INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 3, Number 1, Pages 21–51

EFFECTS OF BASIS SELECTION AND H-REFINEMENT ON ERROR ESTIMATOR RELIABILITY AND SOLUTION EFFICIENCY FOR HIGH-ORDER METHODS IN THREE SPACE DIMENSIONS

PETER K. MOORE

(Communicated by Zhimin Zhang)

Abstract. Designing effective high-order adaptive methods for solving stationary reaction-diffusion equations in three dimensions requires the selection of a finite element basis, *a posteriori* error estimator and refinement strategy. Estimator accuracy may depend on the basis chosen, which in turn, may lead to unreliability or inefficiency via under- or over-refinement, respectively. The basis may also have an impact on the size and condition of the matrices that arise from discretization, and thus, on algorithm effectiveness. Herein, the interaction between these three components is studied in the context of an h-refinement procedure. The effects of these choices on the robustness and efficiency of the algorithm are examined for several linear and nonlinear problems. The results demonstrate that popular choices such as the tensor-product basis or the modified Szabó-Babuška basis have significant shortcomings but that promising alternatives exist.

Key Words. *a posteriori* error estimation, adaptivity, high-order finite element basis.

1. Introduction

Adaptive finite element methods have become ubiquitous in solving systems of nonlinear partial differential equations [6]. Two advantages of these methods are reliability and efficiency. For high-order methods the choice of the finite element space and the performance of *a posteriori* error estimators are two key elements in achieving these goals. An appropriate finite element space should be as small as possible so as to minimize the number of degrees of freedom while maintaining the appropriate order. At the same time it should lead to well-conditioned Jacobian matrices. The adaptive strategy should seek to avoid over- and under-refinement. The former leads to inefficiency while the latter undermines reliability. *A posteriori* error estimates that are asymptotically exact, that is, the error estimates converge to the true error as the grid is refined, help to alleviate these problems. Additionally such estimates should be cheap to compute. The refinement strategy and finite element space, in turn, may effect the performance of the error estimator [3].

In three dimensions the tensor-product (TP) basis on hexahedral elements represents a natural choice [10, 13, 26]. Since this basis is significantly larger than

Received by the editors November 23, 2004 and, in revised form, April 12, 2005.
2000 *Mathematics Subject Classification.* 65N15, 65N30, 65N50.
This research is partially supported by NSF Grant #DMS-0203154.

necessary to obtain an approximation of order p other bases have been introduced called variously hierarchical or serendipity bases [10, 13, 24, 30]. One of the most popular of these bases is the hierarchical basis of Szabó and Babuška [13, 24], henceforth referred to as the SBH basis. However, the asymptotic equivalence property, critical in obtaining hierarchical *a posteriori* error estimates (cf. below and section 3) does not hold for this basis [17]. As a result in two dimensions Adjerid *et al* [2] introduced a modified-SB-hierarchical (mSBH) basis. Their basis adds the minimal number of basis functions to the SBH basis to obtain asymptotic equivalence and is easily extended to three dimensions. As the order p increases the number of possible bases between the mSBH and TP bases also increases. The examples in section 5 show that the TP and mSBH bases are unsatisfactory. In this paper the impact of choosing bases between the mSBH and TP bases on an *a posteriori* error estimator and on solution efficiency is studied.

Numerous *a posteriori* error estimation strategies have been proposed and implemented [4, 6, 25]. One widely-studied class is the hierarchical estimators [4, 25]. Among these estimators is a family of methods that rely on finding an interpolant that is asymptotically equivalent to the finite element solution in the sense of Adjerid *et al* [2]; that is, the finite element and interpolation errors converge at the same rate with the same constant in H^1. Asymptotic equivalence leads to asymptotically exact error estimates. Yu [28] and Adjerid *et al* [2] developed such an estimator for p even. Moore [19] extended their estimator to all orders $p > 1$. Many of the convergence proofs assume an idealized (e.g. uniform) grid. An extension of the estimator to adaptive grids using interpolation error estimates is proposed (cf. section 4 and Appendix B). The influence of an h-refinement strategy on estimator reliability is examined herein.

The h-refinement adaptive code is used to solve reaction-diffusion equations of the form

$$-\Delta u = F(u, \mathbf{x}),\ \mathbf{x} = (x, y, z) \in \Omega \equiv (X_0, X_1) \times (Y_0, Y_1) \times (Z_0, Z_1), \tag{1}$$

together with Dirichlet and/or Neumann boundary conditions. Throughout I assume that (1) has a unique solution of appropriate smoothness. Equation (1) is discretized using the finite element-Galerkin method with a piecewise continuous polynomial basis of degree $p > 1$. The resulting nonlinear system is solved via Newton's method. GMRES [23] with ILUT preconditioning [22] is used to solve the linear systems.

The definition of the admissible grids and rules governing h-refinement are presented in section 2. The finite element discretization is described in section 3. In section 4 the *a posteriori* error estimators are introduced along with a convergence result on h-refined grids. Computational results for three problems are given in section 5. Some conclusions are presented in section 6.

2. Grid definition and h-refinement

The grid, Δ_Ω, for Ω is obtained by recursive trisection, beginning with Ω. Thus, the grid has an octree structure with the root corresponding to Ω. The leaf vertices of the tree are called elements (unrefined elements in [27]). Each element in the grid is made up of element interiors, faces, edges and nodes. These four building blocks are henceforth referred to collectively as **components**. The level of an element in the grid is the length of the path from the root to the element. A vertex with eight subvertices is referred to as a parent vertex and the eight subvertices are its offspring or children. Eight vertices having a common parent are called siblings. The 64 vertices having a common grandparent are called cousins. A grid is said

to be uniform if all its elements are at the same level. A grid is admissible in the sense of Babuška and Rheinboldt [7, 8] if it is defined recursively by the following two rules:

(1) Ω is an admissible grid;
(2) If Δ_Ω is an admissible grid and $\tilde{\Delta}$ is an element of Δ_Ω then the grid obtained from Δ_Ω and the eight elements created by trisecting $\tilde{\Delta}$ is admissible.

Such grids contain two general types of components **regular** and **irregular**. A node Γ of Δ_Ω is regular if for every element $\tilde{\Delta} \in \Delta_\Omega$ such that $\Gamma \in \tilde{\Delta}$, Γ is a corner node of $\tilde{\Delta}$. An edge Γ of Δ_Ω is regular if for every element $\tilde{\Delta} \in \Delta_\Omega$ such that $\Gamma \in \tilde{\Delta}$ its endpoints are corners of $\tilde{\Delta}$. Similarly a face Γ in Δ_Ω is regular if for every element $\tilde{\Delta} \in \Delta_\Omega$ such that $\Gamma \in \tilde{\Delta}$, the four corners of Γ are also corners of $\tilde{\Delta}$. All element interiors are regular. Any component that is not regular is irregular. Irregular components appear at the interface of two elements at different levels in the grid (cf. Figure 1). Two elements Δ' and Δ'' are said to be neighbors if their intersection contains an edge of either Δ' or Δ''.

Several additional rules for governing grid refinement and coarsening have been proposed [9, 15, 27]. The one-irregular rule dictates that on any edge or in the interior of any face there can be no more than one irregular node. Equivalently, two neighboring elements cannot differ by more than one level in the tree. A second rule is the k-neighbor rule. In three dimensions this rule has two variants. In both cases an element at level l is refined if k of its neighbors are at level $l+1$ where neighbors are further classified as face or edge neighbors. Additional details can be found in [16]. Finally the sibling rule requires that all siblings of an element selected for refinement must also be refined. It was introduced in [15] to enable explicit error estimates to be constructed. The k-neighbor and sibling rules tend to reduce the number of irregular components in the grid.

Throughout let Δ represent the canonical element

$$\Delta = [m^x - \frac{h^x}{2}, m^x + \frac{h^x}{2}] \times [m^y - \frac{h^y}{2}, m^y + \frac{h^y}{2}] \times [m^z - \frac{h^z}{2}, m^z + \frac{h^z}{2}]. \tag{2}$$

The elements of Δ_Ω are denoted by

$$\begin{aligned} \Delta_i &= [m_i^x - \frac{h_i^x}{2}, m_i^x + \frac{h_i^x}{2}] \times [m_i^y - \frac{h_i^y}{2}, m_i^y + \frac{h_i^y}{2}] \times [m_i^z - \frac{h_i^z}{2}, m_i^z + \frac{h_i^z}{2}], \\ k &= 1, \ldots, N_{el}, \end{aligned} \tag{3}$$

where (m_i^x, m_i^y, m_i^z) and (h_i^x, h_i^y, h_i^z) are the center and spacing of Δ_i, respectively, and N_{el} is the number of elements.

Following Yu [28] Ω' is a subdomain of Ω if

$$\begin{aligned} \Omega' = (X_0', X_1') \times (Y_0', Y_1') \times (Z_0', Z_1'), \qquad & X_0 \le X_0' < X_1' \le X_1, \\ Y_0 \le Y_0' < Y_1' \le Y_1, \qquad & Z_0 \le Z_0' < Z_1' \le Z_1. \end{aligned} \tag{4}$$

Additionally a subdomain Ω' is called a uniform patch of Ω [28] if there is an index set $\Upsilon' \subseteq \{1, \ldots, N_{el}\}$ such that:

(1) $\Omega' = \cup_{i \in \Upsilon'} \Delta_i$;
(2) $|\Delta_i| = H_{\Omega'}(\text{constant}),\ \forall i \in \Upsilon'$;
(3) If Γ is a node of Δ_i, $i \in \Upsilon'$ then Γ is regular.

Throughout the remainder of the paper let the uniform patches Ω' and Ω'' be such that $\bar{\Omega}' \subset \Omega''$ and $H_{\Omega'} = H_{\Omega''}$.

The error estimates of section 4 depend on the classification of the irregular nodes of the elements based on their number and location.

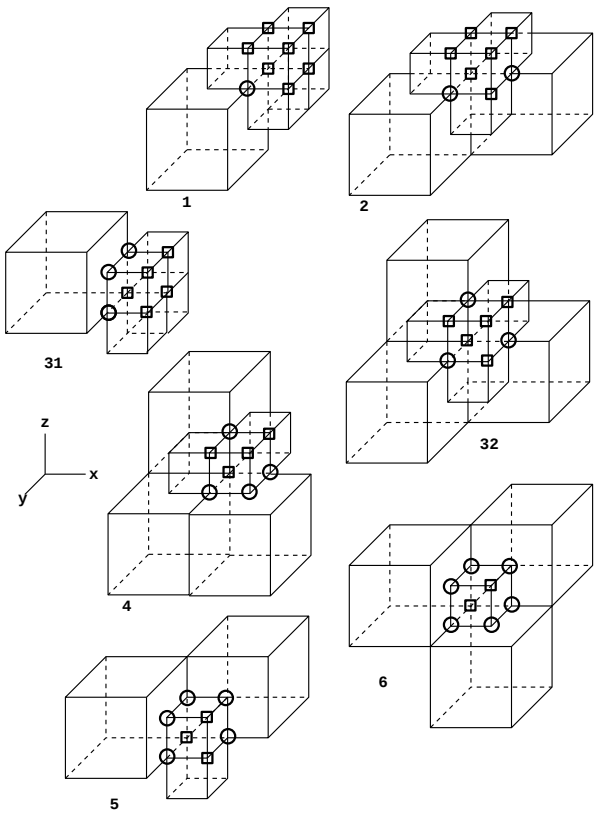

FIGURE 1. The seven cases of irregular nodes based on their number and location in an element. Irregular nodes are indicated by circles, regular nodes by squares.

Lemma 1. *Let Δ_Ω be a grid obtained from Ω by recursive trisection and the one-irregular rule. Let $\Delta_i \in \Delta_\Omega$ be an element and let*

$$\gamma_{\Delta_i} = \{\Gamma \mid \Gamma \text{ is an irregular node of } \Delta_i\}. \tag{5}$$

Then either Δ_i has no irregular nodes or, up to symmetry, one of the seven cases in Figure 1 must hold.

