Interpolation error bounds for curvilinear finite elements and their implications on adaptive mesh refinement

David Moxey · Shankar P. Sastry · Robert M. Kirby

Abstract Mesh generation and adaptive refinement are largely driven by the objective of minimizing the bounds on the interpolation error of the solution of the partial differential equation (PDE) being solved. Thus, the characterization and analysis of interpolation error bounds for curved, high-order finite elements is often desired to efficiently obtain the solution of PDEs when using the finite element method (FEM). Although the order of convergence of the projection error in L^2 is known for both straight-sided and curved-elements [1], an L^∞ estimate as used when studying interpolation errors is not available. Using a Taylor series expansion approach, we derive an interpolation error bound for both straight-sided and curved, high-order elements. The availability of this bound facilitates better node placement for minimizing interpolation error compared to the traditional approach of minimizing the Lebesgue constant as a proxy for interpolation error. This is useful for adaptation of the mesh in regions where increased resolution is needed and where the geometric curvature of the elements is high, e.g, boundary layer meshes. Our numerical experiments indicate that the error bounds derived using our technique are asymptotically similar to the actual error, i.e., if our interpolation error bound for an element is larger than it is for other elements, the actual error is also larger than it is for other elements. This type of bound not only provides an indicator for which curved elements to refine but also suggests whether one should use traditional h-refinement or should modify the mapping function used to define elemental curvature. We have validated our bounds through a series of numerical experiments on both straight-sided and curved elements, and we report a summary of these results.

David Moxey
College of Engineering, Mathematics and Physical Sciences
University of Exeter
E-mail: d.moxey@exeter.ac.uk

Shankar P. Sastry · Robert M. Kirby
Scientific Computing and Imaging Institute
University of Utah
E-mail: {sastry, kirby}@sci.utah.edu

1 Introduction

Many mesh generation and *a posteriori* adaptation strategies for the finite element method (FEM) are motivated by the reduction of interpolation errors over the mesh elements to be generated or refined. For linear finite elements, there is a long tradition of both *a priori* and *a posteriori* error estimates that aid the FEM practitioner. When one moves to (straight-sided) high-order FEM, there are fewer estimates available. Finally, when one considers curvilinear high-order elements with curved sides and faces, which are crucial in correctly aligning the mesh with an underlying geometry, results are even more sparse.

Curvilinear elements are obtained through the use of a geometric mapping from a reference element to the desired element in physical Cartesian space (i.e., world-space). The mapping is typically defined in an isoparametric or subparametric manner using a subset of the same polynomial basis functions that are used to represent the approximated solutions. However, in using this approach, an issue that has always bothered FEM practitioners is the question of how the choice of the geometric mapping functions impacts on the approximation properties of those elements in world-space, i.e., in the natural coordinate system of interest. Botti [1] was among the first to quantify the impact of polynomial mapping functions on the quality of the solution – providing a proof on the order of convergence in an integral norm of an FEM projection operator for quadrilateral elements as a function of elemental polynomial (enrichment) order p and mapping order q.

Many of the contemporary results for bounding the error in finite element interpolation, including [1], have been derived from the Bramble-Hilbert lemma [2]. In their paper, Bramble and Hilbert explicitly mention that this lemma can replace the "standard" Taylor series technique. This approach, however, does not naturally translate to an understanding of a "good" high-order finite element from the perspective of interpolation error, owing to the lack of a constructive proof. This aspect was addressed by Dupont and Scott [3], who were able to provide a constructive proof of the lemma using a Taylor series expansion. Arnold *et al.* [4] proved error bounds for straight-sided quadrilateral finite elements by extending the Bramble-Hilbert lemma. Dekel and Lavitan [5], however, argued that the main drawback of their proof is that the constant associated with the order of convergence depends on a "chunkiness parameter", which blows up as a triangle becomes very thin. In order to accommodate this drawback, a quasi-uniform mesh is assumed in which the parameter is uniformly bounded. Dekel and Leviatan [5] provided a constructive proof for the lemma that showed that the interpolated polynomial in a star-shaped domain obeys the bounds for a straight-sided finite element.

A key drawback of approaches based on the Bramble-Hilbert lemma, in our view, is that these results lead to bounds that are defined in an integral manner and therefore do not offer much insight into how the geometric mapping can be changed in order to reduce the interpolation error. We note that from a practical perspective, the mapping relies on a distribution of points in the reference and Cartesian elements, alongside a set of Lagrange interpolants. Although there is a rich body of literature relating to the effects of the distribution of points in the reference element, the placement of the points in the Cartesian element and how this pertains to the interpolation error are far less well-understood.

In earlier work [6], we highlighted how a first-principles approach, using a Taylor series expansion of the function to be interpolated, leads to a pointwise bound on the interpolation error for second-order straight-sided elements. This bound, at a point in the high-order element, is proportional to the sum of the product of its distance from a high-order node of the triangle with the value of the corresponding shape function. This node-based approach gives FEM and adaptive mesh refinement (AMR) practitioners a method through which nodes can be moved and placed optimally in order to reduce the interpolation error of a given function.

In this paper, we show how this approach can be extended to derive interpolation error bounds that are applicable for any valid (i.e., non self-intersecting) finite element mesh at a polynomial order p and mapping order q. In particular, we highlight that the bound is again dependent on the position of the mesh nodes, and thus our work provides a tighter bound than Lebesgue function-based bounds, but with an assumption on the bounds on high-order derivatives of the function we interpolate. This paper can be viewed as an extension of [5] in the context of the high-order, curved finite elements and as an extension of [1] with constructive proofs for an interpolation version of the theorems stated in their paper. We also consider triangular elements due to their nearly ubiquitous nature in the FEM, as opposed to the quadrilaterals investigated in [1].

As we prove our results from first principles using the Taylor series, it becomes possible to define the quality of a high-order, curved element and to optimally place nodes within a triangle to minimize the error bounds. This approach also enables us to analyze the effect of anisotropy on the error bounds. The proofs presented in this paper are considerably simpler than Bramble-Hilbert lemma-based proofs in earlier work in that they do not require extensive knowledge of functional analysis. We believe that the biggest impact of this paper will be in AMR for engineering applications in which high-order meshes are used.

We conclude with a brief overview of the structure of this paper. In section 2, we reproduce the result of [1] in terms of the order of convergence of solutions under h-refinement, but under the L^∞ norm instead of the L^2 norm. We then outline the bounds in terms for both the interpolation of a function and its derivatives. In Section 3, we present a number of numerical experiments to validate the order of convergence for triangular elements, and examine the tightness of the bounds in the context of anisotropic mesh refinement. Since the research and application of high-order FEM are still relatively nascent, the scope for future research in this field is large. We will indicate future research directions in Section 4, including the study of conditioning of stiffness matrices, node placement, and error estimation techniques for high-order polynomial interpolation.

2 Interpolation Error Bound

In this section, we derive error bounds for the piecewise interpolation error of p-th order triangular elements with a q-th order reference-to-physical space mapping for a function and its derivative. The proofs for the error bounds closely follow the techniques employed by Johnson [7] and our previous paper [6].

2.1 Overview

We begin by providing a brief mathematical framework of the interpolation problem. Let K^e denote a triangular element belonging to the conformal mesh of a domain $\Omega \subset \mathbb{R}^2$, so that $\Omega = \bigcup_e K^e$ and $K^e \cap K^f = \emptyset$ if $e \neq f$. This mesh is then combined with an appropriate high-order polynomial expansion on each element, so that discrete solutions of a given PDE will belong to the space

$$\mathcal{D}^p(\Omega) = \{v \in L^2(\Omega) \mid v|_{K^e} \in \mathbb{P}^p(K^e)\},$$

where $L^2(\Omega)$ denotes the space of square integrable functions on Ω and

$$\mathbb{P}^p(K^e) = \mathrm{span}\{x_1^i x_2^j \mid (x_1, x_2) \in K^e,\ 0 \leq i \leq p,\ 0 \leq i + j \leq p\}$$

is the polynomial space of total degree p on K^e, which is spanned by $N(p) = \frac{1}{2}(p+1)(p+2)$ basis functions.

