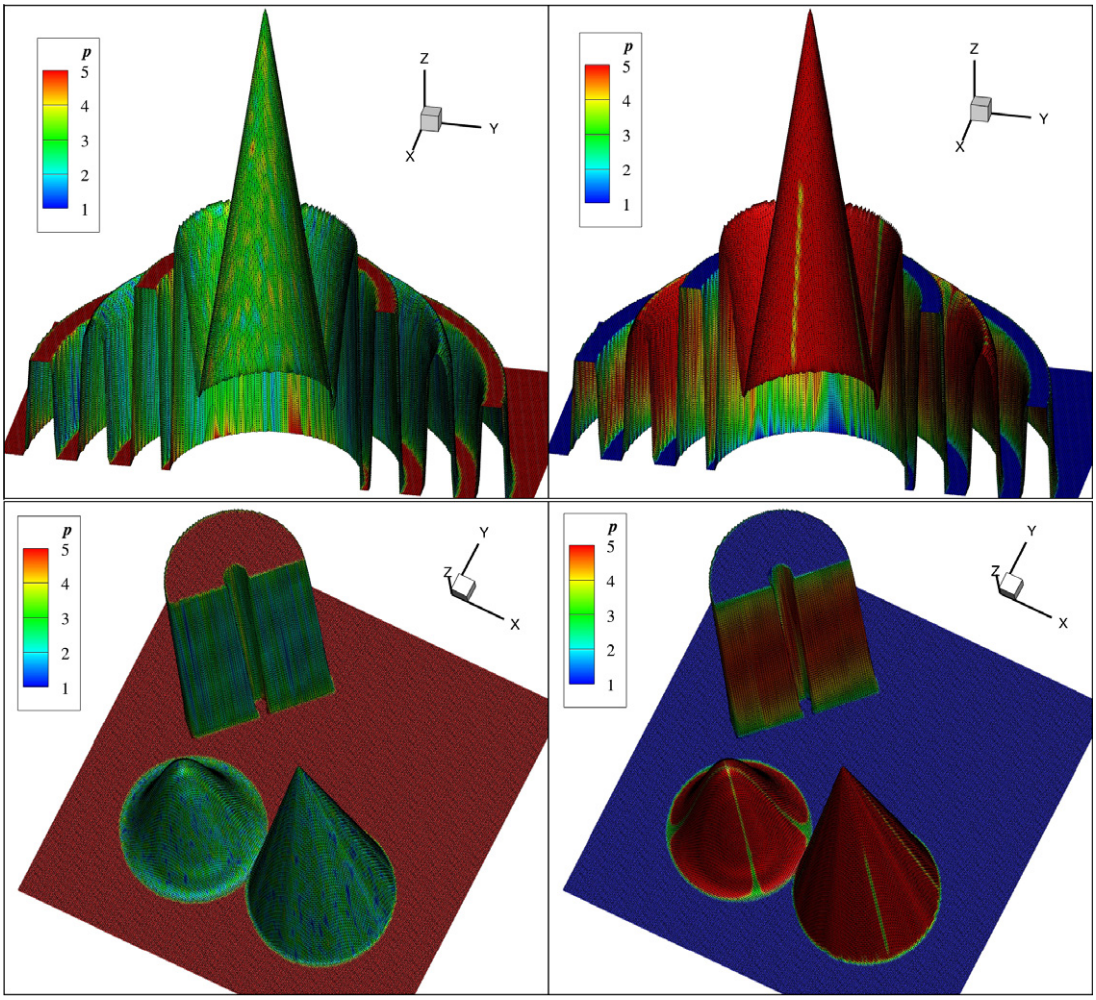

Fig. 14. Here we show the p values mapped over the $p = 1$, L^2-projected profiles at $T = 0.003$ on the top solutions, and at $T = 0.06$ on the bottom solutions using the settings given in Table 4. The solutions on the left use the Type II p-enrichment, and those on the right use the Type I p-enrichment.

coefficients decay, the smoother the local solution. Thus we obtain Eq. (64), which approaches zero as the solution becomes smoother, where setting $c > 0$ is a sharper restriction than the more permissive (i.e. less stable) condition $c < 0$.