Proof. [15]. □

Remark. The location and number of the irregular nodes determines the location and number of the irregular edges and faces.

3. Finite element discretization

The three-dimensional bases depend on the one-dimensional Lobatto polynomials on the interval $\Lambda = [m - h/2, m + h/2]$ given by

$$\Phi_p(\xi; h; m) = \sqrt{\frac{2p-1}{2}} \int_{-1}^{2(\xi-m)/h} P_{p-1}(s)ds, \ p \geq 2, \tag{6}$$

where $P_{p-1}(s)$ is the Legendre polynomial of degree $p-1$. The Lobatto polynomials along with the linear hat functions

$$\Phi_0(\xi;h;m) \;=\; \frac{h/2-(\xi-m)}{h},\; \Phi_1(\xi;h;m) \;=\; \frac{h/2+(\xi-m)}{h}, \tag{7}$$

form a basis for the space of polynomials of degree p on Λ. Henceforth I assume $p>1$.

In moving to three dimensions it is helpful to consider the following index sets:

$$\mathcal{S}^{p,e,f} \doteq \mathcal{S}^N \cup \mathcal{S}^{E,p} \cup \mathcal{S}^{F,p,f} \cup \mathcal{S}^{El,p,e}, \tag{8}$$

where

$$\begin{aligned}
\mathcal{S}^N &\doteq \{(i,j,k) \mid 0\le i,j,k\le 1\}, && (9)\\
\mathcal{S}^{E,p} &\doteq \{(i,j,k) \mid 0\le i,j\le 1,\ 2\le k\le p\}\\
&\cup \{(i,j,k) \mid 0\le i,k\le 1,\ 2\le j\le p\}\\
&\cup \{(i,j,k) \mid 0\le j,k\le 1,\ 2\le i\le p\}, && (10)\\
\mathcal{S}^{F,p,f} &\doteq \{(i,j,k) \mid 0\le i\le 1,\ 2\le j,k\le p,\ j+k\le f\}\\
&\cup \{(i,j,k) \mid 0\le j\le 1,\ 2\le i,k\le p,\ i+k\le f\}\\
&\cup \{(i,j,k) \mid 0\le k\le 1,\ 2\le i,j\le p,\ i+j\le f\}, && (11)\\
\mathcal{S}^{El,p,e} &\doteq \{(i,j,k) \mid 2\le i,j,k\le p,\ i+j+k\le e\}. && (12)
\end{aligned}$$

The corresponding spaces on Δ,

$$\begin{aligned}
S_\Delta^{p,e,f} &\doteq \operatorname{span}\{x^i y^j z^k \mid (i,j,k)\in\mathcal{S}^{p,e,f}\} && (13)\\
&\equiv \operatorname{span}\{\Phi_i(x;h^x;m^x)\Phi_j(y;h^y;m^y)\Phi_k(z;h^z;m^z) \mid (i,j,k)\in\mathcal{S}^{p,e,f}\}, && (14)
\end{aligned}$$

are symmetric in the sense that additional basis functions of the appropriate order are added to each edge and face. The use of symmetry results in a more natural basis and is therefore easier to implement. The spaces

$$S_\Delta^\tau \doteq \operatorname{span}\{\Phi_i(x;h^x;m^x)\Phi_j(y;h^y;m^y)\Phi_k(z;h^z;m^z) \mid (i,j,k)\in\mathcal{S}^\tau\}, \tag{15}$$

where $\tau = N;\ E,p;\ F,p,f;$ and $El,p,e;$ correspond to the spaces spanned by the node, edge, face and interior basis functions on Δ. The basis functions (13) necessary to obtain an approximation of order p can be displayed in three dimensions as a Pascal's tetrahedron (Pascal's triangle in two dimensions [13]) where horizontal planes intersecting the tetrahedron yield all basis functions of degree r, $0\le r\le p$. A truncated Pascal's tetrahedron for $p=9$ containing all basis functions for the TP basis of order $p=3$ is shown in Figure 2. An equivalent tetrahedron can be constructed with $x^iy^jz^k$ replaced by $\Phi_i(x;h^x;m^x)\Phi_j(y;h^y;m^y)\Phi_k(z;h^z;m^z)$.

To guarantee asymptotic equivalence on uniform grids e and f must satisfy (cf. Theorem 1):

$$e\in\begin{cases}\{0,6\} & p=2,\\ \{0,6,7,\dots,3p\} & p=3,4,\\ \{p+1,\dots,3p\} & p>4,\end{cases}\quad f\in\begin{cases}\{0,4\} & p=2,\\ \{p+1,\dots,2p\} & p>2.\end{cases} \tag{16}$$

If e and f take on their minimal values the space is known as the mSBH space and was introduced in two dimensions by Adjerid *et al* [2]. It corresponds to the space of minimal size that is both symmetric and possesses the asymptotic equivalence property with respect to the TP Lobatto interpolant (the SBH basis of Szabó and Babuška [24] does not possess this property and therefore is not studied). As a comparison equivalent basis functions for the SBH ($S_\Delta^{3,0,3}$) and mSBH ($S_\Delta^{3,0,4}$)

spaces are displayed in Figure 2. The latter has six (one per face) more basis functions than the former. Due to symmetry with respect to the faces the mSBH basis has $3(p-1)$ more basis functions per element than the minimal basis that satisfies the asymptotic equivalence property (in two dimensions they are the same). Since the total number of basis functions per element is $O(p^3)$ for $p > 4$ this is not significant. With $e = 3p$ and $f = 2p$ the TP basis is obtained. Its index set has the compact form

$$\mathcal{S}^{p,TP} = \mathcal{S}^{p,3p,2p} = \{(i,j,k) \mid 0 \leq i,j,k \leq p\}, \tag{17}$$

and the corresponding space is referred to as $S_{\Delta}^{p,TP}$. This is the basis of maximal size that is considered. For $p = 3$ the TP basis has 22 more basis functions than the mSBH basis (cf. Figure 2). For large p the number of basis functions is dominated by those associated with the element interior. As a result for larger p the TP basis has nearly six times as many basis functions as the mSBH basis. Any basis that satisfies (16) with $e < 3p$ or $f < 2p$ is referred to as a sub-tensor-product (sub-TP) basis. An example of such a basis for $p = 3$, $S_{\Delta}^{3,7,5}$, is shown in Figure 2. It has 12 more basis functions than the mSBH basis and is considered in Section 5.

The finite element space $S^{\Delta_\Omega,p,e,f}$ is the space of piecewise continuous polynomials of degree p whose restriction to element Δ_i is $S_{\Delta_i}^{p,e,f}$. Thus, if $U \in S^{\Delta_\Omega,p,e,f}$

$$U|_{\Delta_i}(\mathbf{x}) \equiv \sum_{(l,m,n)\in\mathcal{S}^{p,e,f}} U_{l,m,n}^i \Phi_l(x;h_i^x;m_i^x)\Phi_m(y;h_i^y;m_i^y)\Phi_n(z;h_i^z;m_i^z), \tag{18}$$

where the $U_{l,m,n}^i$ are the coordinates associated with element Δ_i. Since Δ_Ω satisfies the one-irregular rule the support of all basis functions associated with regular components is uniformly bounded [9, 27]. The $U_{l,m,n}^i$ associated with irregular components are constrained by continuity across element boundaries. This approach is taken in contrast to the discontinuous finite element method [11]. Allowing discontinuity simplifies the refinement algorithms and thus, the data structures, but requires more degrees of freedom. The $U_{l,m,n}^i$ associated with regular grid components constitute the degrees of freedom for the problem. The total number of degrees of freedom is denoted by N_{dof} while the total number of coordinates is denoted by N_{crd}.

Adaptivity necessitates interpolation of the computed solution from one grid to another. Let Δ_Ω and Δ_Ω^* be the current and previous grids, respectively. Interpolation of the finite element solution U^* on Δ_Ω^* onto Δ_Ω to serve as an initial guess for Newton's method (see below) is done using the TP Lobatto interpolant. The Lobatto interpolant $U_I \in S^{\Delta_\Omega,p,3p,2p}$ on Δ_i has the form

$$U_I|_{\Delta_i}(\mathbf{x}) \equiv \sum_{(l,m,n)\in\mathcal{S}^{p,TP}} \bar{U}_{l,m,n}^i \Phi_l(x;h_i^x;m_i^x)\Phi_m(y;h_i^y;m_i^y)\Phi_n(z;h_i^z;m_i^z). \tag{19}$$

The $\bar{U}_{l,m,n}^i$ are specified by requiring that

$$U_I(\mathbf{x}_j^i) = U^*(\mathbf{x}_j^i),\ j = 1,\ldots,(p+1)^3, \tag{20}$$

where the $\mathbf{x}_j^i$ are the Lobatto points scaled to Δ_i. A Lobatto point of Δ_i, $\mathbf{x}_j^i = (\xi_j^i, \eta_j^i, \nu_j^i)$, satisfies

$$\Phi_{p+1}(\xi_j^i;h_i^x;m_i^x) = \Phi_{p+1}(\eta_j^i;h_i^y;m_i^y) = \Phi_{p+1}(\nu_j^i;h_i^z;m_i^z) = 0. \tag{21}$$

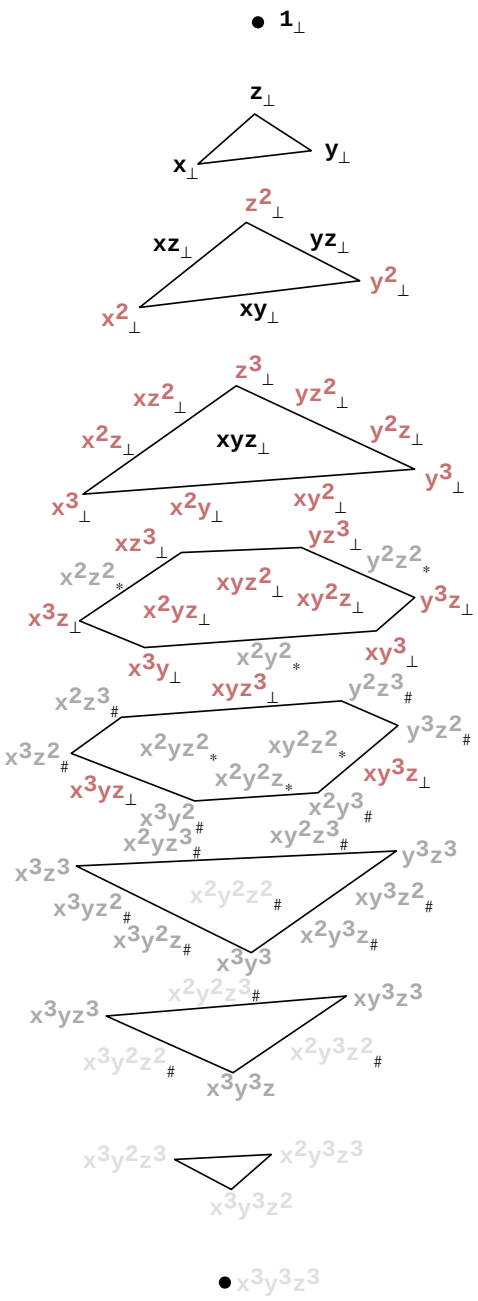

FIGURE 2. Basis functions of the form (13) for the spaces $S_\Delta^{3,0,3}$ (indicated by a $\perp$), $S_\Delta^{3,0,4}$ (indicated by $\perp$ and *), $S_\Delta^{3,7,5}$ (indicated by $\perp$, * and #) and $S_\Delta^{3,9,6}$ (all functions). The basis functions associated with nodes, edges, faces and element interiors are shown in grey scale from darkest to lightest, respectively.