This paper focuses on a key part of both the numerical solution of a PDE and applications in AMR: the polynomial interpolation of a known function $u : \Omega \to \mathbb{R}$ to obtain a discrete approximation $u_h \in \mathcal{D}^p(\Omega)$. This process can be viewed as the application of a projection π so that $u_h = \pi u$, and can be defined in a number of ways depending on the desired outcome: for example in [1], π is an L^2 projection. Here however, in order to provide constructive proofs, we consider π to be the result of a L^∞ interpolation operator, which we will define in the following section with the aide of a reference element. The intent of this work is to obtain convergence properties and bounds on the pointwise interpolation error $\|u - u_h\|_{L^\infty}$ by assuming sufficient regularity of u so as to apply Taylor's theorem. To this end, we therefore consider functions $u \in C_b^k(\Omega)$ that are bounded and k-times continuously differentiable for sufficiently large k, so that $\pi : C_b^k(\Omega) \to \mathcal{D}^p(\Omega)$.

2.2 Reference-to-world mapping

In order to facilitate the definition of a basis spanning $\mathcal{D}^p(K^e)$ on each element, we consider a reference triangle

$$K = \{(\xi_1, \xi_2) \mid \xi_i \in [-1, 1], \xi_1 + \xi_2 \leq 0\}$$

and define a basis on K. To assist in the definition of π, we consider the commonly-used basis of total degree p Lagrange interpolants $\lambda_k(\xi)$, so that a function $f : \mathcal{D}^p(K) \to \mathbb{R}$ can be written as

$$f(\xi) = \sum_{k=1}^{N(p)} \hat{f}_k \lambda_k(\xi),$$

where $\{\hat{\xi}_i\}_{i=1}^{N(p)} \subset K$ denotes a distribution of solvent points within the reference element and $\lambda_i(\hat{\xi}_j) = \delta_{ij}$, where δ_{ij} is the Kronecker delta. In this setting, the interpolation operator π can then be viewed as the mapping such that, for an arbitrary function $v : \Omega \to \mathbb{R}$, πv is the unique polynomial such that $\pi v(\hat{\xi}_k) = v(\hat{\xi}_k)$ for each k and

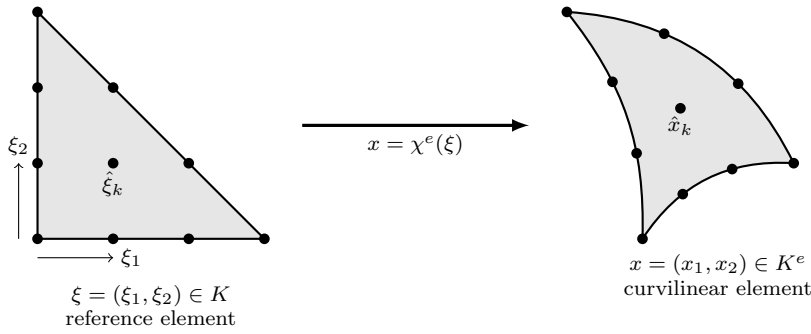$$\pi v(\xi) = \sum_{k=1}^{N(p)} v(\hat{\xi}_k) \lambda_k(\xi). \tag{1}$$

Fig. 1: Notation showing reference-to-world mappings χ^e from the reference element K to a physical element K^e, where the nodes correspond to an order of $p = 3$. The distributions of nodal coordinates are denoted by $\hat{\xi}_k \in K$ and $\hat{x}_k \in K^e$.

To translate this basis on K to a basis on any given element K^e, we require the definition of a diffeomorphism $\chi^e : K \to K^e$ between the reference and physical space element to define a basis over the world-space element that involves polynomials in the reference space. An important aspect of this high-order finite element setting is that each element K^e is generally assumed to be curvilinear. This allows, for example, the mesh elements that touch the domain boundary to be curved so that the mesh accurately represents the underlying geometry. This is particularly important in the context of, for example, fluid dynamics problems. Here, an accurate representation of the geometry is critical in the accurate prediction of important flow features such as separation and recirculation, which define key aspects of the fluid's behaviour.

To define the reference-to-physical space mapping and to enable the use of curved elements, we typically choose to employ the use of mappings that draw from a subset of the basis functions defined on K. That is, we suppose that $\chi^e = (\chi_1^e, \chi_2^e) \in [\mathbb{P}^q(K)]^2$ for some polynomial order $q \leq p$, where if $q = p$ the mapping is referred to as isoparametric, if $q < p$ it is subparametric and if $q > p$ it is superparametric. Each coordinate of a given point $x = (x_1, x_2) \in K^e$ can be expressed as a function of the mapping from a point $\xi \in K$, which in turn is written as a polynomial

$$x_k = \chi_k^e(\xi) = \sum_{i,j=0}^{i+j \leq q} \alpha_{ij,k} \xi_1^i \xi_2^j, \tag{2}$$

where $\alpha_{ij,k}$ are constants depending on the element K^e. Hereafter, $x \in \Omega$ will denote coordinates in the physical Cartesian space and $\xi \in K$ in the reference triangle.

Since we consider Lagrange interpolants as our particular basis on K, another way to define this mapping explicitly is through the definition of nodal points $\{\hat{x}_k = ((\hat{x}_k)_1, (\hat{x}_k)_2)\}_{k=1}^{N(q)} \subset K^e$ on each element. This then yields

$$x_k = \chi_k^e(\xi) = \sum_{i=1}^{N(q)} (\hat{x}_i)_k \lambda_i(\xi)$$

so that under this mapping, the reference points $\hat{\xi}_k$ map onto the physical elemental nodes $\hat{x}_k$. Figure 1 shows a visual representation of this setting. Throughout this paper we make the assumption followed in [8], that χ^e has a positive determinant for every $\xi \in K$ for every element K^e in the mesh, and its inverse is smooth.

Finally, we note in particular that the distributions of nodes $\hat{\xi}_k$ in relation to their interpolation properties on K is a very well-studied topic, as described in e.g. [8]. Generally, the goal of such points is to minimise the Lebesgue constant of π, leading to distributions such as those proposed by Hesthaven [9]. However, the effect of the distribution of $\hat{x}_k$ on the interpolation operator is *far less well-understood* in the context of curvilinear elements. This is the key aim of this paper and the results of the following two sections.

2.3 Convergence order

Our first goal is to consider the convergence order that is obtained in the interpolation process under h-refinement (reduction of element size). That is, given a local element size h we wish to determine the order k such that the L^∞ error $\|u - \pi u\|_{L^\infty} = \mathcal{O}(h^k)$. In [1], Botti showed that the order of convergence for a reference-to-physical space mapping of order q is $k = \lfloor p/q \rfloor + 1$ in the L^2 norm through the use of the Bramble-Hilbert lemma. In this section, we highlight how the same result under the L^∞ norm can be recovered in a first-principles setting, under the assumption of sufficient regularity of u.

Theorem 1 *Given an element size h, The order of convergence of the interpolation error $\|u - \pi u\|_{L^\infty}$ is $\lfloor p/q \rfloor + 1$ under h-refinement of the mesh.*

Proof We begin with a Taylor series expansion of the true solution u in world-space around an arbitrary world-space point $x^* \in K^e$. In a standard Taylor expansion, this is represented in terms of a power series

$$\sum_{0 \leq |\alpha| \leq p} \frac{(x - x^*)^\alpha}{\alpha!} \partial^\alpha u(x^*) + D_{p+1}(\zeta),$$

where α is a 2-tuple multi-index and D is a remainder term of polynomial order $p+1$ depending on a point ζ lying on the line segment between x and x^*. However, for the coming analysis, since we have explicit expressions for x_1 and x_2 in terms of ξ_1 and ξ_2, consider a rearrangement of this series in terms of monomials, so that

$$u(x) = \sum_{i,j=0}^{i+j \leq p} a_{ij} x_1^i x_2^j + D_{p+1}(\zeta). \tag{3}$$

In this form, a_{ij} are summations of constants which depend on x^* and the partial derivatives of u up to order $i + j$ evaluated at x^*.

Before proceeding further, to help clarify the coming argument, we first outline the relationship between the true solution u and its representation in world-space K^e and reference-space K. Recall that these spaces are related through the polynomial mapping $\chi^e : K \to K^e$. The true solution u can therefore either be viewed as a function in the world-space element K^e, or alternatively in the reference element K through the mapping $v := u \circ \chi^e$. However, we note that since $(\chi^e)^{-1}$

is not necessarily polynomial, the interpolation operator πu (with respect to x) is a smooth, but not necessary polynomial, function. Hence when we interpolate in world space we are not forming an interpolating polynomial (in world space) but an interpolating function. We therefore opt to consider u in its mapped coordinate v, under which the interpolation πv is a polynomial function, and leverage the fact that $\|u - \pi u\|_{L^\infty} = \|v - \pi v\|_{L^\infty}$.

