Singularity-free evaluation of collapsed-coordinate orthogonal polynomials

Robert C. Kirby
Texas Tech University

The orthogonal polynomials used in finite and spectral element methods for nonrectangular elements may be defined in terms of *collapsed* coordinates, wherein the shapes are mapped to a square or cube by means of a singular change of variables. The orthogonal basis is a product of specific Jacobi polynomials in these new coordinates. However, implementations of these polynomials require special handling of the singularities in the coordinates. We derive new recurrence relations for these polynomials on triangles, tetrahedra, pyramids, and prisms that work directly in the original coordinates, avoiding any special treatment of singular points and requiring constant auxilliary storage with respect to the polynomial degree. These recurrences are seen to speed up both symbolic and numerical computation of the orthogonal polynomials.

Categories and Subject Descriptors: G.1 [**Numerical analysis**]: —; G.4 [**Mathematical software**]: —*Algorithm Design, Efficiency*; I.1 [**Symbolic and Algebraic Manipulation**]: —

General Terms: Algorithms, performance

Additional Key Words and Phrases: Orthogonal polynomial, recurrence relation, nonrectangular domain

1. INTRODUCTION

Orthogonal polynomials on nonrectangular domains play an important role in certain kinds of spectral and finite element computation. Many discontinuous Galerkin methods [Castillo 2002] are implemented in terms of orthogonal polynomials. They also are an important building block for the nodal spectral elements developed by Hesthaven and Warburton [Hesthaven and Warburton 2002] and for tools for constructing general finite elements such as the FIAT project by the author [Kirby 2004; 2006] and the FEMSTER project by Castillo, Rieben, and White [Castillo et al. 2005].

While orthogonal polynomals are constructed rectangular domains allow orthogonal bases to be constructed by tensor products of Legendre or other polynomials in each coordinate direction, nonrectangular domains require more care. An orthogonal basis on triangles was developed by authors such as Koornwinder [Koornwinder 1975] and Dubiner [Dubiner 1991] (see [Braess 2005] for a survey). These polynomials are the eigenfunctions of a particular Sturm-Liouville problem, and also

Author's address: Robert Kirby; Department of Mathematics and Statistics; Texas Tech University; Lubbock, TX 79409; *Email:*`robert.c.kirby@ttu.edu`. This work was supported by the United States Department of Energy Office of Science under award number DE-FG02-04ER2582.

may constructed by transforming the triangle to a square by a special coordinate mapping. Tensor products of Jacobi polynomials with certain weights are formed in these coordinates, and these are L^2-orthogonal in the original triangle. The book of Karniadakis and Sherwin [Karniadakis and Sherwin 2005] contains a thorough presentation of these results, as well as orthogonal bases on tetrahedra, pyrmaids and prisms.

Recent work by Beuchler and Schöbel [Beuchler and Schöberl 2006] has similar flavor, deriving shape functions for p-finite elements that lead to very sparse element stiffness matrices. They also work in terms of transformed coordinates, using products of integrated Jacobi polynomials, but also force their functions to obey certain boundary properties. This work has been extended to tetrahedra by Beuchler and Pillwein [Beuchler and Pillwein 2007].

The present work does not use these special polynomials directly in finite element computation, but rather uses as a building block to construct a wide range of different finite element bases in the FIAT project []. Here, the essential properties of the orthogonal bases are their conditioning (useful in solving generalized Vandermonde systems) and their hierarchical nature. Jacobi recurrence relations allow stable evaluation to very high order, so they are very flexible tools.

However, the coordinate singularities in the standard representation in collapsed coordinates makes these polynomials difficult to evaluate and especially differentiate at certain points in the domain that turn out to be essential for certain kinds of finite elements. In this paper, we develop with new recurrences for computing the orthogonal bases that avoid the difficulties of the collapsed coordinate systems and hence are more suitable as a general building block for finite element bases as done in the FIAT project []. The new recurrences work directly in the untransformed coordinates; there is no appearance of the coordinate singularity. This has advantages in numerical evaluation of the polynomials, avoiding the need to handle coordinate singularities by special cases in the code and also reducing the amount of auxilliary storage needed to tabulate the polynomials. Avoiding the coordinate singularity greatly simplifies the construction of certain finite element bases, such as those of Hermite and Argyris [], via generalized Vandermonde matrices. After reviewing the general paradigm employed by FIAT in Section 0**??**, we survey some necessary facts about Jacobi polynomials in Section 3, we describe the orthogonal polynomials on triangles and derive new recurrence relations in Section 4. Then, Section 5 extends these ideas to tetrahedra. We derive similar relations for prisms and pyramids in Section 6. We discuss some of the implementation advantages and timing results in Section 0**??**.

2. THE FIAT PARADIGM

In [], the Dubiner polynomials are used as building blocks for construcing general Ciarlet-type finite elements.

Definition 2.1 *Ciarlet[].* A *finite element* is a triple (K, P, N)

Definition 2.2. The *nodal basis* for a finite element (K, P, N) is the set $\{\psi_i\}_{i=1}^{\dim P}$ such that

$$n_i(\psi_j) = \delta_{i,j}, 1 \leq i, j \leq \dim P. \tag{1}$$

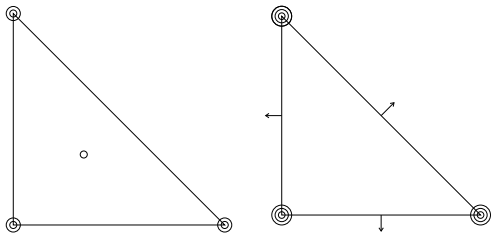

Fig. 1. Degrees of freedom for the cubic Hermite (left) and quintic Argyris (right) elements, which require evaluation and differentiation at the top vertex.

It is not hard to show that if $\{\phi_i\}_{i=1}^{\dim P}$ is a basis for P, then the nodal basis may be obtained by

$$\psi_i = V_{ij}^{-1}\phi_j, \tag{2}$$

where V is the generalized Vandermonde matrix

$$V_{ij} = n_i(\phi_j). \tag{3}$$

The FIAT project, motivated by the fact that many elements appear far more frequently in finite element texts and papers than in codes, makes use of this approach to generate arbitrary order bases for a wide range of elements that are typically perceived as "hard" to implement.