The initial guess at the solution on Δ_Ω is thus given by

(22)
$$U|_{\Delta_i}(\mathbf{x}) = U_I^{e,f}|_{\Delta_i}(\mathbf{x}) \equiv \sum_{(l,m,n)\in\mathcal{S}^{p,e,f}} \bar{U}^i_{l,m,n}\Phi_l(x;h_i^x;m_i^x)\Phi_m(y;h_i^y;m_i^y)\Phi_n(z;h_i^z;m_i^z),$$

where the $\bar{U}^i_{l,m,n}$ are the TP coordinates on Δ_i and $U_I^{e,f}$ is the projection of U_I onto $S^{\Delta_\Omega,p,e,f}$. A slight modification of this strategy that improves the approximation error is used for faces and interiors on elements with irregular components. For example, on a regular face with irregular edges (at most two) the solution U^* along those edges is used.

The Galerkin form of (1) in the case of Dirichlet boundary conditions consists of determining $u \in H^1_E(\Omega)$ such that

$$a(u,v) = (F,v),\ \forall\ v \in H^1_0(\Omega), \tag{23}$$

where

$$a(u,v) = \int_\Omega \nabla u\cdot\nabla v dxdy,\ (F,v) = \int_\Omega Fvdxdy, \tag{24}$$

and the subscripts E and 0 further restrict functions to satisfy the Dirichlet and homogeneous Dirichlet boundary conditions, respectively. The finite element approximation $U \in S_E^{\Delta_\Omega,p,e,f}$ is determined as the solution of

$$a(U,V) = (F,V),\ \forall\ V \in S_0^{\Delta_\Omega,p,e,f}. \tag{25}$$

On a Dirichlet boundary face χ I require that

$$U|_\chi = U_I^{e,f}|_\chi. \tag{26}$$

The coordinates of $U_I^{e,f}$ are found using a generalization to three dimensions of the one-dimensional scheme given in Moore [18].

Explicit utilization of (18) in (25) yields the nonlinear system

$$\mathbf{g}(\hat{\mathbf{U}}) = 0, \tag{27}$$

where $\hat{\mathbf{U}}$ is the vector of the degrees of freedom (coordinates associated with regular components). This system is solved using Newton's method with Jacobian matrix $\mathbf{J} = \partial\mathbf{g}/\partial\hat{\mathbf{U}}$. For each Newton step the Jacobian is fixed and the linear system is solved by GMRES [23] with the ILUT preconditioner of Saad [22].

4. Error estimation and adaptivity

For all spaces $S^{\Delta_\Omega,p,e,f}$ such that e and f satisfy (16) the finite element solution U of (1) on a uniform grid satisfies (under appropriate conditions on F, cf. [29])

$$\|u - U\|_1 = O(H^p), \tag{28}$$

where $H = \max\{h_i^x, h_i^y, h_i^z\}$ for any i. Thus, to leading order, the errors are the same. The difference in accuracy is therefore due to the presence of higher-order terms. The TP solution is the most accurate while the mSBH solution is the least accurate. The same is true for the TP interpolant U_I and its projections $U_I^{e,f}$. In this case the difference is described by the following lemma.

Lemma 2. *Let U_I be the TP interpolant and $U_I^{e,f}$ its projection onto $S^{\Delta_\Omega,p,e,f}$. If Δ_Ω is a uniform grid then*

$$\|U_I - U_I^{e,f}\|_1 = O(H^{\min(e,f)}), \tag{29}$$

where the constant depends on derivatives of u of order $\min(e,f)+1$.

Proof. Appendix A. □

Lemma 2 implies that

$$\|u - U_I^{e,f}\|_1 = \|u - U_I\|_1 + O(H^{\min(e,f)}). \tag{30}$$

Thus, the order of the additional error that results from using the smaller basis is equal to the order of the lowest order term missing from the TP basis. The additional error will be most apparent on coarse grids and for the smaller bases. A similar result is likely to hold for the finite element solution (cf. section 5).

The *a posteriori* error estimation strategy described in this section is based on the formula for the error in the Lobatto interpolant on Δ_i. If Δ_i has no irregular nodes the formula is given by

$$\begin{aligned} \|u - U_I\|^2_{1,\Delta_i} &= \left(\frac{(p-1)!}{(2p-1)!}\right)^2 \frac{h_i^x h_i^y h_i^z}{4(2p+1)}(((h_i^x)^p u^{(p+1,0,0)}(\mathbf{m}^i))^2 \\ &+ ((h_i^y)^p u^{(0,p+1,0)}(\mathbf{m}^i))^2 + ((h_i^z)^p u^{(0,0,p+1)}(\mathbf{m}^i))^2) + O(H^{2p+5}), \end{aligned} \tag{31}$$

where $\mathbf{m}^i = (m_i^x, m_i^y, m_i^z)$ [17]. On elements with irregular nodes the formulas are more complicated (cf. the Appendix B for formulas for the cases $p = 2, \ldots, 5$). However, they involve the same derivatives $u^{(p+1,0,0)}(\mathbf{m}^i)$, $u^{(0,p+1,0)}(\mathbf{m}^i)$ and $u^{(0,0,p+1)}(\mathbf{m}^i)$. The estimator works by computing estimates of these three derivatives.

To that end consider the augmented finite element solution U^{p+1} defined on each element Δ_i by

$$\begin{aligned} U^{p+1}(\mathbf{x})|_{\Delta_i} &= U(\mathbf{x})|_{\Delta_i} + W_i^x \psi_{p+1}(x; h_i^x; m_i^x) + W_i^y \psi_{p+1}(y; h_i^y; m_i^y) \\ &+ W_i^z \psi_{p+1}(z; h_i^z; m_i^z), \end{aligned} \tag{32}$$

where

$$\psi_{p+1}(\xi; h; m) = \frac{h^{p+1}(p-1)!(p+1)!}{4(2p-1)!}\sqrt{\frac{2}{2p+1}}\Phi_p(\xi; h; m). \tag{33}$$

The coefficients W_i^x, W_i^y and W_i^z are determined by requiring that

$$a(U^{p+1}, V^{p+1,k}) = (f(U), V^{p+1,k}),\ k = 1, 2, 3. \tag{34}$$

The function $V^{p+1,1}(\mathbf{x})$ is given by

$$V^{p+1,1}(\mathbf{x}) = \psi_{p+1}(x; h_i^x, m_i^x) g(y; h_i^y; m_i^y) g(z; h_i^z; m_i^z), \tag{35}$$

where

$$g(\xi; h; m) = \begin{cases} \frac{\psi_p(\xi;h;m)}{\xi - m}, & p \text{ odd} \\ \frac{\psi_{p+1}(\xi;h;m)}{\xi - m}, & p \text{ even} \end{cases} \tag{36}$$

with comparable definitions for $V^{p+1,2}$ and $V^{p+1,3}$.

Convergence of the error estimate to the true error on uniform patches can be shown for a certain class of functions F. Specifically, following [29], let

$$F(x, u) = \kappa(u) + \hat{F}(x), \qquad |\kappa(u) - \kappa(v)| < L|u - v|, \tag{37}$$

$$a(u - v, u - v) + (\kappa(u) - \kappa(v), u - v) \geq \alpha\|u - v\|_1^2 \tag{38}$$

where $L, \alpha > 0$. Then the following Theorem holds.

Theorem 1. *Consider the uniform patches Ω' and Ω'' with mesh spacing h^x, h^y and h^z. If $u \in H_0^{p+5}(\Omega'')$ is the solution of (1) and $U \in S_0^{\Delta_\Omega, p, e, f}$ is the solution of*

(25) *with e and f satisfying* (16), *then* $\exists\ C$ *depending only on* Ω', Ω'' *and* Ω *such that*

$$\begin{aligned}\|e\|^2_{1,\Omega'} &= \left(\frac{(p-1)!(p+1)!}{(2p-1)!}\right)^2 \frac{h^x h^y h^z}{4(2p+1)} \sum_{i\in\Upsilon'} (((h^x)^p W_i^x)^2 + ((h^y)^p W_i^y)^2) \\ &+ ((h^z)^p W_i^z)^2) + C(H^{2p+1}\|u\|^2_{p+2,\Omega''} + H^{-1}\|e\|^2_{0,\Omega''}), \end{aligned} \tag{39}$$

where $e = u - U$.

Proof. The proof follows from the proof of Theorem 3.2 in [28] and the proof of Theorem 4.4 in [19] generalized from two to three dimensions. □

Remark. The term $H^{-1}\|e\|^2_{0,\Omega''}$ represents a pollution error [5] and in practice, if the adaptive strategy is effective, is often higher-order than the error on Ω'. When Δ_Ω is a uniform grid (39) holds without this term.

Remark. Theorem 1 suggests that refinement strategies producing grids with large uniform patches may yield more reliable results (cf. section 5).

The error estimator of Theorem 1 is asymptotically exact for all spaces satisfying (16). However, in an adaptive setting, the estimator will be of little practical value if the estimates on coarse grids are poor, especially if they significantly underestimate the error. For finite element solutions in sub-TP spaces Lemma 2 and Theorem 1 suggest the effectiveness of the estimator is governed by two factors: by how well it estimates the error in the TP solution; and by how well the sub-TP space approximates the TP space. Previous work in two dimensions has demonstrated that the estimator is effective when the solution is in the TP space [2, 19]. Thus, the latter source of error may be more critical in determining estimator performance than the former. The degradation in estimator performance due to the use of sub-TP bases is explored in section 5. To help distinguish between the two sources of estimator error the error estimates obtained when using sub-TP bases the are compared with the error in the TP solution.

The error estimator of Theorem 1 is valid only on uniform patches. Typical adaptive grids consist of such patches connected by elements with irregular components. On such grids the asymptotic equivalence property no longer holds [16]. However, formulas for the interpolation error on such elements are computable and presented in Appendix B. As noted earlier they also involve $u^{(p+1,0,0)}(\mathbf{m}^i)$, $u^{(0,p+1,0)}(\mathbf{m}^i)$ and $u^{(0,0,p+1)}(\mathbf{m}^i)$. Computational experience on uniform grids (cf. section 5) indicates that these derivatives are approximated by $(p+1)!W_i^x$, $(p+1)!W_i^y$ and $(p+1)!W_i^z$, respectively. These derivative approximations, together with (31) and the formulas in the Appendix B, are used to generate error indicators, $\|E\|_{1,\Delta_i}$, on all elements of the grid (cf. section 5 for the effectiveness of this approach). This is referred to as the adaptive irregular-grid-indicator strategy while using (31) on all elements is called the adaptive uniform-grid-indicator strategy.

The h-refinement strategy is based on an earlier one-dimensional approach [12]. Error estimates are measured in the root-mean-square-H^1 norm [12]

$$\|E\|_{\Delta_i,rms} = \frac{\|E\|_{1,\Delta_i}}{atol + rtol\|U\|_{1,\Delta_i}},\ \|E\|^2_{\Delta_\Omega,rms} = \sum_{i=1}^{N_{el}} \|E\|^2_{\Delta_i,rms}, \tag{40}$$

where *atol* and *rtol* are the absolute and relative error tolerances, respectively. If on Δ_Ω, $\|E\|_{\Delta_\Omega,rms} > 1$ elements with error indicators larger than $\frac{rf}{\sqrt{N_{el}}}$ are refined as described in section 2 where rf is a refinement safety factor. Eight sibling elements are coarsened into one element if the error indicators on all eight are smaller than

$\frac{cf}{\max(1,2^{p-3})\sqrt{N_{el}}}$ [12] where cf is the coarsening safety factor. If the sibling rule is used the error indicators on the cousins must also satisfy this requirement. A solution is computed on the new grid and the process is repeated until the tolerance is met or too many levels of refinement are needed.

This h-refinement code is referred to by *href* and is called with six parameters *p, e, f, atol, rtol* and *m*, where *atol* and *rtol* are user-prescribed absolute and relative error tolerances and *m* is the mode. Three modes are considered, denoted by *IS*, *UN* and *IN*. The first component (*I* or *U*) indicates whether the irregular- or regular-grid-indicator strategy, respectively, is used. The second component shows whether the sibling rule is used (*S*) or not (*N*). Thus, for example, mode *IN* means that irregular-grid-indicator strategy for error estimation is in use while the sibling rule is not.