The results are shown in Table 4 and Fig. 14. As expected from before, the linear restriction from Section 3.4 is again by far the most accurate of the choice of limiters when it is stable, where it is important to note that in the p-enrichment case the restriction function $\mathfrak{R}$ from Section 3.4 is calculated using $\varepsilon = 10^{-4}$, which has the effect of passing cells containing steep gradients to the dynamic p-enrichment functions $\mathfrak{E}$. This is enough, it turns out, to make the Type I p-enrichment regime unstable with respect to the dynamically adaptive linear restriction limiting regime from Section 3.3 due in part to the function of $\mathfrak{R}$, which creates an unstable p-flickering along sharp profile edges. However, even turning off the p-de-enriching functionality of $\mathfrak{R}$ does not help in this case, since the linear restriction still zeros out the higher order components, which in

the Type I case effectively still allows p to flicker locally, leading to the formation of instabilities along sharp edges. We also note that in Table 4 we have suppressed the L^∞-error, as numerical experimentation suggests that very small changes in the p-enrichment settings $\epsilon, c, \tilde{c}$, and t^w can cause big shifts in L^∞_{loc}, which make the L^∞-error a deceptive measure in the discontinuous p-enrichment case.

Finally, we emphasize that the p-enriched slope limited solutions show substantially better accuracy than the constant-in-p solutions from Section 4.2. This can be attributed in large part to the observation that the majority of error in the solutions is accumulated along the discontinuities, which is precisely where the p-enrichement schemes p-transition between *levels* (see Fig. 14). Hence, in both Type I and Type II cases (i.e. spatially, from either side of the discontinuity) the p-enrichement strongly attenuates (by explicit truncation) the oscillatory instabilities present in these regions, as long as the solution does not flicker unstably between them.

6. Conclusion

We have presented a discontinuous Galerkin finite element method for solving dynamically p-enriched solutions with consistent slope limiting to arbitrary order in two spatial dimensions over generalized coupled systems of PDEs. We have provided a formalism for transforming between the polynomial basis of different regimes in order to move between representation spaces. We then introduced, up to but not including a substantial choice of minmod functions, seven dynamic-in-p slope limiting regimes, and performed numerical experiments on these regimes in order to develop a sense of their strengths and weaknesses. We found that our numerical results suggest that, given discontinuous initial data, slope limiting over fixed order solutions when $p > 1$ is most effectively accomplished by restricting back to the linear case and using a sharp limiter in that regime, rather than keeping the higher order data and trying to limit it in a consistent way – which we found introduces more numerical diffusion (i.e. error) on average over time.

We then presented two types of p-enrichment schemes, fully coupled to the above slope limiting regimes. These schemes are designed to exploit certain properties of the solution, and simple algorithms were implemented. We then tested these coupled systems on the same model problem in order to develop a sense of how dynamic-in-p systems perform relative to fixed-in-p systems. Here again, we found that restricting to the linear case seems to be the most effective (and also, incidentally, efficient) way of limiting a dynamically p-adapting solution. Moreover, we found that in general using the Type I and Type II methods of p-enrichment the accuracy of the solution was substantially improved (i.e. by an order of magnitude) with respect to the native solution using only the dynamic-in-p slope limiters of Section 3.

Future directions include taking the slope limited solution from Section 3 coupled to the p-enrichment scheme from Section 5 and adding dynamic h-adaptivity to it, in order to fully exploit the power of hp-adaptive convergence.

Acknowledgements

The first author would like to thank P.G. Schmitz, Wenhao Wang, Troy Butler, Corey Trahan, Nishant Panda and Jennifer Proft for helpful conversations. In addition, several very helpful insights have been provided by Prof. D. Kuzmin via correspondence, to which we are very grateful. The authors would also like to thank the anonymous reviewers for their careful comments and suggestions, and to further acknowledge the support of the National Science Foundation grants OCI-0749015,OCI-0746232 and DMS-0915118.