For several reasons, the Dubiner basis and its extensions to higher dimensions are an attractive choice for $\{\phi_i\}_{i=1}^{\dim P}$. Being based on Jacobi polynomials, they may be stably and accurately evaluated to quite high degree via simple recurrence relations. Moreover, their L^2 orthogonality tends to lead to well-conditioned V matrices. Also, their hierarchical nature simplifies the construction of certain finite element spaces, such as the Raviart-Thomas space [] over other well-conditioned bases like Bernstein polynomials.

While a very wide range of elements are constructed using this paradigm in the FIAT project, certain classical elements present special difficulties for the standard method of evaluating the Dubiner basis. Consider the cubic Hermite simplex [**?**], in which $P = P_3(K)$ is the space of cubic polynomials. The nodes consist of point values and gradients at the vertices and the point value at the barycenter of K, as shown in Figure 2.

Constructing V in this situation requires differentiating each ϕ_i at the top vertex, exactly the point at which the Dubiner formulae fail. A similar issue occurs for the Hermite tetrahedron. While it is possible but tedious to include analytic derivatives for that single point, and the situation becomes even more so for the Argyris triangle. The polynomial space $P = P_5(K)$ consists of quintic polynomials, and the nodes contain point values along with first and *second* derivatives at each vertex, together with the normal derivatives normal at each edge midpoint, as shown in Figure 2.

The general-purpose paradigm of the FIAT project suggests that it is desirable to treat all the points in the reference domain uniformly. For a particular, fixed basis such as in a spectral element code, the issue may be circumvented by choosing quadrature rules that avoid the singular points altogether, but FIAT cannot make

such *a priori* assumptions if it is to be general.

3. PRELIMINARY FACTS ABOUT JACOBI POLYNOMIALS

Before developing the new recurrence relations, we recall some basic facts about Jacobi polynomials that may be found in a reference such as Abramowitz and Stegun [Abramowitz and Stegun 1964]. We denote the Jacobi polynomials by $P_n^{\alpha,\beta}(x)$. The first two Jacobi polynomials, in terms of α and β are

$$\begin{aligned} P_0^{\alpha,\beta} &= 1 \\ P_1^{\alpha,\beta} &= \frac{1}{2}\left(\alpha - \beta + (\alpha + \beta + 2)\, x\right). \end{aligned} \tag{4}$$

For $n \geq 2$, the three-term recurrence relations

$$P_{n+1}^{\alpha,\beta}(x) = (a_n^{\alpha,\beta} x + b_n^{\alpha,\beta}) P_n^{\alpha,\beta}(x) - c_n^{\alpha,\beta} P_{n-1}^{\alpha,\beta}(x), \tag{5}$$

hold, where the recurrence coefficients are given by

$$\begin{aligned} a_n^{\alpha,\beta} &= \frac{(2n+1+\alpha+\beta)(2n+2+\alpha+\beta)}{2(n+1)(n+1+\alpha+\beta)} \\ b_n^{\alpha,\beta} &= \frac{(\alpha^2-\beta^2)(2n+1+\alpha+\beta)}{2(n+1)(2n+\alpha+\beta)(n+1+\alpha+\beta)} \\ c_n^{\alpha,\beta} &= \frac{(n+\alpha)(n+\beta)(2n+2+\alpha+\beta)}{(n+1)(n+1+\alpha+\beta)(2n+\alpha+\beta)}. \end{aligned} \tag{6}$$

As special cases needed later, the recurrence coefficients when $\alpha = \beta = 0$ are

$$\begin{aligned} a_n^{0,0} &= \frac{2n+1}{n+1} \\ b_n^{0,0} &= 0 \\ c_n^{0,0} &= \frac{n}{n+1}. \end{aligned} \tag{7}$$

4. THE ORTHOGONAL BASIS ON TRIANGLES

In spectral element literature, the triangular basis is typically constructed by mapping the triangle with vertices $(-1,-1)$, $(1,-1)$, $(-1,1)$ to the square $[-1,1]^2$ via the transformation

$$\begin{aligned} \eta_1(x,y) &= 2\left(\frac{1+x}{1-y}\right) - 1, \\ \eta_2(x,y) &= y. \end{aligned} \tag{8}$$

This maps the reference triangle to the reference square, as shown in Figure 2. The singularity at $y \to 1$ is handled by defining $\eta_1(x,1)$ to be -1. To remain inside the reference triangle, x must approach -1 as y approaches 1, making η_1 continuous.

The polynomials are given in terms of the η_1, η_2 coordinates by

$$D^{p,q}(x,y) = P_p^{0,0}(\eta_1)\left(\frac{1-\eta_2}{2}\right)^p P_q^{2p+1,0}(\eta_2). \tag{9}$$

Despite the $\frac{1}{1-y}$ appearing η_1, $D^{p,q}$ is in fact a polynomial both in η_1, η_2 and in x, y. To see this, note that $P_p^{0,0}(\eta_1)$ is a polynomial of degree p in η_1, which means

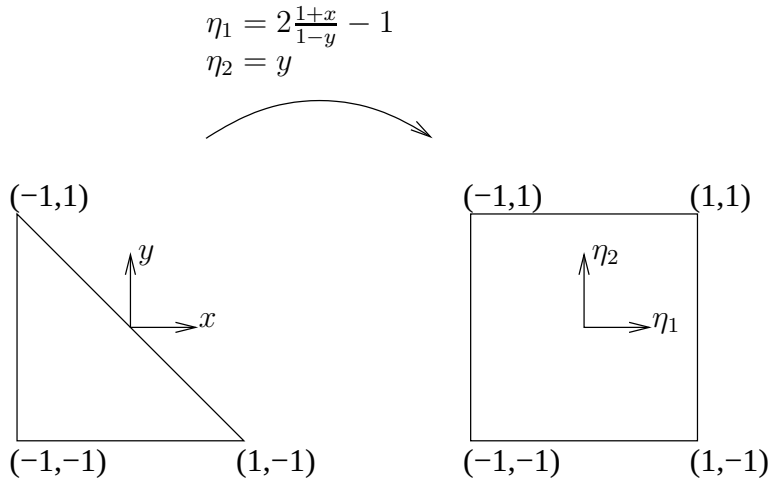

Fig. 2. Reference triangular and square domains with collapsed coordinate transformation.

that it contains a term $(1-y)^{-p}$ in it. By multiplying through by $(1-\eta_2)^p = (1-y)^p$, all of the negative powers will be cancelled.