The bulk of the storage needed by the code is used to store the Jacobian (for the matrix-vector products in GMRES) and the preconditioner (it turns out that N_{dof} is a reasonable proxy for the total storage). One array whose size is limited by available memory is used to store both. Thus, as the number of unknowns grows, the Jacobian also increases in size leaving less room for the ILUT-generated preconditioner. The size and effectiveness of this preconditioner is controlled by two parameters, a drop tolerance *itol* and a maximum row fill-in *ifil*. If insufficient space is available due to a large Jacobian the value of *ifil* is adaptively decreased. This in turn may degrade the performance of GMRES (cf. Example 5.3).

5. Computational results

In the first example the performance of the estimators as a function of the basis is considered on a family of uniform grids. The second and third examples examine the interplay of the estimator performance, the convergence of the iterative solver GMRES and h-adaptivity. The first and second examples are linear, the third is nonlinear. In order to approximately equalize the effects of terms in the error expansion the exact solutions in the first and second examples are chosen so that all derivatives of the same order are equal. In all three examples $itol = 10^{-6}$ while *ifil* is taken to be 1000 in Example 5.1 and 800 (except for $p = 5$ and the smaller tolerance in Example 5.2 where $itol = 1200$) in the other two. The tolerance for GMRES is chosen to be 10^{-12}. The maximum size of the Krylov subspace is set at 10. This is quite small in comparison with the size of the systems on the finest grid in the second two examples and was sufficient in most cases. The h-adaptivity safety factors *rf* and *cf* have values 0.8 and 0.1, respectively. In all cases $rtol = 0$. All calculations are performed in double precision on a Compaq ES40 Alphaserver. The codes are written in a combination of Fortran 77 and Fortran 90. Condition number estimates in Example 5.1 are computed from singular value estimates obtained using ARPACK [14].

Example 5.1 Consider Poisson's equation

$$-\Delta u = F(\mathbf{x}),\ \mathbf{x} \in \Omega \equiv (0,1)^3, \tag{41}$$

where F and the Dirichlet boundary conditions are chosen so that

$$u(\mathbf{x}) = \tanh(5(x + y + z - 1.7)). \tag{42}$$

To study the effect of the finite element space on the estimator performance I solve (41) on a family of uniform grids with $h_i^x = h_i^y = h_i^z = 1/2^j$, $i = 1, \ldots, N_{el}$, $N_{el} = N^3$, $N = 2^j$, $j = 1, \ldots, 4$, and $p = 2, \ldots, 5$. For each discretization the number of degrees of freedom and the effectivity index, $\theta = \frac{\|E\|_1}{\|e\|_1}$ are displayed in Tables 1-4. Since the estimator is asymptotically exact the effectivity index

should approach 1 as the grid is refined. Additionally, the TP-effectivity index $\theta_{TP} = \frac{\|E\|_1}{\|e_{TP}\|_1}$ is also shown, where e_{TP} is the error in the TP finite element solution. In order to examine the accuracy of the derivative approximations W_i^x, W_i^y and W_i^z, $i = 1, \ldots, N_{el}$, the derivative accuracy index,

$$\begin{aligned} \hat{\theta} &= (p+1)! \left(\sum_{i=1}^{N_{el}} \left(W_i^x - \frac{u^{(p+1,0,0)}(\mathbf{m}^i)}{(p+1)!} \right)^2 + \left(W_i^y - \frac{u^{(0,p+1,0)}(\mathbf{m}^i)}{(p+1)!} \right)^2 \right. \\ &+ \left. \left(W_i^z - \frac{u^{(0,0,p+1)}(\mathbf{m}^i)}{(p+1)!} \right)^2 \right)^{\frac{1}{2}} \\ &/ \left(\sum_{i=1}^{N_{el}} ((u^{(p+1,0,0)}(\mathbf{m}^i))^2 + (u^{(0,p+1,0)}(\mathbf{m}^i))^2 + (u^{(0,0,p+1)}(\mathbf{m}^i))^2) \right)^{\frac{1}{2}}, \end{aligned} \tag{43}$$

is computed. For $p > 2$ the number of possible combinations of e and f is quite large so only a representative sample is presented. Among the representatives are the TP solution (the largest possible values of e and f) and the mSBH solution (the smallest values of e and f). I also show results for two other extreme cases, the smallest (largest) possible value of e coupled with the largest (smallest) possible value of f.

For both p even and odd the estimates appear to converge in p for N sufficiently large (this was also observed in two dimensions for p even by Adjerid *et al* [1]). As suggested in section 4 estimator performance improves as e and f approach their maximum (TP) values. This is reflected in the derivative accuracy index. As p increases the effectiveness of the estimators (and derivative estimates) associated with bases for which e and/or f is at its minimum value decreases dramatically. For example, with $p = 5$ and the mSBH basis, the effectivity index ranges between 0.02 and 0.08. More importantly from the standpoint of adaptivity the estimates significantly underestimate the error, even on the finest grid. Notably in these cases the estimators do provide reasonable approximations of the TP error. This is evidence for the hypothesis of section 4 that the primary source of estimator ineffectiveness when using sub-TP spaces is how poorly the sub-TP space approximates the TP space. As p increases larger values of e and f are needed to avoid this.

On the other hand the savings in terms of the number of degrees of freedom from using the mSBH basis is sizable. The difference in N_{dof} becomes more pronounced as p and N increase going from 48% fewer unknowns with $p = 2$ to 73% fewer with $p = 5$ on the finest grid (the latter value is close to the theoretical limit of 83% savings, cf. section 3). The savings achieved by using the sub-TP basis with maximum (minimum) f and minimum (maximum) e is 49% (25%) with $p = 5$ on the finest grid. Additionally condition numbers in Table 5 for Jacobian matrices on the grid with $N = 2$ for a selection of p, e and f show that they become more ill-conditioned as p, e and f increase. When $p = 4$ and $p = 5$ the range in condition numbers between the minimal and maximal values of e and f is dramatic.

While the mSBH basis (except for $p = 2$) and sub-TP bases with the extreme values of f and e do not provide a robust alternatives to the TP basis promising intermediate choices are available. These bases offer reliable error estimation with significantly fewer degrees of freedom and improved Jacobian conditioning than the TP basis. When $p = 3$ choosing $e = 7$ and $f = 5$ (cf. Figure 2) leads to 25% savings on the finest grid with only slight deterioration in error estimation on all grids. For $p = 4$ selecting the basis with $e = 8$ and $f = 7$ leads to 30% savings on the finest

N	e	f	N_{dof}	θ	θ_{TP}	$\hat{\theta}$
2	6	4	125	0.5749	0.5749	0.7070
2	6	0	89	0.7130	1.3330	0.4788
2	0	4	117	0.6020	0.7636	0.6397
2	0	0	81	0.6939	1.3306	0.4816
4	6	4	729	0.6621	0.6621	0.3943
4	6	0	489	0.7746	1.1542	0.3711
4	0	4	665	0.4631	0.6258	0.3596
4	0	0	425	0.6016	1.1192	0.3559
8	6	4	4,913	0.9116	0.9116	0.1472
8	6	0	3,185	0.9435	1.0948	0.1336
8	0	4	4,401	0.8823	0.8899	0.1800
8	0	0	2,673	0.6882	0.9867	0.1960
16	6	4	35,937	0.9753	0.9753	0.03028
16	6	0	22,881	0.9875	1.0003	0.03680
16	0	4	31,841	0.9732	0.9733	0.03449
16	0	0	18,785	0.8636	0.9354	0.1121

TABLE 1. The number of degrees of freedom, effectivity index (θ), TP-effectivity index (θ_{TP}) and derivative accuracy index ($\hat{\theta}$) as a function of N, e and f for $p = 2$ for Example 5.1.

grid with little loss in estimator performance. The choices with $p = 5$, $e = 11$, $f = 9$ and $e = 9$, $f = 8$ result in 18% and 43% savings with comparable estimators. Reductions in condition number range from a factor of 4 when $p = 3$ to 18 when $p = 5$ and $e = 9$, $f = 8$. These choices are examined in the context of adaptivity in the next two examples.

Example 5.2 Consider the linear reaction-diffusion equation

$$-\Delta u = F(\mathbf{x}) - u, \ \mathbf{x} \in \Omega \equiv (0,1)^3, \tag{44}$$

where F and the Dirichlet boundary conditions are chosen so that

$$u(\mathbf{x}) = \tanh(5(x + y + z - 1.0)). \tag{45}$$

I solve (44) using the *h*-adaptive algorithm described in section 4. The initial grid is uniform with $N = 4$. For each order p, solutions for a subset of the values of e, f (cf. Example 5.1) and m are obtained for two (three when $p = 5$) absolute error tolerances. Storage savings are characterized by the *space fraction* χ_s which is the ratio of N_{dof} on the final grid when using *href(p,e,f,atol,0,m)* to the corresponding value of N_{dof} on the final grid obtained using *href(p,3p,2p,atol,0,IN)*. The *time fraction* χ_t is defined similarly with total CPU time replacing N_{dof}. Grid irregularity is measured by $\%irr = \frac{N_{crd} - N_{dof}}{N_{crd}}$ where again N_{crd} and N_{dof} are calculated on the final grid. For each set of parameters *p, e, f, m* and *atol* the number of degrees of freedom, the error in the H^1-seminorm, grid irregularity, space and time fractions and effectivity index on the final grid are displayed in Tables 6-9. The number of levels at which a solution is computed (the number of refinement levels is one less), total number of GMRES iterations, and CPU time used in computing the solution are also shown. For $p = 5$, the four levels of refinement required to satisfy the smallest tolerance, leads to larger linear systems than could be solved with the

N	e	f	N_{dof}	θ	θ_{TP}	$\hat{\theta}$
2	9	6	343	0.1289	0.1289	0.9427
2	9	4	235	0.07189	0.1164	0.9473
2	7	5	275	0.09555	0.1313	0.9414
2	0	6	279	0.1625	0.4714	0.8376
2	0	4	171	0.1778	0.4670	0.8341
4	9	6	2,197	0.6441	0.6441	0.5769
4	9	4	1,477	0.3424	0.6221	0.5992
4	7	5	1,701	0.5379	0.6497	0.5710
4	0	6	1,685	0.3732	1.5165	0.4098
4	0	4	965	0.3201	1.5414	0.4074
8	9	6	15,625	0.8178	0.8178	0.1510
8	9	4	10,441	0.6569	0.7953	0.1741
8	7	5	11,849	0.7987	0.8236	0.1456
8	0	6	11,529	0.6430	1.0460	0.1630
8	0	4	6,345	0.4457	1.1246	0.2174
16	9	6	117,649	0.9568	0.9568	0.05686
16	9	4	78,481	0.8849	0.9466	0.06956
16	7	5	88,209	0.9559	0.9594	0.05347
16	0	6	84,881	0.9758	1.0034	0.01988
16	0	4	45,713	0.7543	1.0796	0.08303

TABLE 2. The number of degrees of freedom, effectivity index (θ), TP-effectivity index (θ_{TP}) and derivative accuracy index ($\hat{\theta}$) as a function of N, e and f for $p = 3$ for Example 5.1.

available resources for most of the cases considered. Thus, in all but three cases, only N_{dof}, χ_s and $\%irr$ are presented at the final (fourth) level. For this reason I also solve (44) *href(5,e,f,0.001,0,m)* for the appropriate choices of e, f and m. Since the results are markedly different than in the other cases they are discussed at the end.

In general as e and f increase N_{dof} increases on the final grid. Substantial savings are achieved by using sub-TP bases. For $p = 2$ the savings range between 40% and 60%. When $p = 3$ a reduction of 30% is achieved when using $e = 7$ and $f = 5$. With the mSBH basis the reduction is 60%. The comparable savings when $p = 4$ are 30% and 70%, respectively. For $p = 5$ the savings at the smaller two tolerances are about 20% with $e = 11$ and $f = 9$, 40% with $e = 9$ and $f = 8$ and about 70% with the mSBH basis. These savings are especially important when $p = 5$ since at the smallest tolerance only the basis with $e = 9$ and $f = 8$ yields a reliable solution. Storage is not significantly effected by the use of the uniform-grid-indicator strategy. All runs with the sibling rule use substantially more storage than their non-sibling counterparts, typically by about 30% to 40%.