To achieve this, we apply the mapping (2) to Eq. (3) to obtain the expression

$$v(\xi) = \sum_{i,j=0}^{i+j \leq p} \left(a_{ij} \left[\sum_{k,l=0}^{k+l \leq q} \alpha_{kl,1} \xi_1^k \xi_2^l \right]^i \left[\sum_{k,l=0}^{k+l \leq q} \alpha_{kl,2} \xi_1^k \xi_2^l \right]^j \right) + D_{p+1}(\zeta), \quad (4)$$

which allows us to "view" the function $u(x)$ as defined in Eq. (3) on the reference element, which we denote by $v(\xi)$.

Now let πv denote the polynomial interpolant of v, which we note is a polynomial of total degree p in ξ. In this case, πv is a summation of monomials $\xi_1^i \xi_2^j$ such that $i + j \leq p$. We note from Equation (4) that under the application of χ^e, the interpolation error $v(\xi) - \pi v(\xi)$ now contains monomials up to order pq, which is greater than the remainder term of order $p+1$ for all $q > 1$. We therefore determine the order of convergence as the minimum value of $i+j$ such that monomials inside the inner summations are of order less than or equal to $p + 1$, since these are the only values that can be supported under the expansion of total degree p.

Finally then, we consider the maximum value of each inner summation where $k+l = q$. This choice of k and l for a given p and q have corresponding leading-order monomials $\xi_1^{ik} \xi_2^{il} \xi_1^{jk} \xi_2^{jl}$ of total degree $q(i + j)$. The convergence order is therefore given by the minimum integer $n = i+j$ such that $qn \geq p+1$, which is clearly given by $n = \lfloor p/q \rfloor + 1$. $\square$

2.4 Bounds on interpolation error

We now consider the effects of the projection operator and its impact on the interpolation error. We start by examining a slightly different Taylor expansion than the one utilised for $u(x)$ in Equation (3). Namely, we consider the expansion around a point $\xi \in K$ evaluated at one of the vertices of the triangle $\hat{\xi}_k$, so that

$$v(\hat{\xi}_k) = v(\xi) + \left(\sum_{i,j=0}^{0<(i+j) \leq pq} b_{ij,k} \xi_1^i \xi_2^j \right) + D_{p+1}(\tilde{\zeta}).$$

Note that if we adopted the form of Equation (3), the summation above would be written in terms of $(\hat{\xi}_k)_1^i (\hat{\xi}_k)_2^j$. In the above expression, we instead consider the summation in terms of $\xi_1^i \xi_2^j$. The coefficients $b_{ij,k}$ depend on a summation of partial derivatives of v evaluated at $\hat{\xi}_k$ up to order $i + j$, as denoted by the subscript. $\tilde{\zeta}$ is a point on the line segment connecting $\hat{\xi}_k$ and ξ.

By substituting the above into the polynomial expansion of Equation (1) we obtain

$$\pi v(\xi) = \sum_{k=1}^{N(p)} \lambda_k(\xi) \left[v(\xi) + \left(\sum_{i,j=0}^{0<(i+j) \leq pq} b_{ij,k} \xi_1^i \xi_2^j \right) + D_{p+1}(\tilde{\zeta}) \right]. \quad (5)$$

As has been shown for linear [7] and quadratic [6] interpolation over a triangle, we will show that the following lemma holds at higher orders, thereby reducing terms in the equation above to zero.

Lemma 1

$$\sum_{k=1}^{N(p)} \lambda_k(\xi) = 1 \tag{6}$$

$$\sum_{k=1}^{N(p)} \lambda_k(\xi) \left(\sum_{i,j=0}^{0 < (i+j) \leq p} b_{ij,k} \xi_1^i \xi_2^j \right) = 0. \tag{7}$$

Note that the summation terminates when $(i+j) = p$, not when $(i+j) = pq$.

Proof Just as it has been obtained by Johnson [7] and in our previous paper [6], the proof of the lemma is obtained by constructing appropriate polynomial functions w for every point in the domain K. To show Equation (6), let $w(\xi) = c$ be a constant function. Then clearly $w = \pi w$, giving $c = \pi w(\xi) = \sum_i \lambda_i(\xi) w(\xi) = c \sum_i \lambda_i(\xi)$ and $\sum_i \lambda_i(\xi) = 1$. To show Equation (7) for known values of partial derivatives (up to p-th order) of a function at a given point, a p-th order polynomial function w can be constructed such that the polynomial's value and partial derivatives are identical to the original function. Moreover, this is exactly evaluated by the interpolation technique. Since we can construct these functions at any point, for every point in the domain, Equation (7) holds. □

By substituting the lemma above into Equation (5) and rearranging the terms, we obtain

$$\pi v(\xi) - v(\xi) = \sum_{k=1}^{N(p)} \lambda_k(\xi) \left(\sum_{(i,j)}^{(i+j) > p} b_{ij,k} \xi_1^i \xi_2^j + D_{p+1}(\tilde{\zeta}) \right).$$

This again highlights that the error is bounded by coefficients of $\xi_i \xi_j$, where $i + j > p$. Furthermore, since we assume that v has sufficient regularity, we may take an explicit form of D_{p+1} so as to obtain a bound on the interpolation error as in [6], which leads to the following theorem:

Theorem 2 *The interpolation error $|u(x) - \pi u(x)|$ over K^e obeys the inequality*

$$|u(x) - \pi u(x)| \leq \frac{c}{(p+1)!} \sum_{i=1}^{N(p)} |\lambda_i^e(x)| \|x - \hat{x}_i\|_2^{p+1} \tag{8}$$

where c is the upper bound on the $(p+1)$-th derivative of u, $\| \cdot \|_2$ is the Euclidean norm on $\mathbb{R}^2$ and $\lambda_i^e = \lambda_i \circ (\chi^e)^{-1}$.

Note that in the above statement, there is no explicit appearance of the mapping order q. Upon first glance therefore, it might be inferred that the interpolation error scales at order $p+1$ based on the term $\|x - \hat{x}_i\|_2^{p+1}$, which lies in contradiction of Theorem 1. However, we note that q does appear implictly and nontrivially in the definition of the shape functions λ_i^e, since these are defined using the reference-to-world mapping χ^e. We will further investigate the validity and tightness of this bound numerically in Section 3.2.

2.5 Bounds on the derivatives of interpolated function

Finally we consider a bound for the error in the derivative of the approximate function by adopting the same approach taken in previous work [7,6]. By differentiating Equation (1) and using the Taylor series expansion at x, we obtain the gradient of the polynomial model function as shown here:

$$\frac{\partial \pi v(\xi)}{\partial \xi_j} = \sum_{k=1}^{N(p)} \frac{\partial \lambda_i(\xi)}{\partial \xi_j} \left(v(\xi) + \left(\sum_{i,j=0}^{(i+j)\leq pq} b_{ij,k} \xi_1^i \xi_2^j \right) + D_{p+1}(\zeta) \right). \tag{9}$$

As described above, it can be shown that the following lemma holds:

Lemma 2

$$\sum_{k=1}^{N(p)} \frac{\partial}{\partial \xi_j} \lambda_k(x) = \frac{\partial}{\partial \xi_j} \sum \lambda_k(x) = 0, \tag{10}$$

$$\sum_{k=1}^{N(p)} \frac{\partial}{\partial \xi_j} \lambda_k(x) \left(\sum_{i,j=1}^{(i+j)\leq p} b_{ij_k} \xi_1^i \xi_2^j \right) = \frac{\partial v}{\partial \xi_j}, \tag{11}$$

$$\tag{12}$$

for $j = \{1, 2\}$.