Instead of the form (9), $D^{p,q}(x,y)$ may instead be written as

$$D^{p,q}(x,y) = \chi^p(x,y)\psi^{p,q}(y), \tag{10}$$

where

$$\begin{aligned} \chi^p(x,y) &= P_p^{0,0}(\eta_1)\left(\frac{1-\eta_2}{2}\right)^p, \\ \psi^{p,q}(y) &= P_q^{2p+1,0}(\eta_2) = P_q^{2p+1,0}(y). \end{aligned} \tag{11}$$

The polynomial $\chi^p(x,y)$ has two arguments but only one index, while $\psi^{p,q}(y)$ has two indices and only one argument. This will be important in establishing the sum-factorization property later.

The more standard formation (such as in [Karniadakis and Sherwin 2005]) of $D^{p,q}(x,y)$ is a separable representation in (η_1,η_2)

$$D^{p,q}(x,y) = \tilde{\psi}_p^a(\eta_1)\tilde{\psi}_{p,q}^b(\eta_2), \tag{12}$$

where

$$\begin{aligned} \tilde{\psi}_p^a(\eta_1) &= P_p^{0,0}(\eta_1), \\ \tilde{\psi}_{p,q}^b(\eta_2) &= \left(\frac{1-\eta_2}{2}\right)^p P_q^{2p+1,0}(\eta_2). \end{aligned} \tag{13}$$

The separability of $D^{p,q}$ in the transformed coordinates is the basis for efficient spectral element algorithms in unstructured coordinates. In the context of the FIAT project, however, the new decomposition (10), has some particular advantages.

If the Dubiner polynomials are computed symbolically, for example, using the standard representation, then the computer must simplify a rational function to a polynomial by cancelling the powers of $1-y$ appearing in both numerators and

denominators. This can be quite expensive in many symbolic packages. Moreover, the coordinate singularity presents limitations in a numerical context. In spectral element methods, the quadrature points at which the basis is evaluated and differentiated may be selected to bypass the singular points (e.g. Radau quadrature). On the other hand, using the Dubiner basis as a building block for other bases as in the FIAT project in principle can require evaluation and differentiation at any point. For example, constructing V for the Hermite and Argyris elements requires derivatives at the top vertex of the triangle, which requires delicate treatment in the collapsed coordinates.

Before presenting the recurrence relations, it is important to point out that although the representation is non-separable, the structure still admits a sum-factorization that admits inner product evaluation of the same complexity as the separated representation. While the focus of this paper is on the flexibility of the new representation and its application to FIAT, the sum-factored algorithm will be presented for completeness later.

It is straightforward to derive recurrence relations for the χ^p and $\psi^{p,q}$ factors comprising $D^{p,q}$

PROPOSITION 4.1. *With*

$$\begin{aligned} \chi^0(x,y) &= 1, \\ \chi^1(x,y) &= \frac{1+2x+y}{2}, \end{aligned} \tag{14}$$

for all $p \geq 1$, the functions $\chi^p(x,y)$ satisfy the three-term recurrence relation

$$\chi^{p+1}(x,y) = \frac{2p+1}{p+1}\left(\frac{1+2x+y}{2}\right)\chi^p(x,y) - \frac{p}{p+1}\left(\frac{1-y}{2}\right)^2\chi^{p-1}(x,y). \tag{15}$$

PROOF. The proof follows from using the standard Legendre recurrence with simple algebraic manipulations on the factors involving coordinates. □

Since $D^{p,0}(x,y) = \chi^p(x,y)$, this simple recurrence calculates a whole swatch of of the Dubiner polynomials. Moreover, $D^{p,1}(x,y) = \chi^p(x,y)P_1^{2p+1,0}(y)$, so that $D^{p,1}(x,y)$ may be obtained from $D^{p,0}(x,y)$ by multiplying by the linear Jacobi polynomial with indices $2p+1, 0$:

$$\begin{aligned} D^{p,1}(x,y) &= D^{p,0}(x,y)P_1^{2p+1,0}(y) \\ &= \frac{1}{2}D^{p,0}(x,y)\left(2p+1+(2p+3)\,y\right). \end{aligned} \tag{16}$$

Finally, for each p, a three-term recurrence in q follows straightforwardly from the Jacobi recurrence:

PROPOSITION 4.2. *For each p, the orthogonal polynomials satisfy a recurrence relation in the second index. For $q \geq 1$,*

$$D^{p,q+1}(x,y) = (a_q^{2p+1,0}y + b_q^{2p+1,0})D^{p,q}(x,y) - c_q^{2p+1,0}D^{p,q-1}(x,y). \tag{17}$$

These recurrences are assembled together in Algorithm 1, which for inputs x, y and the degree d, calcuates all of the Dubiner polynomials. Since the collapsed coordinate singularity never appears, this algorithm may be used for any x, y in the

reference triangle and may be differentiated (by hand or automatic differentiation) to obtain as many derivatives as desired.

Algorithm 1 Computes all triangular orthogonal polynomials by recurrence

$D^{0,0}(x,y) := 1$
$D^{1,0}(x,y) := \frac{1+2x+y}{2}$
for $p \leftarrow 1, d-1$ **do**
  $D^{p+1,0}(x,y) := \left(\frac{2p+1}{p+1}\right)\left(\frac{1+2x+y}{2}\right) D^{p,0}(x,y) - \left(\frac{p}{p+1}\right)\left(\frac{1-y}{2}\right)^2 D^{p-1,0}(x,y)$
end for
for $p \leftarrow 0, d-1$ **do**
  $D^{p,1}(x,y) := D^{p,0}(x,y)\left(\frac{1+2p+(3+2p)y}{2}\right)$
end for
for $p \leftarrow 0, d-1$ **do**
  for $q \leftarrow 1, d-p-1$ **do**
    $D^{p,q+1}(x,y) := \left(a_q^{2p+1,0} y + b_q^{2p+1,0}\right) D^{p,q}(x,y) - c_q^{2p+1,0} D^{p,q-1}(x,y)$
  end for
end for

5. TETRAHEDRAL EXPANSION

The tetrahedral polynomials may be defined on the tetrahedron with vertices $(-1,-1,-1)$, $(1,-1,-1)$, $(-1,1,-1)$,$(-1,-1,1)$ by mapping to the cube $[-1,1]^3$ using the coordinate transformation

$$\begin{aligned} \eta_1(x,y,z) &= \frac{-2(1+x)}{y+z} - 1 \\ \eta_2(x,y,z) &= \frac{2(1+y)}{1-z} - 1 \\ \eta_3(x,y,z) &= z \end{aligned} \tag{18}$$

As with the triangular case, we define η_1 along the line $y+z =$ and η_2 along $z = 1$ to make the functions continuous. The reference tetrahedron and cube are shown in Figure 3.