In all cases the tolerance is met when using bases larger than the mSBH basis. In fact in most cases the error is substantially less than the tolerance. This is due to the combination of binary refinement with high order elements since a reduction in the size of elements by a factor of two yields a significantly greater reduction in the error. For $p = 2$ and 3 the error in the solution obtained using the mSBH

N	e	f	N_{dof}	θ	θ_{TP}	$\hat{\theta}$
2	12	8	729	0.7723	0.7723	0.6110
2	12	5	513	0.3458	0.6838	0.7033
2	8	7	557	0.6536	1.6075	0.2310
2	0	8	513	0.3098	2.2223	0.1398
2	0	5	297	0.2910	2.0332	0.06523
4	12	8	4,913	0.6926	0.6926	0.2147
4	12	5	3,473	0.09183	0.2072	0.8434
4	8	7	3,585	0.6717	1.1316	0.5957
4	0	8	3,185	0.1176	2.0827	1.9130
4	0	5	1,745	0.1133	2.0081	1.8075
8	12	8	35,937	1.1383	1.1383	0.06164
8	12	5	25,569	0.2769	0.6104	0.6249
8	8	7	25,505	1.2264	1.3630	0.1553
8	0	8	22,113	0.2080	2.3042	0.9660
8	0	5	11,745	0.2047	2.4080	1.0344
16	12	8	274,625	0.9990	0.9990	0.01033
16	12	5	196,289	0.7254	0.9033	0.1367
16	8	7	191,937	1.0043	1.0049	0.06604
16	0	8	164,033	0.2700	1.1345	0.1536
16	0	5	85,697	0.2626	1.2332	0.2532

TABLE 3. The number of degrees of freedom, effectivity index (θ), TP-effectivity index (θ_{TP}) and derivative accuracy index ($\hat{\theta}$) as a function of N, e and f for $p = 4$ for Example 5.1.

basis satisfies the tolerance on the final grid. When $p = 4$ and 5 (with the two smaller tolerances) the error obtained using this basis is significantly larger than the error achieved by the other bases. Moreover, when $p = 5$, it does not meet the tolerance requirement in three out of four cases. The reason is clearly the poor performance of the error indicators as is already observed on uniform grids in the previous example.

With the exception of the mSBH basis when $p = 4$ and 5 the reliability of the error estimates on the final grid when using the irregular-grid-indicator strategy is quite high. In most situations the error is slightly overestimated which is more desirable. As expected (cf. section 4) for $p = 4$ and 5 using the mSBH basis no longer produces reliable error estimates. However, when $p = 4$, the tolerance is met in three of four cases when using this basis, due as noted to above, to the combination of high-order and binary refinement. Clearly, the mSBH basis is not an appropriate choice for high order. The uniform-grid-indicator-strategy leads to smaller estimates of the error. They are adequate for $p = 2$ and 3 but not for $p = 4$ and 5 (though the tolerance is always met). This justifies the use of the more complicated irregular-grid-indicator approach. As expected, the sibling rule improves grid regularity but its benefits on error estimation are not substantial. Error estimate reliability also improves with increasing refinement as shown in Table 10 for the three bases used when $p = 3$ and $atol = 0.001$ (other values of p, e and f exhibited similar behavior). On the coarser grids the error estimates

N	e	f	N_{dof}	θ	θ_{TP}	$\hat{\theta}$
2	15	10	1,331	0.3975	0.3975	0.8875
2	15	6	971	0.1395	0.3403	0.9049
2	11	9	1,135	0.3360	0.6075	0.8234
2	9	8	871	0.2592	0.9599	0.7235
2	6	10	827	0.08508	1.2357	0.6592
2	6	6	467	0.07903	1.1052	0.6995
4	15	10	9,261	1.0125	1.0125	0.3624
4	15	6	6,861	0.3265	0.9894	0.3875
4	11	9	7,741	0.8667	1.1032	0.2978
4	9	8	5,725	0.4393	1.8975	0.2534
4	6	10	5,229	0.07959	3.0070	0.9962
4	6	6	2,829	0.07655	2.8969	0.9255
8	15	10	68,921	0.6360	0.6360	0.06901
8	15	6	51,641	0.4056	0.6767	0.1100
8	11	9	56,953	0.6317	0.6326	0.05767
8	9	8	41,209	0.5397	0.6395	0.2161
8	6	10	36,665	0.01781	1.0848	0.8328
8	6	6	19,385	0.01928	1.1785	0.9147
16	15	10	531,441	0.9982	0.9982	0.01485
16	15	6	400,881	0.6680	1.0693	0.06017
16	11	9	436,465	0.9983	0.9983	0.01460
16	9	8	312,049	1.0079	1.0124	0.01427
16	6	10	273,393	0.06959	1.2531	0.2805
16	6	6	142,833	0.07583	1.3974	0.4120

TABLE 4. The number of degrees of freedom, effectivity index (θ), TP-effectivity index (θ_{TP}) and derivative accuracy index ($\hat{\theta}$) as a function of N, e and f for $p = 5$ for Example 5.1.

	$p=2$			$p=3$		
e, f	6,4	6,0	0,0	9,6	7,5	0,4
$\kappa_2(\mathbf{J})$	$1.7e2$	$2.6e1$	4.9	$2.0e3$	$4.6e2$	$3.1e1$

	$p=4$			$p=5$			
e, f	12,8	8,7	0,5	15,10	11,9	9,8	6,6
$\kappa_2(\mathbf{J})$	$1.2e4$	$1.2e3$	$5.6e2$	$5.0e4$	$1.1e4$	$2.7e3$	$1.9e2$

TABLE 5. The condition number $\kappa_2(\mathbf{J})$ of the Jacobian matrices on the uniform grid with $N = 2$ for select values of p, e and f for Example 5.1.

underestimate the error, especially when using the mSBH basis. This shortcoming is partly compensated for by the significant reductions in the error for larger p as the grid is refined.

atol	e	f	m	N_{dof}	χ_s	$\|e\|_1$	θ	lvl	% irr	nGM	time	χ_t
5.0e-2	6	4	*IN*	17,445	1.00	0.18e-1	1.01	3	12.8	10	2.9e2	1.00
5.0e-2	6	4	*UN*	16,863	0.97	0.19e-1	0.94	3	13.2	10	2.6e2	0.90
5.0e-2	6	0	*IN*	11,250	0.64	0.19e-1	1.01	3	14.5	9	9.1e1	0.32
5.0e-2	0	0	*IN*	7,401	0.42	0.24e-1	0.98	3	22.8	8	3.6e1	0.12
5.0e-2	6	4	*IS*	22,513	1.29	0.18e-1	0.99	3	7.0	10	4.5e2	1.55
5.0e-2	6	0	*IS*	14,257	0.82	0.18e-1	0.99	3	8.1	9	1.5e2	0.52
5.0e-2	0	0	*IS*	9,821	0.56	0.21e-1	0.88	3	13.6	8	6.9e1	0.24
1.0e-2	6	4	*IN*	118,219	1.00	0.47e-2	1.06	4	9.4	16	5.9e3	1.00
1.0e-2	6	4	*UN*	117,457	0.99	0.47e-2	0.96	4	9.3	16	6.0e3	1.02
1.0e-2	6	0	*IN*	74,470	0.63	0.47e-2	1.03	4	10.8	15	2.5e3	0.42
1.0e-2	0	0	*IN*	61,904	0.52	0.47e-2	0.98	4	12.8	15	1.9e3	0.32
1.0e-2	6	4	*IS*	141,305	1.20	0.44e-2	1.00	4	6.8	16	8.4e3	1.42
1.0e-2	6	0	*IS*	88,673	0.75	0.44e-2	1.00	4	7.9	14	3.5e3	0.59
1.0e-2	0	0	*IS*	71,361	0.60	0.45e-2	0.97	4	9.7	15	2.6e3	0.44

TABLE 6. The number of degrees of freedom, space fraction, error (in the H^1 seminorm), effectivity index and irregularity on the final grid and the number of refinement levels (lvl), total number of GMRES iterations (nGM), total time and time fraction used by *href(2,e,f,atol,0,m)* for Example 5.2.

The savings in CPU time are also significant. For the mSBH basis they are typically more than 80%. For other sub-TP bases they range from 12% to 48% though there are no general trends as p increases. Since using the sibling rule leads to more unknowns the CPU times for these runs are also generally larger, sometimes substantially so.

Not surprisingly the number of GMRES iterations is directly related to N_{dof}. Thus, fewer iterations are needed as e and f decrease (recall also the results on condition numbers from Example 5.1). The differences are not sizable. Since most of the time spent by the algorithm involves the assembling and (incomplete) factoring of $\mathbf{J}$ [20] the impact in savings in CPU time from fewer GMRES iterations is minimal. In all cases less than 10 GMRES iterations were required for convergence. Thus, the size of the Krylov space was not a factor in performance due to the effectiveness of the preconditioning.

Several factors contribute to the wide variability in the performance of *href* when $p = 5$ at the coarsest tolerance $atol = 0.001$. In this case the sibling rule which leads to slightly more refinement at each level, produces an acceptable (in terms of the error estimate) solution with one fewer level of refinement ($lvl = 2$). Similarly, since the uniform-grid-indicator strategy tends to underestimate the error it also leads to termination after one refinement level. Somewhat surprisingly three refinement levels ($lvl = 4$) are required when $e = f = 6$ and $m = IN$. This is due to the fact that on coarser grids the error when using the mSBH basis is substantially larger than the error with the other bases. Here is an additional reason for rejecting this basis for higher p. In many cases the error estimate significantly overestimates the error on the final level. Clearly, for this tolerance, $p = 5$ is not an optimal choice.

atol	e	f	m	N_{dof}	χ_s	$\lvert e\rvert_1$	θ	lvl	% irr	nGM	time	χ_t
5.0e-3	9	6	*IN*	45,244	1.00	0.16e-2	1.20	3	12.6	12	2.4e3	1.00
5.0e-3	9	6	*UN*	44,092	0.97	0.16e-2	0.92	3	12.2	12	2.3e3	0.96
5.0e-3	7	5	*IN*	33,607	0.74	0.16e-2	1.18	3	14.8	12	1.4e3	0.58
5.0e-3	0	4	*IN*	17,548	0.39	0.23e-2	0.85	3	20.5	10	3.8e2	0.16
5.0e-3	9	6	*IS*	63,127	1.40	0.12e-2	1.02	3	7.0	12	3.3e3	1.38
5.0e-3	7	5	*IS*	47,127	1.04	0.12e-2	1.02	3	8.2	12	2.0e3	0.83
5.0e-3	0	4	*IS*	28,489	0.63	0.17e-2	0.74	3	8.4	10	8.4e2	0.35
1.0e-3	9	6	*IN*	259,057	1.00	0.25e-3	1.12	4	10.0	18	3.0e4	1.00
1.0e-3	9	6	*UN*	253,522	0.98	0.26e-3	0.93	4	10.3	18	3.1e4	1.03
1.0e-3	7	5	*IN*	191,594	0.74	0.25e-3	1.11	4	11.8	17	1.9e4	0.63
1.0e-3	0	4	*IN*	99,137	0.38	0.36e-3	0.85	4	20.0	15	5.9e3	0.20
1.0e-3	9	6	*IS*	339,121	1.31	0.21e-3	1.12	4	6.8	18	4.4e4	1.47
1.0e-3	7	5	*IS*	251,457	0.97	0.21e-3	1.11	4	8.1	18	2.8e4	0.93
1.0e-3	0	4	*IS*	126,113	0.49	0.28e-3	0.86	4	11.6	16	9.2e3	0.31

TABLE 7. The number of degrees of freedom, space fraction, error (in the H^1 seminorm), effectivity index and irregularity on the final grid and the number of refinement levels (lvl), total number of GMRES iterations (nGM), total time and time fraction used by *href(3,e,f,atol,0,m)* for Example 5.2.