By substituting these equations into Equation (9), we obtain

$$\frac{\partial \pi v(x)}{\partial \xi_j} - \frac{\partial v(x)}{\partial \xi_j} = \sum_{i=1}^{N(p)} \frac{\partial \lambda_i(x)}{\partial \xi_j} D_{p+1}(\zeta).$$

Theorem 3 *The h-refinement of a mesh results in the $\lfloor \frac{p}{q} \rfloor$-order convergence of the solution.*

Proof As we saw in the proof for the previous theorem, the coefficient of $\xi_i \xi_j$ when $(i+j) = (p+1)$ is a function of $\left(\lfloor \frac{p}{q} \rfloor + 1 \right)^{\text{th}}$ partial derivative of $u(x)$, but the $\frac{\partial \lambda_i(x)}{\partial \xi_j}$ is inversely proportional to the size of the element. Thus, the order of convergence for the derivative reduces by one. $\square$

3 Numerical Experiments

In this section, we perform a number of numerical experiments to validate the theory of the previous section. In Section 3.1, we consider a series of convergence tests on uniformly-refined grids of geometric orders up to $q = 6$ and validate the order of convergence as $\lfloor p/q \rfloor + 1$, as proven in Theorem 1. In Section 3.2, we examine the error bound on the interpolation error and investigate the tightness of the bound derived in Theorem 2 in the context of anisotropic grids.
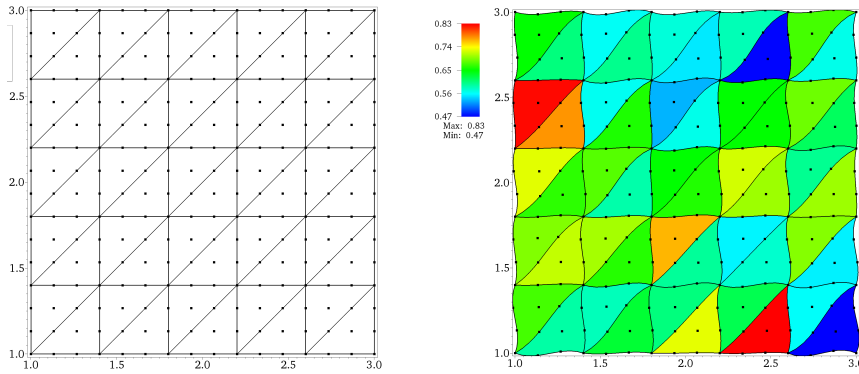
Fig. 2: High-order structured mesh using $N = 5$ subdivisions at geometry order $q = 3$. The original, straight-sided mesh is shown on the left. On the right we see the resulting curvilinear mesh once the perturbation process is applied, which is used for convergence testing. Colouring of the curved mesh denotes the magnitude of the scaled Jacobian of $\nabla\chi^e$ for each element.

3.1 Validation of convergence order

In this section, we provide a validation of the convergence order result proven in Theorem 1 and in [1]. In these tests, we consider the interpolation of the smooth function $u(x) = \sin(\pi x)\sin(\pi y)$ onto a polynomial space of degree six. For geometry orders $1 \leq q \leq 6$, we then construct a series of h-refined triangular meshes and examine the order of convergence for each refinement respectively, comparing it to the theoretical result. Our methodology is therefore a significant expansion from the results presented in [1], given that the domain adopts triangular elements as opposed to quadrilaterals, and we observe the theoretical results across a broader range of geometry orders than the quadratic orders considered in [1].

The starting point for this series of tests is a square domain Ω with corners at $(x, y) = (1, 1)$ and $(3, 3)$ respectively. We generate a series of structured, straight-sided triangular meshes of equal size through the subdivision of each side of Ω into N equally-sized segments, so that the element size $h = 2/N$, as shown in the left hand image of Figure 2. We consider subdivisions in the range $10^0 \leq N \leq 10^2$ so that the coarsest mesh consists of two triangles and the finest of 2×10^4 triangles. To generate meshes at each geometry order q, we first generate a straight-sided high-order mesh so that each edge of the triangle consists of $q + 1$ points lying on a line segment. The $q - 1$ edge-interior nodes of each triangle are then shifted by a small, random amount in the edge-normal direction. This is done in an alternating fashion, so that neighbouring edge nodes use both the interior-facing and exterior-facing normal directions. This process then guarantees the mesh is of the desired order q. The interior nodes locations are then calculated through a standard Gordon-Hall blending, and the perturbation is reduced appropriately at higher orders so as to ensure that every mesh is valid. An example of these high-order meshes can be seen in the right-hand image of Figure 2, which shows an $N = 5$ subdivided mesh at geometry order $q = 3$. This figure also shows the
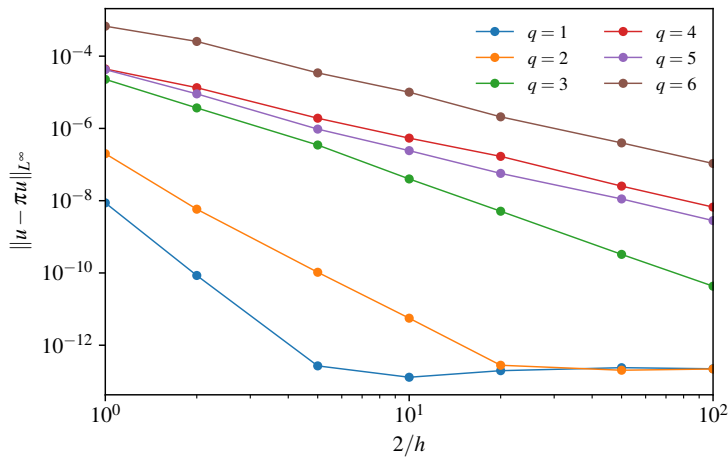
Fig. 3: Interpolation error $\|u-\pi u\|_{L^\infty}$ for the function $u(x_1, x_2) = \sin(\pi x_1)\sin(\pi x_2)$ and its sixth-order polynomial interpolant πu.

validity and randomised nature of the mesh by visualising the scaled Jacobian

$$J_s^e = \frac{\min_{\xi \in K}\{\det \nabla \chi^e(\xi)\}}{\max_{\xi \in K}\{\det \nabla \chi^e(\xi)\}}$$

associated with each element K^e, with reference-to-physical space map χ^e. In particular J_s^e is positive across the mesh, so that every element is valid.

We use the Nektar++ spectral/hp element framework [10] to perform the polynomial interpolation πu at an order of $p = 6$ of the function u. This is performed on every mesh generated at different subdivisions N and geometry orders q. Figure 3 shows the interpolation error recorded in each case, where the error is sampled using $N(20) = 231$ points within each element in order to obtain an accurate representation of the error. We note that this error saturates at a small value of around 10^{-13}, as seen in the results of $q = 1$ and $q = 2$, past which further h-refinement does not lead to reduction in the error.

The results show a clear trend in a reduction of convergence order as q is increased in a 'fan' of error curves, in line with the theoretical results of the previous section. In particular, convergence rates obtained through a simple linear regression are calculated as 1.86, 1.91, 1.89, 3.01, 4.31 and 6.45 for $q = 6$ to $q = 1$ respectively, which align well with the theoretical order of $\lfloor p/q \rfloor + 1$.

From these results we can therefore conclude that the geometry order has a significant impact of the order of convergence for even moderate polynomial orders. For example, using even the smallest amount of curvature (i.e. $q = 2$) leads to a reduction of convergence order of nearly a factor of two.

3.2 Examination of error bounds

In this section, we present results that show the correlation between the derived error bound of Eq. (8) and the actual interpolation error for various smooth func-
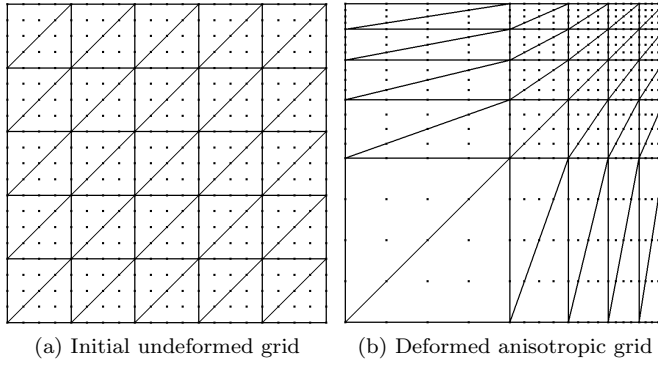
(a) Initial undeformed grid   (b) Deformed anisotropic grid

Fig. 4: Illustration of grid generation process under the deformation map g with $f(x) = x^4$. Undeformed mesh on the left hand side and deformed grid on the right hand side. High-order nodes are constructed after deformation and are shown at order $p = 5$.

tions. To highlight the usefulness of these bounds for adaptive mesh refinement, in which grids tend to be refined in an anisotropic fashion, we adjust our methodology to observe the interpolation error and bound on a sequence of grids that are constructed at various levels of anisotropy.

3.2.1 Generation of straight-sided anisotropic grids

In order to generate anisotropic meshes that form the starting point of these experiments, we consider a regular straight-sided triangulation of the same square domain Ω with corners at $(1, 1)$ and $(3, 3)$ as in the previous section. We initially consider a grid with $N = 5$ equally-sized elements along each edge of the square, as shown in Figure 4a. The grid is then deformed into one that is anisotropic (as in Figure 4b) by transforming each coordinate of the triangular vertices according to a one-dimensional function $g_f : [1, 3] \to [1, 3]$. This is defined analytically as

$$ g_f(x) = f^{-1} \left[\tfrac{1}{2}(f(3) - f(1))(x - 1) + f(1) \right], $$

where $f : [1, 3] \to \mathbb{R}$ is a smooth, invertible, strictly increasing function that controls the spacing of the elements. Under this definition, the endpoints of the domain are preserved since $g_f(1) = 1$ and $g_f(3) = 3$, but the spacing of element edges grows according to f. An example of this process for $f(x) = x^4$ is shown in Figure 4. Additionally, it is important to note that g_f is only applied to the three linear vertices of each triangle and *not* to the high-order nodes – these are constructed after the mesh is deformed.