The tetrahedral polynomials [Karniadakis and Sherwin 2005] are given by

$$D^{p,q,r}(x,y,z) = P_p^{0,0}(\eta_1)\left(\frac{1-\eta_2}{2}\right)^p P_q^{2p+1,0}(\eta_2)\left(\frac{1-\eta_3}{2}\right)^{p+q} P_r^{2p+2q+2,0}(\eta_3), \tag{19}$$

where η_1, η_2, η_3 are functions of x, y, z given by (18). These polynomials may be written in separated form as

$$D^{p,q,r}(x,y,z) = \tilde{\psi}_p^a(\eta_1)\tilde{\psi}_{p,q}^b(\eta_2)\tilde{\psi}_{p,q,r}^c(\eta_3), \tag{20}$$

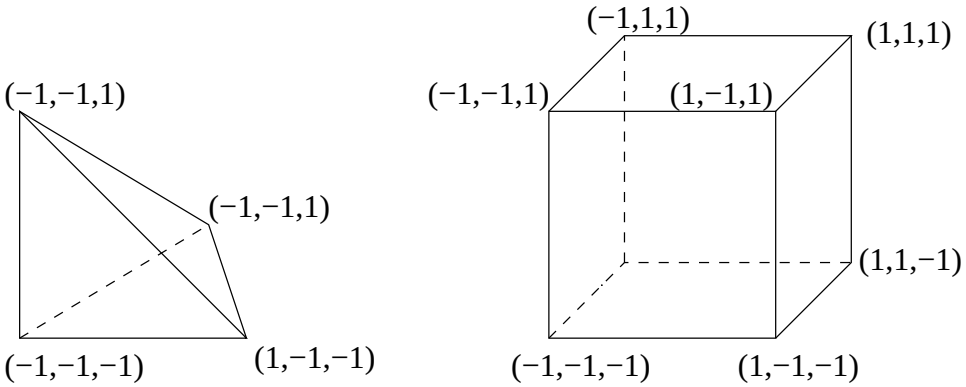

Fig. 3. Reference tetrahedron and cube.

where

$$\begin{aligned}\tilde{\psi}^a_p(z) &= P^{0,0}_p(z),\\ \tilde{\psi}^b_{p,q}(z) &= \left(\frac{1-z}{2}\right)^p P^{2p+1,0}_q(z),\\ \tilde{\psi}^c_{p,q,r}(z) &= \left(\frac{1-z}{2}\right)^{p+q} P^{2p+2q+2,0}_r(z).\end{aligned} \tag{21}$$

As in the triangular case, this separated representation is useful in turning integrals over the tetrahedron into products of one-dimensional integrals on the cube. However, $\tilde{\psi}^a_p$ and $\tilde{\psi}^{a,b}_q$ are both rational functions of the Cartesian coordinates, leading to similar complications in evaluating and differentiating.

We define each polynomial to be the product of three simpler polynomials in the standard (non-collapsed) coordinates by

$$D^{p,q,r}(x,y) = \chi^p(x,y,z)\psi^{p,q}(y,z)\omega^{p,q,r}(z), \tag{22}$$

where

$$\begin{aligned}\chi^p(x,y,z) &= P^{0,0}_p(\eta_1)\left(\frac{1-\eta_2}{2}\right)^p\left(\frac{1-\eta_3}{2}\right)^p,\\ \psi^{p,q}(y,z) &= P^{2p+1,0}_q(\eta_2)\left(\frac{1-\eta_3}{2}\right)^q,\\ \omega^{p,q,r}(z) &= P^{2p+2q+2,0}_r(\eta_3).\end{aligned} \tag{23}$$

It is not hard to see that $\chi^p(x,y,z)$ is polynomial of degree p in the variables x,y,z, $\psi^{p,q}(y,z)$ is a polynomial of degree q in y,z, and $\omega^{p,q,r}(z)$ is a polynomial of degree r in z alone, giving $D^{p,q,r}(x,y,z)$ as polynomial of degree $p+q+r$. The set of polynomials over the reference tetrahedron with degree no greater than d is given by

$$P_d = \{D^{p,q,r}(x,y,z) : 0 \le p,q,r, p+q+r \le d\} \tag{24}$$

As for the triangular basis, recurrence relations may be derived directly in the

(x, y, z) coordinates. First, with $q = r = 0$, the Legendre recurrence relations and some simple algebraic manipulations give rise to a recurrence in p.

PROPOSITION 5.1. *Starting with*

$$\begin{aligned} \chi^0(x,y,z) &= 1 \\ \chi^0(x,y,z) &= \frac{2+2x+y+z}{2}, \end{aligned} \tag{25}$$

the tetrahedral expansion functions satisfy

$$\begin{aligned} D^{p+1,0,0}(x,y,z) &= \chi^{p+1}(x,y,z) \\ &= \left(\frac{2p+1}{p+1}\right)\left(\frac{2x+2+y+z}{2}\right)\chi^p(x,y,z) \\ &\quad - \left(\frac{p}{p+1}\right)\left(\frac{y+z}{2}\right)^2\chi^{p-1}(x,y,z) \\ &= \left(\frac{2p+1}{p+1}\right)\left(\frac{2x+2+y+z}{2}\right)D^{p,0,0}(x,y,z) \\ &\quad - \left(\frac{p}{p+1}\right)\left(\frac{y+z}{2}\right)^2 D^{p-1,0,0}(x,y,z) \end{aligned} \tag{26}$$

Now, with all $D^{p,0,0}$ evaluated, it is simple to compute $D^{p,1,0}(x, y, z)$ by multiplying a linear polynomial:

$$\begin{aligned} D^{p,1,0}(x,y,z) &= \chi^p(x,y,z)\psi^{p,1}(y,z) \\ &= \chi^p(x,y,z)\left(\frac{1-\eta_3}{2}\right)P_1^{2p+1,0}(\eta_2) \\ &= \chi^p(x,y,z)\left(\frac{1+2p+(3+2p)\eta_2}{2}\right)\left(\frac{1-z}{2}\right) \\ &= \chi^p(x,y,z)\left(p(1+y)+\frac{2+3y+z}{2}\right) \end{aligned} \tag{27}$$

Now, the Jacobi recurrence relations may be used to derive a recurrence for general q with $r = 0$.