References

[1] R. Abgrall, On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation, J. Comput. Phys. 114 (1) (1994) 45–58, doi:10.1006/jcph.1994.1148. ISSN 0021-9991.
[2] V. Aizinger, C. Dawson, A discontinuous Galerkin method for two-dimensional flow and transport in shallow water, Adv. Water Res. 25 (1) (2002) 67–84, doi:10.1016/S0309-1708(01)00019-7. ISSN 0309-1708. URL http://www.sciencedirect.com/science/article/B6VCF-44PK3KB-5/2/51beaaea1191c299bcd3a0d40beca43d.
[3] D.N. Arnold, F. Brezzi, B. Cockburn, D. Marini, Discontinuous Galerkin methods for elliptic problems, in: Discontinuous Galerkin methods, Lecture Notes Computer Science and Engineering, vol. 11, Springer, Berlin, 2000, pp. 89–101. Newport, RI, (1999).
[4] L.V. Ballestra, R. Sacco, Numerical problems in semiconductor simulation using the hydrodynamic model: a second-order finite difference scheme, J. Comput. Phys. 195 (1) (2004) 320–340, doi:10.1016/j.jcp.2003.10.002. ISSN 0021-9991.
[5] W. Bangerth, O. Kayser-Herold, Data structures and requirements for hp finite element software, ACM Trans. Math. Softw. 36 (2009) 4:1–4:31. ISSN 0098-3500. URL http://doi.acm.org/10.1145/1486525.1486529.
[6] T. Barth, D.C. Jesperson, The design and application of upwind schemes and unstructured meshes, AIAA paper 89 (0366) (1989) 1–12.
[7] J.B. Bell, C.N. Dawson, G.R. Shubin, An unsplit, higher order Godunov method for scalar conservation laws in multiple dimensions, J. Comput. Phys. 74 (1) (1988) 1–24, doi:10.1016/0021-9991(88)90065-4. ISSN 0021-9991. URL http://www.sciencedirect.com/science/article/B6WHY-4DD1T8P-N0/2/aba1bf519b0924a0a20968665aa37091.
[8] S. Bunya, E.J. Kubatko, J.J. Westerink, C. Dawson, A wetting and drying treatment for the Runge–Kutta discontinuous Galerkin solution to the shallow water equations, Comput. Methods Appl. Mech. Engg. 198 (17-20) (2009) 1548–1562, doi:10.1016/j.cma.2009.01.008. ISSN 0045-7825.
[9] A. Burbeau, P. Sagaut, A dynamic p-adaptive discontinuous Galerkin method for viscous flow with shocks, Comput. Fluids 34 (4-5) (2005) 401–417, doi:10.1016/j.compfluid.2003.04.002. ISSN 0045-7930.
[10] L. Demkowicz, Computing with hp-adaptive finite elements, Applied Mathematics and Nonlinear Science Series, vol. 1, Chapman & Hall/CRC, Boca Raton, FL, 2007. ISBN 978-1-58488-671-6; 1-58488-671-4. One and two dimensional elliptic and Maxwell problems, With 1 CD-ROM (UNIX).
[11] L. Demkowicz, A new discontinuous Petrov-Galerkin method with optimal test functions, Part V: solution of 1D Burgers and Navier-Stokes equations (2010) 34. http://www.ices.utexas.edu/media/reports/2010/1025.pdf.
[12] L.J. Durlofsky, B. Engquist, S. Osher, Triangle based adaptive stencils for the solution of hyperbolic conservation laws, J. Comput. Phys. 98 (1) (1992) 64–73, doi:10.1016/0021-9991(92)90173-V. ISSN 0021-9991. http://www.sciencedirect.com/science/article/B6WHY-4DD1P88-NW/2/14f5775efbf9049e31e12411e2e34238.