3.2.2 Methodology

In the experiments that follow, we aim to measure the interpolation error of various smooth functions $u(x)$ at polynomial orders $p \leq 5$, along with the error bounds

of Eq. (8) derived in the previous section. The error between u and its resulting interpolant πu is calculated using a 10-th order set of high-order nodes within each element in order to obtain an accurate value of the error. The upper bound on the $(p+1)$-th derivative of u, represented as c in Eq. (8), is calculated at the same set of points. Finally, in order to assess the asymptotic properties of the tightness of the bound according to the anisotropic nature of the meshes, we report the L^∞ error and bound in only the smallest element of the mesh.

We consider meshes generated through the choice of $f(x) = x^n$ for $1 \leq n \leq 4$ which leads to a series of increasingly anisotropic meshes. Additionally, we consider two non-polynomial mappings $f(x) = \sin(\pi x/10)$ and $f(x) = \exp(x)$. For the experiments below, two orders of geometry are considered. We first perform experiments for straight-sided meshes on which $q = 1$, so as to recover the 'classical' result that convergence order is dictated by the $(p+1)$-th order derivatives. We then consider meshes at order $q = 2$, which are constructed in a similar manner to Section 3.1 by slightly perturbing the mid point of all the interior edges. This results in a mesh where every element is curved. Since the perturbation is small, the elements remain valid (i.e. the determinant of the Jacobian is greater than 0 at all points inside every element).

3.2.3 Results for $q = 1$

In Table 1, we compare the error bounds and the interpolation error for the function $u(x) = \sin(x\pi/10) + \sin(y\pi/10)$ when it is being interpolated on the meshes described above. The results clearly indicate a monotonic relationship between the theoretical interpolation error bound and the actual error. We also verified that the bounds were respected for every point for which the error was sampled. In most cases, for high-order expansions, as the error bounds increase, the interpolation error also increases. For a given $p-$order expansion, the ratio of the bound and the error is nearly a constant. We also see that the bounds are "tighter" for the lower-order expansions than they are for higher-order expansions, i.e., the ratio of the error to the bound was close to one for lower-order expansions then the higher-order expansions.

Table 2 provides the results for interpolation of the function $u(x) = e^x + e^y$. The error is a larger value here because the function values themselves are larger, and they grow faster than the previous function, but the trends are identical. The bounds and error reduce as the order of expansion grows, and the monotonic relationship between them is maintained.

Finally, Table 3 shows the results for the polynomial function $x^6 + x^5 + x^4 + 4x^3 + x^2 + x + y^6 + y^5 + y^4 + 4y^3 + y^2 + y$. The bounds and the error are much closer to each other because the growth of the function is limited in a polynomial function due to its vanishing seventh derivative. The trend is identical to the previous two cases.

3.2.4 Results for $q = 2$

We now consider the interpolation of the same functions as in the section above, but at an increased geometric order of $q = 2$. To avoid superparametric mappings where $q > p$, we consider polynomial orders $p > 1$. The results from the experiments are reported in Tables 4, 5 and 6. The observed results are extremely similar to

the those obtained for the linear meshes above, where again the ratios of actual error and the error bounds are nearly constant for a given order p. Consequently, the theoretical error bound and the actual error are asymtotically related. We also see that the bounds are "tighter" for lower-order expansions than for higher-order expansions.

3.2.5 Summary

Our results show that the theoretical error bound and the actual error are asymptotically related, i.e., when the theoretical bounds are large or small, the actual error is also large or small, respectively. The theoretical bounds are functions of the appropriate derivatives of the function being interpolated. Thus, our paper shows that the appropriate derivative ought to be used to estimate the interpolation error and adaptively refine the mesh. The algorithm that should be to used refine the mesh, however, is an entirely different problem. Currently, the mesh refinement algorithm uses the approximate Hessian information to refine the mesh, but our paper shows that other high-order derivatives must be used to refine the mesh.

4 Conclusion and Future Work

We have provided an alternative derivation from first principles, using a Taylor series expansion, for the interpolation error bounds for curvilinear finite elements. Specifically, we have demonstrated the effect of reference-to physical mapping on the order of convergence of the interpolation error under h-refinement. We characterized and visualized the error bounds for certain types of curved high-order elements. We have shown its implications on adaptive refinement of high-order meshes through carefully designed numerical experiments. Our results indicate that the $(\lfloor p/q \rfloor + 1)$-th order partial derivative should direct adaptation for p-th order finite element meshes for q-th order geometries..

As mentioned in Sections 1 and 2, node placement in a high-order element is typically dictated by Lebesgue constant minimization. This approach assumes that the properties of the function being interpolated are unknown. In engineering applications, however, partial derivatives can be estimated through sophisticated techniques [11–15]. Thus, minimization of our error bounds is likely to produce an element whose error bounds are smaller. For some applications, such as in solid mechanics, the derivatives of the solution are more important than the solution itself. In such cases, too, minimization of our error bounds is likely to be ideal. Future research efforts should be carried out to determine suitable node placement using our error bounds.

Our bounds can be modified for Hermite finite elements in which even the partial derivatives of the solution of PDEs are computed directly from the FEM. A study of error bounds and their effect on the AMR for such meshes will be interesting for both researchers and engineers. We note that it is relatively difficult to adapt Bramble-Hilbert lemma-based proofs for anisotropic functions and Hermite elements. As the Taylor series proof is based on first principles, and it directly incorporates the use of shape functions, all the objective can be achieved relatively easily for any type of element and function.

	$p = 1$			$p = 2$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$5.0516 \cdot 10^{-4}$	$1.7983 \cdot 10^{-3}$	$2.809 \cdot 10^{-1}$	$1.2579 \cdot 10^{-6}$	$5.3251 \cdot 10^{-5}$	$2.362 \cdot 10^{-2}$
x^3	$8.3045 \cdot 10^{-4}$	$3.0258 \cdot 10^{-3}$	$2.745 \cdot 10^{-1}$	$2.7023 \cdot 10^{-6}$	$1.1519 \cdot 10^{-4}$	$2.345 \cdot 10^{-2}$
x^2	$1.5101 \cdot 10^{-3}$	$5.7049 \cdot 10^{-3}$	$2.647 \cdot 10^{-1}$	$6.8295 \cdot 10^{-6}$	$2.9422 \cdot 10^{-4}$	$2.321 \cdot 10^{-2}$
x	$3.0407 \cdot 10^{-3}$	$1.2165 \cdot 10^{-2}$	$2.500 \cdot 10^{-1}$	$2.0502 \cdot 10^{-5}$	$8.9722 \cdot 10^{-4}$	$2.285 \cdot 10^{-2}$
$\sin(\cdot)$	$4.5336 \cdot 10^{-3}$	$1.8913 \cdot 10^{-2}$	$2.397 \cdot 10^{-1}$	$3.8745 \cdot 10^{-5}$	$1.7137 \cdot 10^{-3}$	$2.260 \cdot 10^{-2}$
e^x	$7.0366 \cdot 10^{-4}$	$2.5425 \cdot 10^{-3}$	$2.768 \cdot 10^{-1}$	$2.0933 \cdot 10^{-6}$	$8.9012 \cdot 10^{-5}$	$2.351 \cdot 10^{-2}$