PROPOSITION 5.2. *For $q \geq 1$, the functions $D^{p,q,0}(x, y, z)$ satisfy the three-term recurrence*

$$D^{p,q+1,0}(x,y,z) = \alpha_q^{2p+1,0}D^{p,q,0}(x,y) - \gamma_q^{2p+1,0}D^{p,q-1,0}(x,y), \tag{28}$$

where

$$\begin{aligned} \alpha_q^{2p+1,0} &= a_q^{2p+1,0}\left(\frac{1+2y+z}{2}\right) + b_q^{2p+1,0}\left(\frac{1-z}{2}\right) \\ \gamma_q^{2p+1,0} &= c_q^{2p+1,0}\left(\frac{1-z}{2}\right)^2 \end{aligned} \tag{29}$$

After computing $D^{p,q,0}$ for all necessary p, q, the polynomials $D^{p,q,1}$ may be

computed by multiplying by a linear Jacobi polynomial:

$$\begin{aligned} D^{p,q,1}(x,y,z) &= D^{p,q,0}(x,y,z)P_1^{2p+2q+2,0}(z) \\ &= D^{p,q,0}(x,y,z)\left(1+p+q+(2+q+p)z\right). \end{aligned} \tag{30}$$

Using the Jacobi recurrence relations in r gives rise to the final recurrence

PROPOSITION 5.3. *For all $p,q \geq 0$ and $r \geq 1$, the polynomials $D^{p,q,r}(x,y,z)$ satisfy the three-term recurrence*

$$\begin{aligned} D^{p,q,r+1}(x,y,z) = &\left(a_r^{2p+2q+2,0}z + b_r^{2p+2q+2,0}\right) D^{p,q,r}(x,y,z) \\ &- c_r^{2p+2q+2,0} D^{p,q,r-1}(x,y,z). \end{aligned} \tag{31}$$

These results are collected in Algorithm 2 for tabulating the orthogonal tetrahedral expansion. As in the triangular case, differentiating manually or by AD is straightforward.

6. OTHER THREE-DIMENSIONAL SHAPES

Besides cubes (with tensor products of Legendre polynomials), orthogonal polynomial bases are known on prisms and pyramids.

Consider the reference prism shown in Figure 4, taking the triangle with vertices $(-1,-1)$,$(1,-1)$, and $(-1,1)$ in the $x-y$ plane and extending it in the z axis from $z=-1$ to $z=1$. The standard polynomial space on this shape is obtained by taking polynomials of total degree d in x,y times polynomials of degree d in z. An orthogonal basis for this space is easily obtained by introducing

$$\phi^{p,q,r}(x,y,z) = D^{p,q}(x,y)P_r^{0,0}(z), \tag{32}$$

where $D^{p,q}$ are the orthogonal polynomials constructed above. Tensor products of $D^{p,q}(x,y)$ may be used along with the Legendre polynomials in z to compute these polynomials. The polynomial space of degree d is then spanned by

$$\left\{\phi^{p,q,r}(x,y,z) : 0 \leq p,q,r, \quad p+q \leq d, 0 \leq r \leq d\right\}. \tag{33}$$

Prisms hence require no new recurrence relations, simply using the results of Section 4 and taking tensor products with the Legendre polynomials in z.

Pyramids do not have the same tensor-product structure, but the same techniques developed for triangles and tetrahedra may still be applied to get singularity-free recurrences.

Define the reference pyramid, shown in Figure 5 by

$$\left\{x,y,z : -1 \leq x,y,z \leq 1, x+z \leq 1, y+z \leq 1\right\}, \tag{34}$$

as shown in Figure 5.

This pyramid is mapped to $[-1,1]^3$ via the transformation

$$\begin{aligned} \bar{\eta}_1 &= 2\frac{1+x}{1-z} - 1 \\ \eta_2 &= 2\frac{1+y}{1-z} - 1 \\ \eta_3 &= z \end{aligned} \tag{35}$$

Algorithm 2 Computes all tetrahedral polynomials by recurrence

$F_1 := \frac{2+2x+y+z}{2}$
$F_2 := \left(\frac{y+z}{2}\right)^2$
$F_3 := \frac{2+3y+z}{2}$
$F_4 := \frac{1+2y+z}{2}$
$F_5 := \frac{1-z}{2}$
$D^{0,0,0} := 1$
$D^{1,0,0} := F_1$
for $p \leftarrow 1,\ d-1$ **do**
    $D^{p+1,0,0} := \left(\frac{2p+1}{p+1}\right) F_1 D^{p,0,0} - \left(\frac{p}{p+1}\right) F_2 D^{p-1,0,0}$
end for
for $p \leftarrow 0,d-1$ **do**
    $D^{p,1,0} := (p(1+y)+F_3)D^{p,0,0}$
end for
for $p \leftarrow 0,d-2$ **do**
    for $q \leftarrow 1,d-p-1$ **do**
        $D^{p,q+1,0} := \left(a_q^{2p+1,0}F_4 + b_q^{2p+1,0}F_5\right) D^{p,q,0} - c_q^{2p+1,0}\left(F_5\right)^2 D^{p,q-1,0}$
    end for
end for
for $p \leftarrow 0,d-1$ **do**
    for $q \leftarrow 0,d-p-1$ **do**
        $D^{p,q,1} := (1+p+q+(2+q+p)z)D^{p,q,0}$
    end for
end for
for $p \leftarrow 0,d-2$ **do**
    for $q \leftarrow 0,d-p-2$ **do**
        for $r \leftarrow 1,d-p-q-1$ **do**
            $D^{p,q,r+1} := \left(a_r^{2p+2q+2,0}z + b_r^{2p+2q+2,0}\right) D^{p,q,r} - c_r^{2p+2q+2,0} D^{p,q,r-1}$
        end for
    end for
end for

as described in [Karniadakis and Sherwin 2005]. The pyramidic orthogonal polynomials are represented separably on the cube by

$$\phi^{p,q,r}(x,y,z) = \tilde{\psi}_p^a(\bar{\eta}_1)\tilde{\psi}_q^a(\eta_2)\tilde{\psi}^c(\eta_3), \tag{36}$$

where $\tilde{\psi}^a$ and $\tilde{\psi}^c$ are as defined in (21). The terms may be regrouped into polynomials in the pyramidic coordinates by

$$\phi^{p,q,r}(x,y,z) = P_p^{0,0}(\bar{\eta}_1)\left(\frac{1-z}{2}\right)^p P_q^{0,0}(\eta_2)\left(\frac{1-z}{2}\right)^q P_r^{2p+2q+2,0}(z). \tag{37}$$