atol	e	f	m	N_{dof}	χ_s	$\lvert e\rvert_1$	θ	lvl	% irr	nGM	time	χ_t
5.0e-4	12	8	*IN*	106,873	1.00	0.12e-3	1.23	3	9.8	15	1.0e4	1.00
5.0e-4	12	8	*UN*	104,161	0.97	0.13e-3	0.70	3	9.5	15	1.0e4	1.00
5.0e-4	8	7	*IN*	74,287	0.70	0.12e-3	1.30	3	11.5	12	6.3e3	0.63
5.0e-4	0	5	*IN*	34,615	0.32	0.70e-3	0.22	3	16.3	12	1.5e3	0.15
5.0e-4	12	8	*IS*	148,169	1.39	0.70e-4	1.06	3	5.4	15	1.4e4	1.40
5.0e-4	8	7	*IS*	102,945	0.96	0.70e-4	1.06	3	7.2	12	9.0e3	0.90
5.0e-4	0	5	*IS*	48,977	0.46	0.32e-3	0.28	3	9.1	12	2.3e3	0.23
5.0e-5	12	8	*IN*	730,773	1.00	0.71e-5	1.18	4	6.5	22	1.2e5	1.00
5.0e-5	12	8	*UN*	711,977	0.97	0.98e-5	0.79	4	6.8	23	1.1e5	0.96
5.0e-5	8	7	*IN*	509,474	0.70	0.62e-5	1.17	4	8.4	17	9.7e4	0.88
5.0e-5	0	5	*IN*	217,465	0.30	0.33e-4	0.29	4	12.5	18	2.3e4	0.21
5.0e-5	12	8	*IS*	906,649	1.24	0.50e-5	1.04	4	4.8	22	1.1e5	0.96
5.0e-5	8	7	*IS*	624,961	0.85	0.50e-5	1.04	4	6.4	16	1.1e5	0.96
5.0e-5	0	5	*IS*	267,385	0.37	0.16e-4	0.40	4	9.7	18	3.1e4	0.28

TABLE 8. The number of degrees of freedom, space fraction, error (in the H^1 seminorm), effectivity index and irregularity on the final grid and the number of refinement levels (lvl), total number of GMRES iterations (nGM), total time and time fraction used by *href(4,e,f,atol,0,m)* for Example 5.2.

atol	e	f	m	N_{dof}	χ_s	$\lvert e\rvert_1$	θ	lvl	% irr	nGM	time	χ_t
1.0e-3	15	10	*IN*	104,246	1.00	0.84e-4	1.83	3	15.4	20	1.3e4	1.00
1.0e-3	15	10	*UN*	28,496	0.27	0.57e-3	0.29	2	9.8	12	1.8e3	0.14
1.0e-3	11	9	*IN*	82,468	0.79	0.85e-4	2.79	3	18.3	17	1.1e4	0.82
1.0e-3	9	8	*IN*	59,509	0.57	0.13e-3	3.25	3	22.0	15	6.0e3	0.46
1.0e-3	6	6	*IN*	47,453	0.46	0.88e-3	0.063	4	26.4	17	4.9e3	0.38
1.0e-3	15	10	*IS*	38,161	0.37	0.34e-3	2.16	2	6.1	12	2.4e3	0.18
1.0e-3	11	9	*IS*	31,453	0.30	0.34e-3	2.11	2	7.1	10	1.9e3	0.14
1.0e-3	9	8	*IS*	36,355	0.35	0.18e-3	1.04	2	2.9	9	2.1e3	0.16
1.0e-3	6	6	*IS*	71,277	0.68	0.82e-3	0.09	4	19.4	18	8.2e3	0.62
5.0e-5	15	10	*IN*	203,026	1.00	0.12e-4	1.08	3	7.8	19	3.1e4	1.00
5.0e-5	15	10	*UN*	203,026	1.00	0.12e-4	0.61	3	7.8	19	3.1e4	1.00
5.0e-5	11	9	*IN*	165,907	0.82	0.12e-4	1.08	3	9.0	15	2.5e4	0.81
5.0e-5	9	8	*IN*	121,630	0.60	0.13e-4	1.10	3	12.5	14	1.6e4	0.52
5.0e-5	6	6	*IN*	53,500	0.26	0.30e-3	0.06	3	15.9	12	4.2e3	0.14
5.0e-5	15	10	*IS*	287,691	1.42	0.40e-5	1.27	3	4.4	18	4.1e4	1.32
5.0e-5	11	9	*IS*	235,723	1.17	0.40e-5	1.27	3	5.2	15	3.4e4	1.10
5.0e-5	9	8	*IS*	167,755	0.83	0.40e-5	1.26	3	6.5	14	2.2e4	0.71
5.0e-5	6	6	*IS*	75,987	0.37	0.77e-4	0.09	3	9.6	12	5.8e3	0.19
2.5e-6	15	10	*IN*	1,257,386	1.00			4	6.2			
2.5e-6	15	10	*UN*	1,219,686	0.97			4	6.8			
2.5e-6	11	9	*IN*	1,025,503	0.82			4	7.2			
2.5e-6	9	8	*IN*	739,987	0.59	0.59e-6	1.00	4	8.8	18	2.2e5	
2.5e-6	6	6	*IN*	377,137	0.30	0.54e-5	0.10	4	13.9	16	8.7e4	
2.5e-6	15	10	*IS*	1,768,231	1.41			4	3.9			
2.5e-6	11	9	*IS*	1,444,351	1.15			4	4.5			
2.5e-6	9	8	*IS*	1,021,615	0.81			4	6.8			
2.5e-6	6	6	*IS*	476,293	0.38	0.20e-5	0.12	4	9.3	16	1.1e5	

TABLE 9. The number of degrees of freedom, space fraction, error (in the H^1 seminorm), effectivity index and irregularity on the final grid and the number of refinement levels (lvl), total number of GMRES iterations (nGM), total time and time fraction used by *href(5,e,f,atol,0,m)* for Example 5.2.

level	$e, f = 9, 6$	$e, f = 7, 5$	$e, f = 0, 4$
1	0.7911	0.6729	0.3421
2	0.8750	0.8556	0.4292
3	0.9674	0.9660	0.7429
4	1.1170	1.1111	0.8528

TABLE 10. The effectivity index as a function of the level of refinement, in solving (44) using *href(3,e,f,0.001,0,IN)*.

Example 5.3 Consider the nonlinear reaction-diffusion equation

$$\epsilon \Delta u = u^2, \ \mathbf{x} \in \Omega \equiv (0,1)^3, \tag{46}$$

with

$$u(\mathbf{x}) = 1, \ \mathbf{x} \in \partial\Omega. \tag{47}$$

Since (46) is nonlinear an initial guess $u_0(\mathbf{x})$ of the solution on the coarse grid is needed. Here u_0 is taken to be

$$u_0(\mathbf{x}) = 1 - \sin(\pi x)\sin(\pi y)\sin(\pi z). \tag{48}$$

The solution is nearly zero on the interior with boundary layers (of thickness $O(\sqrt{\epsilon})$ at each of the faces. As in the previous example the initial grid is the $4 \times 4 \times 4$ uniform grid.

Equations (46)-(47) are solved with $\epsilon = 4.0 \times 10^{-3}$, two absolute error tolerances $atol = 5.0 \times 10^{-2}$ and 5.0×10^{-3} and $rtol = 0$. The values of p, e and f considered are those that performed well in the previous example. In all cases $m = IN$. Displayed in Table 11 are N_{dof}, the approximate error $|E|_1$ on the final grid, the number of refinement levels (with 1 indicating the initial grid), the total number of GMRES iterations, nGM, and time. For all runs a maximum of three Newton steps is performed at each level while the Jacobian $\mathbf{J}$ is computed once per level. In all cases this maximum is required at each level. Improvements on this aspect of the algorithm are needed.

When $atol = 5.0 \times 10^{-2}$ optimal performance is achieved with $p = 3$, $e = 0$ and $f = 4$. Runs with suboptimal bases require fewer unknowns and GMRES iterations and shorter times. These results are in agreement with observations in Example 5.2. Solutions obtained with $p = 5$ are slower and require more unknowns than with $p = 3$ and 4. This is primarily due to the fact that for $p = 4$ and 5 only one additional grid was necessary and it was uniform. Thus, the estimated errors for $p = 5$ are considerably less than the tolerance. Results for $p = 2$ are the least efficient involving significantly larger numbers of unknowns on the final grid and CPU times a factor of 10 larger than all but one of the other discretizations (although the error is somewhat smaller).

At the smaller tolerance, with one exception, performance improves as p increases. Storage savings are more dramatic than reductions in CPU time. Again, in all cases suboptimal bases are more efficient than the TP bases with comparable error estimates. Solutions when $p = 2$ could not be obtained with the memory resources available. In two of the cases when $p = 3$ insufficient space was available to generate an effective preconditioner due to the large Jacobian. The result is a suboptimal GMRES performance as evidenced by the large number of iterations. In the remaining cases the number of GMRES iterations needed for convergence was less than 13. This suggests that finding a good preconditioner is more important than the size of the Krylov space for these types of problems. The optimal choice both in terms of storage and time is $p = 5$, $e = 9$ and $f = 8$, although the mSBH basis with $p = 3$ also performs well. This behavior is typical of h-refinement strategies in one dimension where the use of higher p improves the performance as the tolerance decreases [12].

6. Conclusions

Herein I have presented a h-refinement algorithm with various bases and orders for solving reaction-diffusion equations in three dimensions. The bases range from the mSBH basis introduced by Adjerid *et al* [2] to the TP basis for $p = 2, \ldots, 5$.

atol	p	e	f	N_{dof}	$\|E\|_1$	lvl	nGM	time
5.0e-2	2	6	4	838,429	0.25e-1	5	57	3.14e4
5.0e-2	2	6	0	526,261	0.25e-1	5	49	1.68e4
5.0e-2	3	9	6	75,367	0.44e-1	3	35	2.60e3
5.0e-2	3	7	5	56,443	0.43e-1	3	30	1.38e3
5.0e-2	3	0	4	29,155	0.44e-1	3	24	4.33e2
5.0e-2	4	12	8	35,937	0.40e-1	2	28	1.07e3
5.0e-2	4	8	7	25,505	0.41e-1	2	21	8.27e2
5.0e-2	5	15	10	68,921	0.68e-2	2	39	3.86e3
5.0e-2	5	11	9	56,953	0.69e-2	2	33	3.41e3
5.0e-2	5	9	8	41,209	0.69e-2	2	28	2.40e3
5.0e-3	3	9	6	1,963,513	0.40e-2	5	4283	9.90e4
5.0e-3	3	7	5	1,446,675	0.39e-2	5	3677	7.83e4
5.0e-3	3	0	4	716,087	0.25e-2	5	52	3.59e4
5.0e-3	4	12	8	1,030,577	0.20e-2	4	90	7.58e4
5.0e-3	4	8	7	698,923	0.22e-2	4	62	7.00e4
5.0e-3	5	15	10	341,251	0.18e-2	3	76	4.50e4
5.0e-3	5	11	9	280,087	0.19e-2	3	63	3.56e4
5.0e-3	5	9	8	199,999	0.21e-2	3	53	2.46e4

TABLE 11. The number of degrees of freedom, estimated error (in the H^1 seminorm), on the final grid and the number of refinement levels (lvl), total number of GMRES iterations (nGM) and total time used by *href(p,e,f,atol,0,m)* for Example 5.3.

One of two variants of a hierarchical *a posteriori* error estimation strategy drive the adaptivity. Since continuity is enforced across element boundaries irregular nodes, edges and faces are present in the grid. Several grid "smoothing" rules are used to reduce grid irregularity including a sibling rule. The interaction of the choice of basis, error estimation procedure and sibling rule are examined and their impact on the overall performance of the method is investigated for three problems.

For higher-order approximations the mSBH basis is not satisfactory due to poor reliability of the error estimates. Likewise the TP basis is not recommended since it leads to larger and more poorly conditioned linear algebra problems and longer CPU times. Intermediate sub-TP bases are found that offer reliable error estimates and substantial savings in storage and CPU time. The use of the irregular-grid-indicator strategy, especially at high orders, improves reliability. The benefits of the sibling rule (e.g., slightly more accurate error estimates) are not significant enough to overcome its shortcomings (e.g., increased storage and less efficiency).