(a) linear and quadratic finite element meshes

	$p = 3$			$p = 4$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$5.3574 \cdot 10^{-9}$	$2.5123 \cdot 10^{-6}$	$2.132 \cdot 10^{-3}$	$1.0649 \cdot 10^{-11}$	$9.7307 \cdot 10^{-8}$	$1.094 \cdot 10^{-4}$
x^3	$1.4560 \cdot 10^{-8}$	$6.9880 \cdot 10^{-6}$	$2.083 \cdot 10^{-3}$	$3.7729 \cdot 10^{-11}$	$3.4800 \cdot 10^{-7}$	$1.084 \cdot 10^{-4}$
x^2	$4.8588 \cdot 10^{-8}$	$2.4177 \cdot 10^{-5}$	$2.009 \cdot 10^{-3}$	$1.7468 \cdot 10^{-10}$	$1.6310 \cdot 10^{-6}$	$1.070 \cdot 10^{-4}$
x	$2.0010 \cdot 10^{-7}$	$1.0543 \cdot 10^{-4}$	$1.897 \cdot 10^{-3}$	$1.0702 \cdot 10^{-9}$	$1.0171 \cdot 10^{-5}$	$1.052 \cdot 10^{-4}$
$\sin(\cdot)$	$4.5028 \cdot 10^{-7}$	$2.4739 \cdot 10^{-4}$	$1.820 \cdot 10^{-3}$	$3.0483 \cdot 10^{-9}$	$2.9321 \cdot 10^{-5}$	$1.039 \cdot 10^{-4}$
e^x	$1.0432 \cdot 10^{-8}$	$4.9652 \cdot 10^{-6}$	$2.101 \cdot 10^{-3}$	$2.4728 \cdot 10^{-11}$	$2.2737 \cdot 10^{-7}$	$1.087 \cdot 10^{-4}$

(b) cubic and quartic finite element meshes

	$p = 5$		
$f(x)$	Error	Bound	Ratio
x^4	$3.2152 \cdot 10^{-13}$	$3.2581 \cdot 10^{-9}$	$9.868 \cdot 10^{-5}$
x^3	$3.7925 \cdot 10^{-13}$	$1.4982 \cdot 10^{-8}$	$2.531 \cdot 10^{-5}$
x^2	$9.4169 \cdot 10^{-13}$	$9.5124 \cdot 10^{-8}$	$9.899 \cdot 10^{-6}$
x	$7.5297 \cdot 10^{-12}$	$8.4833 \cdot 10^{-7}$	$8.875 \cdot 10^{-6}$
$\sin(\cdot)$	$2.5532 \cdot 10^{-11}$	$3.0042 \cdot 10^{-6}$	$8.497 \cdot 10^{-6}$
e^x	$3.4927 \cdot 10^{-13}$	$9.0016 \cdot 10^{-9}$	$3.880 \cdot 10^{-5}$

(c) quintic finite element meshes

Table 1: The error bounds and the actual interpolation error over a domain with meshes of varying polynomial order p. The function being interpolated is $u(x) = \sin\left(\frac{x\pi}{10}\right) + \sin\left(\frac{y\pi}{10}\right)$.

Finally, we note that there has been little effort to characterize the conditioning of the stiffness matrix as a function of the (geometric) curvature of a triangle for high-order elements. Several open problems in the area need to addressed for the high-order FEM to be used for engineering applications, and in theory our perspective on the error bounds facilitates this type of study.

Acknowledgment

	$p = 1$			$p = 2$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$1.1958 \cdot 10^{-1}$	$1.8313 \cdot 10^{-1}$	$6.529 \cdot 10^{-1}$	$1.2459 \cdot 10^{-3}$	$5.4227 \cdot 10^{-3}$	$2.297 \cdot 10^{-1}$
x^3	$1.9330 \cdot 10^{-1}$	$3.0274 \cdot 10^{-1}$	$6.385 \cdot 10^{-1}$	$2.5974 \cdot 10^{-3}$	$1.1526 \cdot 10^{-2}$	$2.253 \cdot 10^{-1}$
x^2	$3.4231 \cdot 10^{-1}$	$5.5557 \cdot 10^{-1}$	$6.161 \cdot 10^{-1}$	$6.2611 \cdot 10^{-3}$	$2.8653 \cdot 10^{-2}$	$2.185 \cdot 10^{-1}$
x	$6.6066 \cdot 10^{-1}$	1.1361	$5.815 \cdot 10^{-1}$	$1.7413 \cdot 10^{-2}$	$8.3797 \cdot 10^{-2}$	$2.077 \cdot 10^{-1}$
$\sin(\cdot)$	$9.5474 \cdot 10^{-1}$	1.7147	$5.567 \cdot 10^{-1}$	$3.1083 \cdot 10^{-2}$	$1.5536 \cdot 10^{-1}$	$2.000 \cdot 10^{-1}$
e^x	$1.6478 \cdot 10^{-1}$	$2.5600 \cdot 10^{-1}$	$6.436 \cdot 10^{-1}$	$2.0339 \cdot 10^{-3}$	$8.9626 \cdot 10^{-3}$	$2.269 \cdot 10^{-1}$

(a) linear and quadratic finite element meshes

	$p = 3$			$p = 4$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$1.2844 \cdot 10^{-5}$	$2.5583 \cdot 10^{-4}$	$5.020 \cdot 10^{-2}$	$1.0614 \cdot 10^{-2}$	$9.9091 \cdot 10^{-1}$	$1.071 \cdot 10^{-2}$
x^3	$3.4312 \cdot 10^{-5}$	$6.9918 \cdot 10^{-4}$	$4.907 \cdot 10^{-2}$	$3.6498 \cdot 10^{-7}$	$3.4819 \cdot 10^{-5}$	$1.048 \cdot 10^{-2}$
x^2	$1.1143 \cdot 10^{-4}$	$2.3545 \cdot 10^{-3}$	$4.732 \cdot 10^{-2}$	$1.6086 \cdot 10^{-6}$	$1.5884 \cdot 10^{-4}$	$1.012 \cdot 10^{-2}$
x	$4.3920 \cdot 10^{-4}$	$9.8473 \cdot 10^{-3}$	$4.460 \cdot 10^{-2}$	$9.0936 \cdot 10^{-6}$	$9.5001 \cdot 10^{-4}$	$9.572 \cdot 10^{-3}$
$\sin(\cdot)$	$9.5647 \cdot 10^{-4}$	$2.2429 \cdot 10^{-2}$	$4.264 \cdot 10^{-2}$	$2.4383 \cdot 10^{-5}$	$2.6583 \cdot 10^{-3}$	$9.172 \cdot 10^{-3}$
e^x	$2.4737 \cdot 10^{-5}$	$4.9994 \cdot 10^{-4}$	$4.947 \cdot 10^{-2}$	$2.4187 \cdot 10^{-7}$	$2.2894 \cdot 10^{-5}$	$1.056 \cdot 10^{-2}$

(b) cubic and quartic finite element meshes

	$p = 5$		
$f(x)$	Error	Bound	Ratio
x^4	$7.8782 \cdot 10^{-10}$	$3.3179 \cdot 10^{-7}$	$2.374 \cdot 10^{-3}$
x^3	$3.4779 \cdot 10^{-9}$	$1.4990 \cdot 10^{-6}$	$2.320 \cdot 10^{-3}$
x^2	$2.0724 \cdot 10^{-8}$	$9.2636 \cdot 10^{-6}$	$2.237 \cdot 10^{-3}$
x	$1.6701 \cdot 10^{-7}$	$7.9231 \cdot 10^{-5}$	$2.107 \cdot 10^{-3}$
$\sin(\cdot)$	$5.4881 \cdot 10^{-7}$	$2.7236 \cdot 10^{-4}$	$2.015 \cdot 10^{-3}$
e^x	$2.1204 \cdot 10^{-9}$	$9.0636 \cdot 10^{-7}$	$2.339 \cdot 10^{-3}$

(c) quintic finite element meshes

Table 2: The error bounds and the actual interpolation error over a domain with meshes of varying polynomial order p. The function being interpolated is $u(x) = e^x + e^y$.

References

1. L. Botti, "Influence of reference-to-physical frame mappings on approximation properties of discontinuous piecewise polynomial spaces," *Journal of Scientific Computing*, vol. 52, no. 3, pp. 675–703, 2012.
2. J. H. Bramble and S. R. Hilbert, "Estimation of linear functionals on sobolev spaces with application to fourier transforms and spline interpolation," *SIAM Journal on Numerical Analysis*, vol. 7, no. 1, pp. 112–124, 1970.
3. T. Dupont and R. Scott, "Polynomial approximation of functions in sobolev spaces," *Mathematics of Computation*, vol. 34, no. 150, pp. 441–463, 1980.