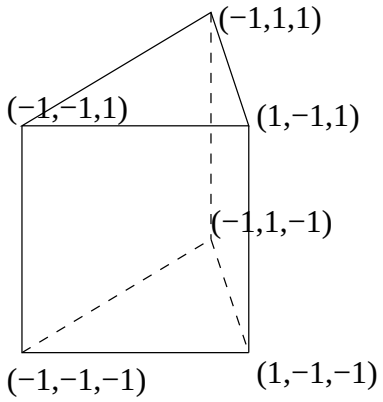

Fig. 4. Reference prism and its coordinates

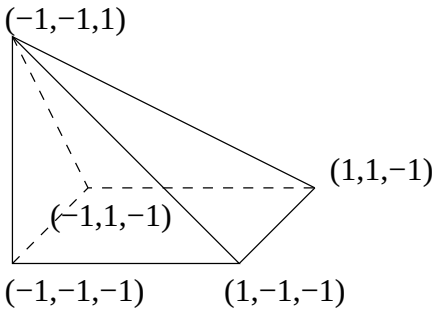

Fig. 5. Reference prism and its coordinates

By defining

$$\begin{aligned}\chi_1^p(x,z) &= P_p^{0,0}(\bar{\eta}_1)\left(\frac{1-z}{2}\right)^p,\\ \chi_2^q(y,z) &= P_q^{0,0}(\eta_2)\left(\frac{1-z}{2}\right)^q,\\ \omega^{p,q,r}(z) &= P_r^{2p+2q+2,0}(z),\end{aligned} \tag{38}$$

we obtain a decomposition

$$\phi^{p,q,r}(x,y,z) = \chi_1^p(x,y)\chi_2^q(y,z)\omega^{p,q,r}(z) \tag{39}$$

into polynomials amenable to recurrence relations. The natural space of polynomials is spanned by the orthogonal set

$$\left\{\phi^{p,q,r}(x,y,z) : 0 \le p,q \le d, p+q+r \le d\right\}. \tag{40}$$

The recurrence relations for χ_1^p and χ_2^p are derived in exactly the same way as for the χ^p defined for triangular and tetrahedral expansions.

PROPOSITION 6.1. *Starting from*

$$\begin{aligned}\chi_1^0(x,y) &= 1,\\ \chi_1^1(x,y) &= \frac{1+2x+z}{2},\end{aligned} \tag{41}$$

there holds the three-term recurrence relation

$$\begin{aligned}\chi_1^p(x,z) = &\left(\frac{2p+1}{p+1}\right)\left(\frac{1+2x+z}{2}\right)\chi_1^p(x,z)\\ &-\left(\frac{p}{p+1}\right)\left(\frac{1-z}{2}\right)^2\chi_1^{p-1}(x,z)\end{aligned} \tag{42}$$

This allows computation of $\phi^{p,0,0}(x,y,z)$ for all p. Just as before, $\phi^{p,1,0}$ follows from multiplication:

$$\begin{aligned}\phi^{p,1,0}(x,y,z) &= \phi^{p,0,0}(x,y,z)P_1^{0,0}(\eta_2)\left(\frac{1-z}{2}\right)\\ &= \phi^{p,0,0}(x,y,z)\left(2\frac{1+y}{1-z}-1\right)\left(\frac{1-z}{2}\right)\\ &= \phi^{p,0,0}(x,y,z)\left(\frac{1+2y+z}{2}\right).\end{aligned} \tag{43}$$

Hence, all $\phi^{p,q,0}$ are constructing by a recurrence on q:

PROPOSITION 6.2. *Starting from*

$$\begin{aligned}\chi_2^0(y,z) &= 1,\\ \chi_2^1(y,z) &= \frac{1+2y+z}{2},\end{aligned} \tag{44}$$

there holds the three-term recurrence relation

$$\begin{aligned}\chi_2^p(y,z) = &\left(\frac{2p+1}{p+1}\right)\left(\frac{1+2y+z}{2}\right)\chi_1^p(y,z)\\ &-\left(\frac{p}{p+1}\right)\left(\frac{1-z}{2}\right)^2\chi_1^{p-1}(y,z).\end{aligned} \tag{45}$$

Therefore, for all $q \geq 1$,

$$\begin{aligned}\phi^{p,q+1,0}(x,y,z) = &\left(\frac{2p+1}{p+1}\right)\left(\frac{1+2y+z}{2}\right)\phi^{p,q,0}(x,y,z)\\ &-\left(\frac{p}{p+1}\right)\left(\frac{1-z}{2}\right)^2\phi^{p,q-1,0}(x,y,z)\end{aligned} \tag{46}$$

Given all of the $\phi^{p,q,0}(x,y,z)$, all of the $\phi^{p,q,1}(x,y,z)$ are readily constructed by the computation

$$\begin{aligned}\phi^{p,q,1}(x,y,z) &= \phi^{p,q,0}(x,y,z)P_1^{2p+2q+2,0}(z)\\ &= \phi^{p,q,0}(x,y,z)\left(1+p+q+(2+q+p)z\right).\end{aligned} \tag{47}$$

The recurrence in r is identical to that for tetrahedra.

PROPOSITION 6.3. *For all $p, q \geq 0$ and $r \geq 1$, the pyramidic polynomials $\phi^{p,q,r}(x,y,z)$ satisfy the three-term recurrence*

$$\begin{aligned}\phi^{p,q,r+1}(x,y,z) = \left(a_p^{2p+2q+2,0} z + b_q^{2p+2q+2,0}\right) \phi^{p,q,r}(x,y,z) \\ - c_p^{2p+2q+2,0} \phi^{p,q,r-1}(x,y,z).\end{aligned} \tag{48}$$

An algorithm for tabulating the pyramidic functions and differentiation rules follow in similar fashion.