[13] M. Feistauer, J. Felcman, I. Straškraba, Mathematical and Computational Methods for Compressible Flow, Numerical Mathematics and Scientific Computation, Oxford University Press, 2003. ISBN 0-19-850588-4.
[14] R. Ghostine, G. Kesserwani, R. Mosé, J. Vazquez, A. Ghenaim, An improvement of classical slope limiters for high-order discontinuous Galerkin method, Int. J. Numer. Methods Fluids 59 (4) (2009) 423–442, doi:10.1002/fld.1823. ISSN 0271-2091. URL http://dx.doi.org/10.1002/fld.1823.
[15] W.F. Godoy, P.E. DesJardin, On the use of flux limiters in the discrete ordinates method for 3D radiation calculations in absorbing and scattering media, J. Comput. Phys. 229 (9) (2010) 3189–3213, doi:10.1016/j.jcp.2009.12.037. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2009.12.037.
[16] H. Hoteit, Ph. Ackerer, R. Mosé, J. Erhel, B. Philippe, New two-dimensional slope limiters for discontinuous Galerkin methods on arbitrary meshes, Int. J. Numer. Methods Eng. 61 (14) (2004) 2566–2593, doi:10.1002/nme.1172. ISSN 0029-5981. URL http://dx.doi.org/10.1002/nme.1172.
[17] L. Isoardi, G. Chiavassa, G. Ciraolo, P. Haldenwang, E. Serre, Ph. Ghendrih, Y. Sarazin, F. Schwander, P. Tamain, Penalization modeling of a limiter in the Tokamak edge plasma, J. Comput. Phys. 229 (6) (2010) 2220–2235, doi:10.1016/j.jcp.2009.11.031. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2009.11.031.
[18] C. Jin, K. Xu, A unified moving grid gas-kinetic method in Eulerian space for viscous flow computation, J. Comput. Phys. 222 (1) (2007) 155–175, doi:10.1016/j.jcp.2006.07.015. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2006.07.015.
[19] G. Kanschat, Multilevel methods for discontinuous galerkin fem on locally refined meshes, Comput. Struct. 82 (28) (2004) 2437–2445, doi:10.1016/j.compstruc.2004.04.015. ISSN 0045-7949. URL http://www.sciencedirect.com/science/article/B6V28-4DBJGG5-4/2/5cb85d27cc196137146048cd3d9d4c33. Preconditioning methods: algorithms, applications and software environments.
[20] E.J. Kubatko, J.J. Westerink, C. Dawson, hp discontinuous Galerkin methods for advection dominated problems in shallow water flow, Comput. Methods Appl. Mech. Eng. 196 (1–3) (2006) 437–451, doi:10.1016/j.cma.2006.05.002. ISSN 0045-7825. http://www.sciencedirect.com/science/article/B6V29-4M1CYTM-1/2/6c45c85d20d17690046881a795b0b04d.
[21] E.J. Kubatko, J.J. Westerink, C. Dawson, Semi discrete discontinuous Galerkin methods and stage-exceeding-order, strong-stability-preserving Runge–Kutta time discretizations, J. Comput. Phys. 222 (2) (2007) 832–848, doi:10.1016/j.jcp.2006.08.005. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2006.08.005.
[22] E.J. Kubatko, C. Dawson, J.J. Westerink, Time step restrictions for Runge–Kutta discontinuous Galerkin methods on triangular grids, J. Comput. Phys. 227 (23) (2008) 9697–9710, doi:10.1016/j.jcp.2008.07.026. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2008.07.026.
[23] E.J. Kubatko, S. Bunya, C. Dawson, J.J. Westerink, Dynamic *p*-adaptive Runge–Kutta discontinuous Galerkin methods for the shallow water equations, Comput. Methods Appl. Mech. Eng. 198 (21-26) (2009) 1766–1774, doi:10.1016/j.cma.2009.01.007. ISSN 0045-7825). URL http://www.sciencedirect.com/science/article/B6V29-4VDY7X4-1/2/36e49328fea4e4f751d689510b7e3b3f. Advances in Simulation-Based Engineering Sciences - Honoring J. Tinsley Oden.
[24] E.J. Kubatko, S. Bunya, C. Dawson, J.J. Westerink, C. Mirabito, A performance comparison of continuous and discontinuous finite element shallow water models, J. Sci. Comput. 40 (1-3) (2009) 315–339, doi:10.1007/s10915-009-9268-2. ISSN 0885-7474. URL http://dx.doi.org/10.1007/s10915-009-9268-2.
[25] D. Kuzmin, A vertex-based hierarchical slope limiter for *p*-adaptive discontinuous Galerkin methods, J. Comput. Appl. Math. 233 (12) (2010) 3077–3085. ISSN 0377-0427, doi: http://dx.doi.org/10.1016/j.cam.2009.05.028.
[26] D. Levy, C.-W. Shu, J. Yan, Local discontinuous Galerkin methods for nonlinear dispersive equations, J. Comput. Phys. 196 (2) (2004) 751–772. ISSN 0021-9991.
[27] Y. Liu, C.-W. Shu, E. Tadmor, M. Zhang, Central discontinuous Galerkin methods on overlapping cells with a nonoscillatory hierarchical reconstruction, SIAM J. Numer. Anal. 45 (6) (2007) 2442–2467, doi:10.1137/060666974 (electronic) ISSN 0036-1429. URL http://dx.doi.org/10.1137/060666974.