$f(x)$	$p = 1$			$p = 2$		
	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$1.8406 \cdot 10^1$	$1.9010 \cdot 10^1$	$9.682 \cdot 10^{-1}$	$2.4130 \cdot 10^{-1}$	$5.6290 \cdot 10^{-1}$	$4.286 \cdot 10^{-1}$
x^3	$2.9577 \cdot 10^1$	$3.1427 \cdot 10^1$	$9.411 \cdot 10^{-1}$	$5.0344 \cdot 10^{-1}$	1.1964	$4.207 \cdot 10^{-1}$
x^2	$5.1866 \cdot 10^1$	$5.7671 \cdot 10^1$	$8.993 \cdot 10^{-1}$	2.1463	2.9743	$7.216 \cdot 10^{-1}$
x	$9.8457 \cdot 10^1$	$1.1794 \cdot 10^2$	$8.348 \cdot 10^{-1}$	3.8058	8.6987	$4.375 \cdot 10^{-1}$
$\sin(\cdot)$	$1.4039 \cdot 10^2$	$1.7800 \cdot 10^2$	$7.887 \cdot 10^{-1}$	3.4262	$1.6128 \cdot 10^1$	$2.124 \cdot 10^{-1}$
e^x	$2.5268 \cdot 10^1$	$2.6574 \cdot 10^1$	$9.508 \cdot 10^{-1}$	$3.9412 \cdot 10^{-1}$	$9.3037 \cdot 10^{-1}$	$4.236 \cdot 10^{-1}$

(a) linear and quadratic finite element meshes

$f(x)$	$p = 3$			$p = 4$		
	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$2.3854 \cdot 10^{-3}$	$2.6557 \cdot 10^{-2}$	$8.982 \cdot 10^{-2}$	$1.2688 \cdot 10^{-5}$	$1.0286 \cdot 10^{-3}$	$1.233 \cdot 10^{-2}$
x^3	$6.4234 \cdot 10^{-3}$	$7.2579 \cdot 10^{-2}$	$8.850 \cdot 10^{-2}$	$4.4270 \cdot 10^{-5}$	$3.6144 \cdot 10^{-3}$	$1.224 \cdot 10^{-2}$
x^2	$2.1121 \cdot 10^{-2}$	$2.4441 \cdot 10^{-1}$	$8.641 \cdot 10^{-2}$	$1.9966 \cdot 10^{-4}$	$1.6488 \cdot 10^{-2}$	$1.210 \cdot 10^{-2}$
x	$8.4889 \cdot 10^{-2}$	1.0222	$8.304 \cdot 10^{-2}$	$1.1716 \cdot 10^{-3}$	$9.8617 \cdot 10^{-2}$	$1.188 \cdot 10^{-2}$
$\sin(\cdot)$	$1.8751 \cdot 10^{-1}$	2.3283	$8.053 \cdot 10^{-2}$	$3.2306 \cdot 10^{-3}$	$2.7595 \cdot 10^{-1}$	$1.170 \cdot 10^{-2}$
e^x	$4.6177 \cdot 10^{-3}$	$5.1897 \cdot 10^{-2}$	$8.897 \cdot 10^{-2}$	$2.9184 \cdot 10^{-5}$	$2.3765 \cdot 10^{-3}$	$1.228 \cdot 10^{-2}$

(b) cubic and quartic finite element meshes

$f(x)$	$p = 5$		
	Error	Bound	Ratio
x^4	$3.0600 \cdot 10^{-8}$	$3.4441 \cdot 10^{-5}$	$8.884 \cdot 10^{-4}$
x^3	$1.3819 \cdot 10^{-8}$	$1.5560 \cdot 10^{-4}$	$8.881 \cdot 10^{-4}$
x^2	$8.5389 \cdot 10^{-8}$	$9.6161 \cdot 10^{-4}$	$8.879 \cdot 10^{-4}$
x	$7.3032 \cdot 10^{-7}$	$8.2247 \cdot 10^{-3}$	$8.879 \cdot 10^{-4}$
$\sin(\cdot)$	$2.5105 \cdot 10^{-6}$	$2.8273 \cdot 10^{-2}$	$8.879 \cdot 10^{-4}$
e^x	$8.3561 \cdot 10^{-8}$	$9.4085 \cdot 10^{-5}$	$8.881 \cdot 10^{-4}$

(c) quintic finite element meshes

Table 3: The error bounds and the actual interpolation error over a domain with meshes of varying polynomial order p. The function being interpolated is $u(x) = x^6 + x^5 + x^4 + 4x^3 + x^2 + x + y^6 + y^5 + y^4 + 4y^3 + y^2 + y$.

4. D. N. Arnold, D. Boffi, and R. S. Falk, "Approximation by quadrilateral finite elements," *Math. Comp.*, vol. 71, no. 239, pp. 909–922, 2002.
5. S. Dekel and D. Levitan, "The bramble-hilbert lemma for complex domains," *SIAM J. Math. Anal.*, vol. 35, no. 5, pp. 1103–1112, 2004.
6. S. P. Sastry and R. M. Kirby, "On interpolation errors over quadratic nodal triangular finite elements," in *Proceedings of the 22nd International Meshing Roundtable* (J. Sarrate and M. Staten, eds.), pp. 349–366, Springer International Publishing, 2013.
7. C. Johnson, *Numerical Solution of Partial Differential Equations by the Finite Element Method.* Cambridge, England: Cambridge University Press, 1987.
8. G. E. Karniadakis and S. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics.* OUP, 2nd ed., 2005.
9. J. S. Hesthaven, "From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex," *SIAM J. Numer. Anal.*, vol. 35, pp. 655–676, 1998.
10. C. D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. de Grazia, S. Yakovlev, J.-E. Lombard, D. Ekelschot, B. Jordi, H. Xu, Y. Mohamied, C. Eskilsson, B. Nelson, P. Vos, C. Biotto, R. M. Kirby, and S. J. Sherwin, "Nektar++: An open-source spectral/hp element framework," *Comp. Phys. Comm.*, vol. 192, pp. 205–219, 2015.

	$p = 2$			$p = 3$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$2.7367 \cdot 10^{-4}$	$6.6713 \cdot 10^{-4}$	$4.102 \cdot 10^{-1}$	$1.0760 \cdot 10^{-4}$	$7.1280 \cdot 10^{-4}$	$1.510 \cdot 10^{-1}$
x^3	$4.5058 \cdot 10^{-4}$	$1.1213 \cdot 10^{-3}$	$4.018 \cdot 10^{-1}$	$1.8627 \cdot 10^{-4}$	$1.1934 \cdot 10^{-3}$	$1.561 \cdot 10^{-1}$
x^2	$8.2179 \cdot 10^{-4}$	$2.0902 \cdot 10^{-3}$	$3.932 \cdot 10^{-1}$	$3.5739 \cdot 10^{-4}$	$2.2263 \cdot 10^{-3}$	$1.605 \cdot 10^{-1}$
x	$1.7668 \cdot 10^{-3}$	$4.3163 \cdot 10^{-3}$	$4.093 \cdot 10^{-1}$	$7.4876 \cdot 10^{-4}$	$4.6165 \cdot 10^{-3}$	$1.622 \cdot 10^{-1}$
$\sin(\cdot)$	$2.6894 \cdot 10^{-3}$	$6.4607 \cdot 10^{-3}$	$4.163 \cdot 10^{-1}$	$1.0996 \cdot 10^{-3}$	$6.9221 \cdot 10^{-3}$	$1.588 \cdot 10^{-1}$
e^x	$3.8076 \cdot 10^{-4}$	$9.4667 \cdot 10^{-4}$	$4.022 \cdot 10^{-1}$	$1.5679 \cdot 10^{-4}$	$1.0094 \cdot 10^{-3}$	$1.553 \cdot 10^{-1}$

(a) quadratic and cubic finite element meshes

	$p = 4$			$p = 5$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$5.1423 \cdot 10^{-7}$	$9.1774 \cdot 10^{-6}$	$5.603 \cdot 10^{-2}$	$2.1453 \cdot 10^{-7}$	$1.0946 \cdot 10^{-5}$	$1.960 \cdot 10^{-2}$
x^3	$1.0924 \cdot 10^{-6}$	$2.0145 \cdot 10^{-5}$	$5.423 \cdot 10^{-2}$	$4.4396 \cdot 10^{-7}$	$2.4029 \cdot 10^{-5}$	$1.848 \cdot 10^{-2}$
x^2	$2.7401 \cdot 10^{-6}$	$5.2170 \cdot 10^{-5}$	$5.252 \cdot 10^{-2}$	$1.0519 \cdot 10^{-6}$	$6.2255 \cdot 10^{-5}$	$1.690 \cdot 10^{-2}$
x	$8.1510 \cdot 10^{-6}$	$1.6064 \cdot 10^{-4}$	$5.074 \cdot 10^{-2}$	$2.8435 \cdot 10^{-6}$	$1.9207 \cdot 10^{-4}$	$1.480 \cdot 10^{-2}$
$\sin(\cdot)$	$1.5162 \cdot 10^{-5}$	$3.0584 \cdot 10^{-4}$	$4.958 \cdot 10^{-2}$	$4.6923 \cdot 10^{-6}$	$3.6767 \cdot 10^{-4}$	$1.276 \cdot 10^{-2}$
e^x	$8.5210 \cdot 10^{-7}$	$1.5530 \cdot 10^{-5}$	$5.487 \cdot 10^{-2}$	$3.5501 \cdot 10^{-7}$	$1.8501 \cdot 10^{-5}$	$1.919 \cdot 10^{-2}$

(b) quartic and quintic finite element meshes

Table 4: The error bounds and the actual interpolation error over a domain with curved meshes of varying polynomial order p. The function being interpolated is $u(x) = \sin\left(\frac{x\pi}{10}\right) + \sin\left(\frac{y\pi}{10}\right)$.