7. SOME ALGORITHMIC CONSIDERATIONS

Flexibility of evaluation and differentiation has been the primary goal in developing this new representation of the orthogonal polynomials. However, polynomials expressed in these orthogonal bases may also be evaluated efficiently using the recurrence relations, even though the representation is not separable. The goal of this section is to point out that efficiency need not be lost with the new approach to the polynomials rather than to present a fully-functional spectral element simulator. The presentation will be restricted to the triangular case, although the techniques are also applicable to the three-dimensional shapes as well.

Let $u \in P_k(\hat{K})$ be given by

$$u(x_0, y_0) = \sum_{0 \leq p+q \leq \dim P_k} \mathrm{u}_{p,q} D^{p,q}(x_0, y_0). \tag{49}$$

Using the recurrence relations, u may be evaluated at any point in an amount of work proportional to $\dim P_k$, using auxilliary storage proportional to k. Algorithm 0**??** is structured to highlight the use of recurrence relations, but it is possible to work degree-by-degree while storing the $D^{p,q}$ values from previous degrees to enable polynomial evaluation.

Algorithm 3 Evaluating a polynomial of degree k expressed in the Dubiner basis

```
1: v := u_{0,0}
2: D^{1,0}(x_0, y_0) := (1+2x_0+y_0)/2
3: v+ = u_{1,0} D^{1,0}(x_0, y_0)
4: D^{0,1}(x_0, y_0) := (1+2x_0+3y_0)/2
5: v+ = u_{0,1} D^{0.1}
6: for d ← 0, k do
7:     for q ← 0, d do
8:         p := d − q
9:         Compute D^{p,q}(x_0, y_0) via recurrence
10:        v+ = u_{p,q} D^{p,q}(x_0, y_0).
11:    end for
12: end for
```

For each degree d in the outer loop of Algorithm 7, only values of the Dubiner basis for the two previos degrees are required. Rather than storing all of the values of the Dubiner basis during the computation, it is only required to store three degrees (the current plus two previous). Therefore, auxilliary storage proportional

to k is required, together with the $O(k^2)$ operations to perform the loop. This is optimal work complexity for polynomial evaluation.

Computing ∇u at (x_0, y_0) is no harder in terms of complexity with respect to k. The recurrence rules may be differentiated, and the same algorithm used. Alternatively, this may make use of the result of Griewank for automatic differentiaion that the derivatives may be obtained in using a constant multiple of the amount of work required for forward evaluation.

More importantly, the new representation does not affect the complexity of evaluating inner products compared to the separable representation. Suppose that a triangular quadrature rule on triangles is of the form

$$\int_{\hat{K}} f \mathrm{dx} = \sum_{i=1}^{Q_1} \sum_{j=1}^{Q_j} w_i^1 w_j^2 f(x_i^j, y_j). \tag{50}$$

This restricts the quadrature points to lie on a fixed set of Q_2 y coordinates. For each y coordinate y_j, there are Q_1 points. However, each y may have a different set of x points. This is exactly the situation that occurs when one maps tensor-product Gauss-type quadrature rules in the (η_1, η_2) coordinates using (8).

Using such a quadrature rule, consider the inner product over one triangle of some function u against a single basis function. This is approximated via the quadrature rule as

$$\begin{aligned}(u, D^{p,q}) &= \int_{\hat{K}} u D^{p,q} \mathrm{dx} \\ &\approx (u, D^{p,q})_\delta \\ &\equiv \sum_{i=1}^{Q_1} \sum_{j=1}^{Q_j} w_i^1 w_j^2 u(x_i^j, y_j) D^{p,q}(x_i^j, y_j) \\ &\equiv \end{aligned} \tag{51}$$

Typically, a polynomial basis of degree k is used so that $Q_1 = Q_2 = \mathcal{O}(k)$. This makes the inner product against a single $D^{p,q}$ an $O(k^2)$ process. With $O(k^2)$ total basis functions, this makes the whole inner product cost $O(k^4)$ to evaluate. This process can be reduced to $O(k^3)$ in the separable representation via sum-factorization [Karniadakis and Sherwin 2005].

However, the representation (10) also admits this improved complexity. To see this, insert (0**??**) into (51) and factor out $\psi^{p,q}(y_j)$ to obtain

$$(u, D^{p,q})_\delta = \sum_{j=1}^{Q_2} \psi^{p,q}(y_j) \sum_{i=1}^{Q_1} w_i w_j \chi^p(x_i^j, y_j) u(x_i^j, y_j). \tag{52}$$

As in the separated case, the innermost sum is over $O(k)$ points and is independent of q. Define the array

$$f_{p,j} \equiv \sum_{i=1}^{Q_1} w_i w_j \chi^p(x_i^j, y_j) u(x_i^j, y_j). \tag{53}$$

This array is $(k+1) \times Q_1$, so it contains $O(k^2)$ entries. This array may be constructed once and stored in $O(k^2)$ operations. Once this is done, evaluating the inner product

for a given $D^{p,q}$ is accomplished by

$$(u, D^{p,q})_\delta = \sum_{j=1}^{Q_2} \psi^{p,q}(y_j) f_{p,j}, \tag{54}$$

which is an $O(k)$ operation for each p, q pair, reducing the total complexity to $O(k^3)$. Similar complexity-reducing sum factorizations are available for the three-dimensional bases.

8. SOME EXPERIMENTS

Representing differentiation as matrices provides a way of treating all points in the reference domain uniformly. That is, for $u = \sum_{i=1}^{\dim P} \mathrm{u}_i \phi_i$, where there is some linear ordering of the Dubiner basis functions, then there exists a matrix D_x such that

$$\frac{\partial u}{\partial x} = \sum_{i=1}^{\dim P} (\mathrm{D}_x \mathrm{u})_i \phi_i. \tag{55}$$

As a result, any polynomial expressed in the Dubiner basis may be differentiated at any point simply by computing the matrix-vector product and using the standard technique for evalating the basis functions. Higher derivatives are evaluated by repeated matrix-vector multiplication. Constructing this matrix requires differentiating the basis functions at a set of $\dim P$ unisolvent points, but these may be selected away from the boundary. This approach has been used in the FIAT project [].