[28] H. Luo, J. Baum, R. Lhner, A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids, J. Comput. Phys. 227 (20) (2008) 8875–8893, doi:10.1016/j.jcp.2008.06.035. ISSN 0021-9991. URL http://www.sciencedirect.com/science/article/B6WHY-4T13CS3-2/2/cacbb700cf043776d94ac1bd3a985bed.
[29] C. Michoski, J.A. Evans, P.G. Schmitz, A. Vasseur, Quantum hydrodynamics with trajectories: the nonlinear conservation form mixed/discontinuous Galerkin method with applications in chemistry, J. Comput. Phys. 228 (23) (2009) 8589–8608, doi:10.1016/j.jcp.2009.08.011. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2009.08.011.
[30] C. Michoski, J.A. Evans, P.G. Schmitz, Multiscale discontinuous Galerkin hp-adaptive chemical reacters I: quiescent reactors, preprint, 2011a.
[31] C. Michoski, J.A. Evans, P.G. Schmitz, A. Vasseur, A discontinuous Galerkin method for viscous compressible multifluids, J. Comput. Phys. 229 (6) (2010) 2249–2266, doi:10.1016/j.jcp.2009.11.033. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2009.11.033.
[32] J. Murillo, P. García-Navarro, J. Burguete, Conservative numerical simulation of multi-component transport in two-dimensional unsteady shallow water flow, J. Comput. Phys. 228 (15) (2009) 5539–5573, doi:10.1016/j.jcp.2009.04.039. ISSN 0021-9991.URL http://dx.doi.org/10.1016/j.jcp.2009.04.039.
[33] J. Palaniappan, S.T. Miller, R.B. Haber, Sub-cell shock capturing and spacetime discontinuity tracking for nonlinear conservation laws, Int. J. Numer. Methods Fluids 57 (9) (2008) 1115–1135, doi:10.1002/fld.1850. ISSN 0271-2091. URL http://dx.doi.org/10.1002/fld.1850.
[34] P.P. Persson, J. Peraire, Sub-cell shock capturing for discontinuous Galerkin methods, Forty-fourth AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA., Online: 5–18, 2006.
[35] J.X. Qiu, C.W. Shu, Runge–Kutta discontinuous Galerkin method using WENO limiters, Siam J. Sci. Comput. 26 (3) (2005) 907–929.
[36] S.J. Ruuth, Global optimization of explicit strong-stability-preserving Runge–Kutta methods, Math. Comp. 75 (253) (2006) 183–207, doi:10.1090/S0025-5718-05-01772-2 (electronic) ISSN 0025-5718. URL http://dx.doi.org/10.1090/S0025-5718-05-01772-2.
[37] C.-W. Shu, S. Osher, Efficient implementation of essentially nonoscillatory shock-capturing schemes, J. Comput. Phys. 77 (2) (1988) 439–471. ISSN 0021-9991.
[38] J.W. Thomas, Numerical partial differential equations: finite difference methods, Texts in Applied Mathematics, vol. 22, Springer-Verlag, New York, 1995. ISBN 0-387-97999-9.
[39] G. Tóth, Y. Ma, T.I. Gombosi, Hall magnetohydrodynamics on block-adaptive grids, J. Comput. Phys. 227 (14) (2008) 6967–6984, doi:10.1016/j.jcp.2008.04.010. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2008.04.010.
[40] B. van Leer, Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method [J. Comput. Phys. 32(1) (1979) 101–136], J. Comput. Phys. 135 (2) (1997) 227–248. ISSN 0021-9991. With an introduction by Ch. Hirsch, Commemoration of the 30th anniversary {of J. Comput. Phys.}.
[41] L. Wang, D.J. Mavriplis, Adjoint-based hp adaptive discontinuous Galerkin methods for the 2D compressible Euler equations, J. Comput. Phys. 228 (20) (2009) 7643–7661, doi:10.1016/j.jcp.2009.07.012. ISSN 0021-9991. URL http://dx.doi.org/10.1016/j.jcp.2009.07.012.
[42] G.M. Ward, D.I. Pullin, A hybrid, center-difference, limiter method for simulations of compressible multicomponent flows with Mie-Grüneisen equation of state, J. Comput. Phys. 229 (8) (2010) 2999–3018, doi:10.1016/j.jcp.2009.12.027. ISSN 0021-9991 URL http://dx.doi.org/10.1016/j.jcp.2009.12.027.
[43] Z. Xu, Y. Liu, C.W. Shu, Hierarchical reconstruction for discontinuous Galerkin methods on unstructured grids with a WENO-type linear reconstruction and partial neighboring cells, Comput. Phys. 228 (2009) 194–2212. doi: http://dx.doi.org/10.1016/j.jcp.2008.11.025. URL http://dx.doi.org/10.1016/j.jcp.2008.11.025 (ISSN 0021-9991).
[44] J. Zhu, J.X. Qiu, C.W. Shu, M. Dumbser, Runge-kutta discontinuous Galerkin method using WENO limiters II: Unstructured meshes, J. Comput. Phys. 227 (9) (2008) 4330–4353.