11. G. Kunert, "Toward anisotropic mesh construction and error estimation in the finite element method," *Numerical Methods for Partial Differential Equations*, vol. 18, no. 5, pp. 625–648, 2002.

12. L. Kamenski, *Anisotropic Mesh Adaptation Based on Hessian Recovery and A Posteriori Error Estimates.* PhD thesis, Technischen Universität Darmstadt, 2009.

13. M. Ainsworth and J. Oden, "A posteriori error estimation in finite element analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 142, no. 12, pp. 1 – 88, 1997.

14. T. Apel, *Anisotropic Finite Elements: Local Estimates and Applications.* Stuttgart, Leipzig: B.G. Teubner, 1999.

15. J. S. Ovall, "Function, gradient, and hessian recovery using quadratic edgebump functions," *SIAM Journal on Numerical Analysis*, vol. 45, no. 3, pp. 1064–1080, 2007.

	$p = 2$			$p = 3$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$6.4982 \cdot 10^{-2}$	$1.6781 \cdot 10^{-1}$	$2.872 \cdot 10^{-1}$	$2.5493 \cdot 10^{-2}$	$1.7930 \cdot 10^{-1}$	$1.422 \cdot 10^{-1}$
x^3	$1.0528 \cdot 10^{-1}$	$2.8207 \cdot 10^{-1}$	$3.733 \cdot 10^{-1}$	$4.3512 \cdot 10^{-2}$	$3.0021 \cdot 10^{-1}$	$1.449 \cdot 10^{-1}$
x^2	$1.8725 \cdot 10^{-1}$	$5.2580 \cdot 10^{-1}$	$3.561 \cdot 10^{-1}$	$8.1472 \cdot 10^{-2}$	$5.6003 \cdot 10^{-1}$	$1.455 \cdot 10^{-1}$
x	$3.8358 \cdot 10^{-1}$	1.0857	$3.533 \cdot 10^{-1}$	$1.6364 \cdot 10^{-1}$	1.1613	$1.409 \cdot 10^{-1}$
$\sin(\cdot)$	$5.6426 \cdot 10^{-1}$	1.6252	$3.472 \cdot 10^{-1}$	$2.3244 \cdot 10^{-1}$	1.7412	$1.335 \cdot 10^{-1}$
e^x	$8.9481 \cdot 10^{-2}$	$2.3813 \cdot 10^{-1}$	$3.758 \cdot 10^{-1}$	$3.6826 \cdot 10^{-2}$	$2.5391 \cdot 10^{-1}$	$1.450 \cdot 10^{-1}$

(a) quadratic and cubic finite element meshes

	$p = 4$			$p = 5$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$5.0731 \cdot 10^{-4}$	$9.4691 \cdot 10^{-3}$	$5.358 \cdot 10^{-2}$	$2.0494 \cdot 10^{-4}$	$1.1294 \cdot 10^{-2}$	$1.815 \cdot 10^{-2}$
x^3	$1.0459 \cdot 10^{-3}$	$2.0422 \cdot 10^{-2}$	$5.121 \cdot 10^{-2}$	$4.0313 \cdot 10^{-4}$	$2.4360 \cdot 10^{-2}$	$1.655 \cdot 10^{-2}$
x^2	$2.5010 \cdot 10^{-3}$	$5.1476 \cdot 10^{-2}$	$4.859 \cdot 10^{-2}$	$8.7661 \cdot 10^{-4}$	$6.1428 \cdot 10^{-2}$	$1.427 \cdot 10^{-2}$
x	$6.8745 \cdot 10^{-3}$	$1.5201 \cdot 10^{-1}$	$4.522 \cdot 10^{-2}$	$2.0356 \cdot 10^{-3}$	$1.8176 \cdot 10^{-1}$	$1.120 \cdot 10^{-2}$
$\sin(\cdot)$	$1.2006 \cdot 10^{-2}$	$2.8094 \cdot 10^{-1}$	$4.274 \cdot 10^{-2}$	$2.9168 \cdot 10^{-3}$	$3.3774 \cdot 10^{-1}$	$8.636 \cdot 10^{-3}$
e^x	$8.2486 \cdot 10^{-4}$	$1.5844 \cdot 10^{-2}$	$5.206 \cdot 10^{-2}$	$3.2876 \cdot 10^{-4}$	$1.8876 \cdot 10^{-2}$	$1.742 \cdot 10^{-2}$

(b) quartic and quintic finite element meshes

Table 5: The error bounds and the actual interpolation error over a domain with curved meshes of varying polynomial order p. The function being interpolated is $u(x) = e^x + e^y$.

	$p = 2$			$p = 3$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$1.0011 \cdot 10^1$	$2.6335 \cdot 10^1$	$3.801 \cdot 10^{-1}$	3.9247	$2.8138 \cdot 10^1$	$1.395 \cdot 10^{-2}$
x^3	$1.6129 \cdot 10^1$	$4.4265 \cdot 10^1$	$3.644 \cdot 10^{-1}$	6.6660	$4.7112 \cdot 10^1$	$1.415 \cdot 10^{-2}$
x^2	$2.8424 \cdot 10^1$	$8.2514 \cdot 10^1$	$3.445 \cdot 10^{-1}$	$1.2371 \cdot 10^1$	$8.7885 \cdot 10^1$	$1.408 \cdot 10^{-2}$
x	$5.7127 \cdot 10^1$	$1.7039 \cdot 10^2$	$3.353 \cdot 10^{-1}$	$2.4447 \cdot 10^1$	$1.8224 \cdot 10^2$	$1.341 \cdot 10^{-2}$
$\sin(\cdot)$	$8.2804 \cdot 10^1$	$2.5504 \cdot 10^2$	$3.247 \cdot 10^{-1}$	$3.4241 \cdot 10^1$	$2.7325 \cdot 10^2$	$1.253 \cdot 10^{-2}$
e^x	$1.3736 \cdot 10^1$	$3.7370 \cdot 10^1$	$3.676 \cdot 10^{-1}$	5.6524	$3.9847 \cdot 10^1$	$1.419 \cdot 10^{-2}$

(a) quadratic and cubic finite element meshes

	$p = 4$			$p = 5$		
$f(x)$	Error	Bound	Ratio	Error	Bound	Ratio
x^4	$9.8301 \cdot 10^{-2}$	1.8273	$5.380 \cdot 10^{-2}$	$3.9751 \cdot 10^{-2}$	2.1794	$1.824 \cdot 10^{-2}$
x^3	$2.0282 \cdot 10^{-1}$	3.9410	$5.146 \cdot 10^{-2}$	$7.8280 \cdot 10^{-2}$	4.7009	$1.665 \cdot 10^{-2}$
x^2	$4.8546 \cdot 10^{-1}$	9.9337	$4.887 \cdot 10^{-2}$	$1.7040 \cdot 10^{-1}$	$1.1854 \cdot 10^1$	$1.438 \cdot 10^{-2}$
x	1.3354	$2.9335 \cdot 10^1$	$4.552 \cdot 10^{-2}$	$3.9539 \cdot 10^{-1}$	$3.5075 \cdot 10^1$	$1.127 \cdot 10^{-2}$
$\sin(\cdot)$	2.3318	$5.4215 \cdot 10^1$	$4.301 \cdot 10^{-2}$	$5.6445 \cdot 10^{-1}$	$6.5175 \cdot 10^1$	$8.660 \cdot 10^{-3}$
e^x	$1.5991 \cdot 10^{-1}$	3.0575	$5.230 \cdot 10^{-2}$	$6.3813 \cdot 10^{-2}$	3.6424	$1.752 \cdot 10^{-2}$

(b) quartic and quintic finite element meshes

Table 6: The error bounds and the actual interpolation error over a domain with curved meshes of varying polynomial order p. The function being interpolated is $u(x) = x^6 + x^5 + x^4 + 4x^3 + x^2 + x + y^6 + y^5 + y^4 + 4y^3 + y^2 + y$.