The data suggest that a strategy that uses the TP basis on the coarsest (initial) grid and reduced bases as the grid is refined might lead to better performance [21]. In most cases the final error was much smaller than the tolerance. This suggests that *hp*-adaptivity is necessary to improve efficiency. *Hp*-adaptivity is also needed to enhance reliability since on coarsest grids the error estimates are not as accurate for higher p. Clearly the strategy used in solving the nonlinear system has a critical impact on the overall performance of the algorithm and further investigation is needed.

7. Appendix A

Proof of Lemma 2. Since Δ_Ω is uniform let $H_x = h_i^x$, $H_y = h_i^y$ and $H_z = h_i^z$ for any i. Let $\Delta_i \in \Delta_\Omega$ and $\hat{\mathcal{S}}^{p,e,f} = \mathcal{S}^{p,TP} \backslash \mathcal{S}^{p,e,f}$. Then

$$U_I - U_I^{e,f} = \sum_{(l,m,n)\in\hat{\mathcal{S}}^{p,e,f}} \bar{U}_{l,m,n}^i \Phi_l(x;H_x;m_i^x)\Phi_m(y;H_y;m_i^y)\Phi_n(z;H_z;m_i^z). \tag{49}$$

Alternatively using the notation of [17], (49) can be rewritten as

$$U_I - U_I^{e,f} = \sum_{(l,m,n)\in\hat{\mathcal{S}}^{p,e,f}} \tilde{U}_{l,m,n}^i \tilde{\Phi}_l(x;H_x;m_i^x)\tilde{\Phi}_m(y;H_y;m_i^y)\tilde{\Phi}_n(z;H_z;m_i^z). \tag{50}$$

where

$$\tilde{U}_{l,m,n}^i = c_{l,m,n} H_x^l H_y^m H_z^n u^{(l,m,n)}(\mathbf{m}^i) + O(H^{l+m+n+2}), \tag{51}$$

and

$$\tilde{\Phi}_j(w;H_w;m_i^w) = \begin{cases} 1 & j=0, \\ \frac{2}{H_w}(w-m_i^w) & j=1, \\ \Phi_j(w;H_w;m_i^w) & j\geq 2, \end{cases} = \begin{cases} P_0(w) & j=0, \\ P_1(\frac{2}{H_w}(w-m_i^w)) & j=1, \\ \sqrt{\frac{2j-1}{2}}\int_{-1}^{2(w-m_i^w)/H_w} P_{j-1}(s)ds & j\geq 2. \end{cases} \tag{52}$$

From (50)-(52) it follows that

$$\begin{aligned} \|(U_I &- U_I^{e,f})^{(1,0,0)}\|_{0,\Delta_i}^2 \\ &= \sum_{(l,m,n)\in\hat{\mathcal{S}}^{p,e,f}} \sum_{(\bar{l},\bar{m},\bar{n})\in\hat{\mathcal{S}}^{p,e,f}} (c_{l,m,n}c_{\bar{l},\bar{m},\bar{n}} H_x^{l+\bar{l}} H_y^{m+\bar{m}} H_z^{n+\bar{n}} u^{(l,m,n)}(\mathbf{m}^i) u^{(\bar{l},\bar{m},\bar{n})}(\mathbf{m}^i) \\ &+ O(H^{l+m+n+\bar{l}+\bar{m}+\bar{n}})) \int_{m_x^i-H_x/2}^{m_x^i+H_x/2} \tilde{\Phi}_l'(x;H_x;m_x^i)\tilde{\Phi}_{\bar{l}}'(x;H_x;m_x^i)dx \\ &\times \int_{m_y^i-H_y/2}^{m_y^i+H_y/2} \tilde{\Phi}_m(y;H_y;m_y^i)\tilde{\Phi}_{\bar{m}}(y;H_y;m_y^i)dy \int_{m_z^i-H_z/2}^{m_z^i+H_z/2} \tilde{\Phi}_n(z;H_z;m_z^i)\tilde{\Phi}_{\bar{n}}(z;H_z;m_z^i)dz. \end{aligned} \tag{53}$$

Using (52) yields

$$\begin{aligned} \int_{m_x^i-H_x/2}^{m_x^i+H_x/2} \tilde{\Phi}_l'(x;H_x;m_x^i)\tilde{\Phi}_{\bar{l}}'(x;H_x;m_x^i)dx &= O(H_x^{-1}), \\ \int_{m_y^i-H_y/2}^{m_y^i+H_y/2} \tilde{\Phi}_m(y;H_y;m_y^i)\tilde{\Phi}_{\bar{m}}(y;H_y;m_y^i)dy &= O(H_y), \\ \int_{m_z^i-H_z/2}^{m_z^i+H_z/2} \tilde{\Phi}_n(z;H_z;m_z^i)\tilde{\Phi}_{\bar{n}}(z;H_z;m_z^i)dz &= O(H_z). \end{aligned} \tag{54}$$

Combining (53) and (54) and summing over i gives

$$\|(U_I - U_I^{e,f})^{(1,0,0)}\|_0^2 = O(H^{l+m+n+\bar{l}+\bar{m}+\bar{n}-2}). \tag{55}$$

with comparable results for $\|(U_I - U_I^{e,f})^{(0,1,0)}\|_0^2$ and $\|(U_I - U_I^{e,f})^{(0,0,1)}\|_0^2$. The same techniques yield

$$\|(U_I - U_I^{e,f})\|_0^2 = O(H^{l+m+n+\bar{l}+\bar{m}+\bar{n}+1}). \tag{56}$$

The result follows from (51), (55) and (56) since $l+m+n \geq \min(e,f)+1$ and $\bar{l}+\bar{m}+\bar{n} \geq \min(e,f)+1$. □

8. Appendix B

The formulas for the interpolation error on the elements shown in Figure 1 for $p = 2, 3, 4$ and 5 are given below (formulas for $p = 1$ are given in [15]). For simplicity the midpoints of the elements are assumed to be $\mathbf{0} = (0,0,0)$. All formulas are computed using Maple. In the case of $p = 2$ and $p = 4$ some of the terms change signs for symmetrical configurations.

p = 2

Case 1:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^5 h_y h_z}{720}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y^7 h_z}{1,440 h_x}(u^{(0,3,0)}(\mathbf{0}))^2 + O(H^8),\\
\|e_y\|_0^2 &= \frac{h_x h_y^5 h_z}{270}(u^{(0,3,0)}(\mathbf{0}))^2 + O(H^8),\\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^5}{720}(u^{(0,0,3)}(\mathbf{0}))^2 + \frac{h_x h_y^7}{1,440 h_z}(u^{(0,3,0)}(\mathbf{0}))^2 + O(H^8).
\end{aligned} \tag{57}$$

Case 2:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^5 h_y h_z}{270}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y^7 h_z}{1,440 h_x}(u^{(0,3,0)}(\mathbf{0}))^2 + O(H^8),\\
\|e_y\|_0^2 &= \frac{h_x h_y^5 h_z}{270}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_x^7 h_z}{1,440 h_y}(u^{(3,0,0)}(\mathbf{0}))^2 + O(H^8),\\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^5}{720}(u^{(0,0,3)}(\mathbf{0}))^2 + \frac{h_x^7 h_y}{1,440 h_z}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_y^7}{1,440 h_z}(u^{(0,3,0)}(\mathbf{0}))^2\\
&- \frac{h_x^4 h_y^4}{1,152 h_z}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,3,0)}(\mathbf{0})) + O(H^8).
\end{aligned} \tag{58}$$

Case 3a:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^5 h_y h_z}{720}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y^7 h_z}{480 h_x}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_y h_z^7}{480 h_x}(u^{(0,0,3)}(\mathbf{0}))^2\\
&- \frac{h_y^4 h_z^4}{288 h_x}(u^{(0,3,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8),\\
\|e_y\|_0^2 &= \frac{h_x h_y^5 h_z}{120}(u^{(0,3,0)}(\mathbf{0}))^2 + O(H^8),\\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^5}{120}(u^{(0,0,3)}(\mathbf{0}))^2 + O(H^8).
\end{aligned} \tag{59}$$

Case 3b:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^5 h_y h_z}{270}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y^7 h_z}{1,440 h_x}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_y h_z^7}{1,440 h_x}(u^{(0,0,3)}(\mathbf{0}))^2\\
&- \frac{h_y^4 h_z^4}{1,152 h_x}(u^{(0,3,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8),\\
\|e_y\|_0^2 &= \frac{h_x h_y^5 h_z}{270}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_x^7 h_z}{1,440 h_y}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_z^7}{1,440 h_y}(u^{(0,0,3)}(\mathbf{0}))^2\\
&+ \frac{h_x^4 h_z^4}{1,152 h_y}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8),\\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^5}{270}(u^{(0,0,3)}(\mathbf{0}))^2 + \frac{h_x^7 h_x}{1,440 h_z}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_y^7}{1,440 h_z}(u^{(0,3,0)}(\mathbf{0}))^2\\
&- \frac{h_x^4 h_y^4}{1,152 h_z}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,3,0)}(\mathbf{0})) + O(H^8).
\end{aligned} \tag{60}$$

Case 4:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^5 h_y h_z}{120}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y h_z^7}{1,440 h_x}(u^{(0,0,3)}(\mathbf{0}))^2 + O(H^8),\\
\|e_y\|_0^2 &= \frac{h_x h_y^5 h_z}{120}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_x h_z^7}{1,440 h_y}(u^{(0,0,3)}(\mathbf{0}))^2 + O(H^8),\\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^5}{270}(u^{(0,0,3)}(\mathbf{0}))^2 + \frac{h_x^7 h_y}{480 h_z}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_y^7}{480 h_z}(u^{(0,3,0)}(\mathbf{0}))^2\\
&- \frac{h_x^4 h_y^4}{288 h_z}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,3,0)}(\mathbf{0})) + O(H^8).
\end{aligned} \tag{61}$$

Case 5:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^5 h_y h_z}{120}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y^7 h_z}{480 h_x}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_y h_z^7}{1,440 h_x}(u^{(0,0,3)}(\mathbf{0}))^2 \\
&- \frac{h_y^4 h_z^4}{576 h_x}(u^{(0,3,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8), \\
\|e_y\|_0^2 &= \frac{h_x h_y^5 h_z}{120}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_x^7 h_z}{480 h_y}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_z^7}{1,440 h_y}(u^{(0,0,3)}(\mathbf{0}))^2 \\
&+ \frac{h_x^4 h_z^4}{576 h_y}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8), \\
\|e_z\|_0^2 &= \frac{61 h_x h_y h_z^5}{4,320}(u^{(0,0,3)}(\mathbf{0}))^2 + O(H^8).
\end{aligned} \tag{62}$$

Case 6:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{61 h_x^5 h_y h_z}{4,320}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_y^7 h_z}{1,440 h_x}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_y h_z^7}{1,440 h_x}(u^{(0,0,3)}(\mathbf{0}))^2 \\
&- \frac{h_y^4 h_z^4}{1,152 h_x}(u^{(0,3,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8), \\
\|e_y\|_0^2 &= \frac{61 h_x h_y^5 h_z}{4,320}(u^{(0,3,0)}(\mathbf{0}))^2 + \frac{h_x^7 h_z}{1,440 h_y}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_z^7}{1,440 h_y}(u^{(0,0,3)}(\mathbf{0}))^2 \\
&+ \frac{h_x^4 h_z^4}{1,152 h_y}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,0,3)}(\mathbf{0})) + O(H^8), \\
\|e_z\|_0^2 &= \frac{61 h_x h_y h_z^5}{4,320}(u^{(0,0,3)}(\mathbf{0}))^2 + \frac{h_x^7 h_y}{1,440 h_z}(u^{(3,0,0)}(\mathbf{0}))^2 + \frac{h_x h_y^7}{1,440 h_z}(u^{(0,3,0)}(\mathbf{0}))^2 \\
&- \frac{h_x^4 h_y^4}{1,152 h_z}(u^{(3,0,0)}(\mathbf{0}))(u^{(0,3,0)}(\mathbf{0})) + O(H^8).
\end{aligned} \tag{63}$$

p = 3

Case 1:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^7 h_y h_z}{100,800}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11 h_y^9 h_z}{1,134,000 h_x}(u^{(0,4,0)}(\mathbf{0}))^2 + O(H^{10}), \\
\|e_y\|_0^2 &= \frac{h_x h_y^7 h_z}{12,600}(u^{(0,4,0)}(\mathbf{0}))^2 + O(H^{10}), \\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^7}{100,800}(u^{(0,0,4)}(\mathbf{0}))^2 + \frac{11 h_x h_y^9}{1,134,000 h_z}(u^{(0,4,0)}(\mathbf{0}))^2 + O(H^{10}).
\end{aligned} \tag{64}$$