It turns out that the new recurrence relations are slightly more accurate than the collapsed-coordinate representation, but using automatic differentiation in the recurrence relations turns out to be significantly more accurate than the matrix multiplication formulation in (55). As elements such as Argyris are applicable to fourth order problems, it is important to evaluate at least second derivatives, and even higher derivatives may be needed for stabilization or residual-based error estimates.

Table 0**??** shows the accuracy obtained in tabulating and differentiating the orthogonal expansions on triangles and tetrahedra for various polynomial orders on a lattice consisting of $\dim P_{11} = \left(\Pi_{i=1}^{d}(d+i)\right)/d!$ equispaced points. The exact answer was obtained by evaluating the new recurrence relations in rational arithmetic and using automatic differentiation to evaluate the derivatives. Then, the new recurrence relations were computed in double precision floating point arithmetic, with derivatives computed also by AD in double precision as well. The standard separated formulation was then used to tabulate the polynomials in double precision, and (0**??**) was used to compute each of the partial derivatives by matrix multiplication. Table 0**??** reports the errors obtained for the triangular and tetrahedral polynomials, with columns indexed by polynomial degree and rows indexed by order of differentiation. Each entry represents the maximum error over all points in the lattice for the entire set partial derivatives of a given order for the polynomials of a given degree.

Table I. Error in tabulating triangular polynomials using the separated representation and the new recurrences. The polynomial degree k increases going down the rows, and the columns are indexed by the order of differentation. Although the separated expansions are usually slightly more accurate than the new recurrences, the error incurred by differentiating with matrix multiplication grows much more quickly than AD in the new recurrences.

		0	1	2	3
3	old	5.4E-16	1.2E-14	6.0E-14	2.3E-13
	new	4.8E-16	3.8E-15	7.8E-15	7.1E-15
4	old	5.4E-16	4E-14	2.7E-13	1.7E-12
	new	8.3E-16	8.0E-15	2.5E-14	3.9E-14
5	old	5.7E-16	9.9E-14	1.2E-12	5.3E-12
	new	9.7E-16	1.2E-14	8.6E-14	3.4E-13
6	old	5.7E-16	4.1E-13	4.7E-12	4.3E-11
	new	1.5E-15	1.6E-14	1.9E-13	1.4E-12
7	old	8.9E-16	9.3E-13	1.2E-11	1.4E-10
	new	1.9E-15	2.8E-14	3.0E-13	2.6E-12
8	old	3.6E-15	5.2E-12	9.1E-11	7E-10
	new	3.6E-15	5.7E-14	4.9E-13	6.7E-12

Table II. Error in tabulating tetrahedral polynomials using the separated representation and the new recurrences. The polynomial degree k increases going down the rows, and the columns are indexed by the order of differentation. Although the separated expansions are usually slightly more accurate than the new recurrences, the error incurred by differentiating with matrix multiplication grows much more quickly than AD in the new recurrences.

		0	1	2	3
3	old	1.2E-15	4.6E-14	1.1E-13	3.1E-13
	new	1.7E-15	6.5E-15	7.8E-15	7.1E-15
4	old	1.5E-15	8.0E-14	7.4E-13	3.8E-12
	new	2.9E-15	1E-14	3.6E-14	7.2E-14
5	old	2.7E-15	6.8E-13	3.5E-12	3.5E-11
	new	4E-15	2.0E-14	8.7E-14	3.4E-13
6	old	2.8E-15	1.9E-12	3.0E-11	2.6E-10
	new	4E-15	3.8E-14	2.3E-13	1.4E-12
7	old	2.8E-15	6.1E-12	1.1E-10	1.5E-09
	new	5.0E-15	1.1E-13	4.9E-13	3.6E-12
8	old	3.6E-15	2.0E-11	3.8E-10	3.2E-09
	new	6.1E-15	1.1E-13	1.2E-12	8.1E-12

9. CONCLUSIONS

REFERENCES

Abramowitz, M. and Stegun, I. A. 1964. *Handbook of mathematical functions with formulas, graphs, and mathematical tables.* National Bureau of Standards Applied Mathematics Series, vol. 55. For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C.

Beuchler, S. and Pillwein, V. 2007. Sparse shape functions for tetrahedral p-FEM using integrated Jacobi polynomials. *Computing 80,* 4, 345–375.

Beuchler, S. and Schöberl, J. 2006. New shape functions for triangular p-FEM using integrated Jacobi polynomials. *Numer. Math. 103,* 3, 339–366.

Braess, D.

Braess, D. 2005. Approximation on simplices and orthogonal polynomials. In *Trends and Applications in Constructive Approximation*, M. de Bruin, D. Mache, and J. Szabados, Eds. Birkhuser Verlag, Basel, 53–60.

Castillo, P. 2002. Performance of discontinuous Galerkin methods for elliptic PDEs. *SIAM J. Sci. Comput. 24,* 2, 524–547 (electronic).

Castillo, P., Rieben, R., and White, D. 2005. FEMSTER: An object-oriented class library of high-order discrete differential forms. *ACM Trans. Math. Software 31,* 4, 425–457.

Dubiner, M. 1991. Spectral methods on triangles and other domains. *J. Sci. Comput. 6,* 4, 345–390.

Hesthaven, J. S. and Warburton, T. 2002. Nodal high-order methods on unstructured grids. I. Time-domain solution of Maxwell's equations. *J. Comput. Phys. 181,* 1, 186–221.

Karniadakis, G. E. and Sherwin, S. J. 2005. *Spectral/hp element methods for computational fluid dynamics*, Second ed. Numerical Mathematics and Scientific Computation. Oxford University Press, New York.

Kirby, R. C. 2004. Algorithm 839: FIAT, a new paradigm for computing finite element basis functions. *ACM Trans. Math. Software 30,* 4, 502–516.

Kirby, R. C. 2006. Optimizing FIAT with level 3 BLAS. *ACM Trans. Math. Software 32,* 2, 223–235.

Koornwinder, T. 1975. Two-variable analogues of the classical orthogonal polynomials. In *Theory and application of special functions (Proc. Advanced Sem., Math. Res. Center, Univ. Wisconsin, Madison, Wis., 1975)*. Academic Press, New York, 435–495. Math. Res. Center, Univ. Wisconsin, Publ. No. 35.

...