Case 2:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^7 h_y h_z}{12,600}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11 h_y^9 h_z}{1,134,000 h_x}(u^{(0,4,0)}(\mathbf{0}))^2 \\
&+ \frac{h_x^3 h_y^5 h_z}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0})) + O(H^{10}), \\
\|e_y\|_0^2 &= \frac{h_x h_y^7 h_z}{12,600}(u^{(0,4,0)}(\mathbf{0}))^2 + \frac{11 h_x^9 h_z}{1,134,000 h_y}(u^{(4,0,0)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5 h_y^3 h_z}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0})) + O(H^{10}), \\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^7}{100,800}(u^{(0,0,4)}(\mathbf{0}))^2 + \frac{11 h_x^9 h_y}{1,134,000 h_z}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11 h_x h_y^9}{1,134,000 h_z}(u^{(0,4,0)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5 h_y^5}{259,200 h_z}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0})) + O(H^{10}).
\end{aligned} \tag{65}$$

Case 3a:

$$
\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^7 h_y h_z}{100,800}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11h_y^9 h_z}{378,000h_x}(u^{(0,4,0)}(\mathbf{0}))^2 \\
&+ \frac{11h_y h_z^9}{378,000h_x}(u^{(0,0,4)}(\mathbf{0}))^2 + O(H^{10}), \\
\|e_y\|_0^2 &= \frac{11h_x h_y^7 h_z}{50,400}(u^{(0,4,0)}(\mathbf{0}))^2 + O(H^{10}), \\
\|e_z\|_0^2 &= \frac{11h_x h_y h_z^7}{50,400}(u^{(0,0,4)}(\mathbf{0}))^2 + O(H^{10}).
\end{aligned} \tag{66}
$$

Case 3b:

$$
\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^7 h_y h_z}{12,600}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11h_y^9 h_z}{1,134,000h_x}(u^{(0,4,0)}(\mathbf{0}))^2 + \frac{11h_y h_z^9}{1,134,000h_x}(u^{(0,0,4)}(\mathbf{0}))^2 \\
&+ \frac{h_y^5 h_z^5}{259,200h_x}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + \frac{h_x^3 h_y^5 h_z}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0})) \\
&+ \frac{h_x^3 h_y h_z^5}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + O(H^{10}), \\
\|e_y\|_0^2 &= \frac{h_x h_y^7 h_z}{12,600}(u^{(0,4,0)}(\mathbf{0}))^2 + \frac{11h_x^9 h_z}{1,134,000h_y}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11h_x h_z^9}{1,134,000h_y}(u^{(0,0,4)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5 h_z^5}{259,200h_y}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + \frac{h_x^5 h_y^3 h_z}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0})) \\
&+ \frac{h_x h_y^3 h_z^5}{129,600}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + O(H^{10}), \\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^7}{12,600}(u^{(0,0,4)}(\mathbf{0}))^2 + \frac{11h_x^9 h_y}{1,134,000h_z}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11h_x h_y^9}{1,134,000h_z}(u^{(0,4,0)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5 h_y^5}{259,200h_z}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0})) + \frac{h_x^5 h_y h_z^3}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) \\
&+ \frac{h_x h_y^5 h_z^3}{129,600}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + O(H^{10}).
\end{aligned} \tag{67}
$$

Case 4:

$$
\begin{aligned}
\|e_x\|_0^2 &= \frac{11h_x^7 h_y h_z}{50,400}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11h_y h_z^9}{1,134,000h_x}(u^{(0,0,4)}(\mathbf{0}))^2 \\
&+ \frac{h_x^3 h_y h_z^5}{86,400}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + O(H^{10}), \\
\|e_y\|_0^2 &= \frac{11h_x h_y^7 h_z}{50,400}(u^{(0,4,0)}(\mathbf{0}))^2 + \frac{11h_x h_z^9}{1,134,000h_y}(u^{(0,0,4)}(\mathbf{0}))^2 \\
&+ \frac{h_x h_y^3 h_z^5}{86,400}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + O(H^{10}), \\
\|e_z\|_0^2 &= \frac{h_x h_y h_z^7}{12,600}(u^{(0,0,4)}(\mathbf{0}))^2 + \frac{11h_x^9 h_y}{378,000h_z}(u^{(4,0,0)}(\mathbf{0}))^2 + \frac{11h_x h_y^9}{378,000h_z}(u^{(0,4,0)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5 h_y h_z^3}{86,400}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + \frac{h_x h_y^5 h_z^3}{86,400}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0})) + O(H^{10}).
\end{aligned} \tag{68}
$$

Case 5:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{11h_x^7h_yh_z}{50,400}(u^{(4,0,0)}(\mathbf{0}))^2+\frac{11h_y^9h_z}{378,000h_x}(u^{(0,4,0)}(\mathbf{0}))^2+\frac{h_yh_z^9}{378,000h_x}(u^{(0,0,4)}(\mathbf{0}))^2\\
&+ \frac{h_x^3h_y^5h_z}{43,200}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0}))+O(H^{10}),\\
\|e_y\|_0^2 &= \frac{h_xh_y^7h_z}{50,400}(u^{(0,4,0)}(\mathbf{0}))^2+\frac{11h_x^9h_z}{378,000h_y}(u^{(4,0,0)}(\mathbf{0}))^2+\frac{11h_xh_z^9}{378,000h_y}(u^{(0,0,4)}(\mathbf{0}))^2\\
&+ \frac{h_x^5h_y^3h_z}{43,200}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0}))+O(H^{10}),\\
\|e_z\|_0^2 &= \frac{79h_xh_yh_z^7}{201,600}(u^{(0,0,4)}(\mathbf{0}))^2+O(H^{10}).
\end{aligned}$$

(69)

Case 6:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{79h_x^7h_yh_z}{201,600}(u^{(4,0,0)}(\mathbf{0}))^2+\frac{11h_y^9h_z}{1,134,000h_x}(u^{(0,4,0)}(\mathbf{0}))^2+\frac{11h_yh_z^9}{1,134,000h_x}(u^{(0,0,4)}(\mathbf{0}))^2\\
&+ \frac{h_y^5h_z^5}{259,200h_x}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0}))+\frac{h_x^3h_y^5h_z}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0}))\\
&+ \frac{h_x^3h_yh_z^5}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0}))+O(H^{10}),\\
\|e_y\|_0^2 &= \frac{79h_xh_y^7h_z}{201,600}(u^{(0,4,0)}(\mathbf{0}))^2+\frac{11h_x^9h_z}{1,134,000h_y}(u^{(4,0,0)}(\mathbf{0}))^2+\frac{11h_xh_z^9}{1,134,000h_y}(u^{(0,0,4)}(\mathbf{0}))^2\\
&+ \frac{h_x^5h_z^5}{259,200h_y}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0}))+\frac{h_x^5h_y^3h_z}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0}))\\
&+ \frac{h_xh_y^3h_z^5}{129,600}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0}))+O(H^{10}),\\
\|e_z\|_0^2 &= \frac{79h_xh_yh_z^7}{201,600}(u^{(0,0,4)}(\mathbf{0}))^2+\frac{11h_x^9h_y}{1,134,000h_z}(u^{(4,0,0)}(\mathbf{0}))^2+\frac{11h_xh_y^9}{1,134,000h_z}(u^{(0,4,0)}(\mathbf{0}))^2\\
&+ \frac{h_x^5h_y^5}{259,200h_z}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,4,0)}(\mathbf{0}))+\frac{h_x^5h_yh_z^3}{129,600}(u^{(4,0,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0}))\\
&+ \frac{h_xh_y^5h_z^3}{129,600}(u^{(0,4,0)}(\mathbf{0}))(u^{(0,0,4)}(\mathbf{0}))+O(H^{10}).
\end{aligned}$$

(70)

p = 4

Case 1:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^9h_yh_z}{25,401,600}(u^{(5,0,0)}(\mathbf{0}))^2+\frac{19h_y^{11}h_z}{213,373,440h_x}(u^{(0,5,0)}(\mathbf{0}))^2+O(H^{12}),\\
\|e_y\|_0^2 &= \frac{11h_xh_y^9h_z}{9,525,600}(u^{(0,5,0)}(\mathbf{0}))^2+O(H^{12}),\\
\|e_z\|_0^2 &= \frac{h_xh_yh_z^9}{25,401,600}(u^{(0,0,5)}(\mathbf{0}))^2+\frac{19h_xh_y^{11}}{213,373,440h_z}(u^{(0,5,0)}(\mathbf{0}))^2+O(H^{12}).
\end{aligned}$$

(71)

Case 2:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{11h_x^9h_yh_z}{9,525,600}(u^{(5,0,0)}(\mathbf{0}))^2+\frac{19h_y^{11}h_z}{213,373,440h_x}(u^{(0,5,0)}(\mathbf{0}))^2+O(H^{12}),\\
\|e_y\|_0^2 &= \frac{11h_xh_y^9h_z}{9,525,600}(u^{(0,5,0)}(\mathbf{0}))^2+\frac{19h_x^{11}h_z}{213,373,440h_y}(u^{(5,0,0)}(\mathbf{0}))^2+O(H^{12}),\\
\|e_z\|_0^2 &= \frac{h_xh_yh_z^9}{25,401,600}(u^{(0,0,5)}(\mathbf{0}))^2+\frac{19h_x^{11}h_y}{213,373,440h_z}(u^{(5,0,0)}(\mathbf{0}))^2\\
&+ \frac{19h_xh_y^{11}}{213,373,440h_z}(u^{(0,5,0)}(\mathbf{0}))^2+O(H^{12}).
\end{aligned}$$

(72)

Case 3a:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^9 h_y h_z}{25,401,600}(u^{(5,0,0)}(\mathbf{0}))^2 + \frac{19h_y^{11}h_z}{71,124,480h_x}(u^{(0,5,0)}(\mathbf{0}))^2 \\
&+ \frac{19h_y h_z^{11}}{71,124,480h_x}(u^{(0,0,5)}(\mathbf{0}))^2 - \frac{h_y^6 h_z^6}{12,700,800h_x}(u^{(0,5,0)}(\mathbf{0}))(u^{(0,0,5)}(\mathbf{0})) + O(H^{12}), \\
\|e_y\|_0^2 &= \frac{43h_x h_y^9 h_z}{12,700,800}(u^{(0,5,0)}(\mathbf{0}))^2 + O(H^{12}), \\
\|e_z\|_0^2 &= \frac{43h_x h_y h_z^9}{12,700,800}(u^{(0,0,5)}(\mathbf{0}))^2 + O(H^{12}).
\end{aligned} \tag{73}$$

Case 3b:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{11h_x^9 h_y h_z}{9,525,600}(u^{(5,0,0)}(\mathbf{0}))^2 + \frac{19h_y^{11}h_z}{213,373,440h_x}(u^{(0,5,0)}(\mathbf{0}))^2 \\
&+ \frac{19h_y h_z^{11}}{213,373,440h_x}(u^{(0,0,5)}(\mathbf{0}))^2 + O(H^{12}), \\
\|e_y\|_0^2 &= \frac{11h_x h_y^9 h_z}{9,525,600}(u^{(0,5,0)}(\mathbf{0}))^2 + \frac{19h_x^{11}h_z}{213,373,440h_y}(u^{(5,0,0)}(\mathbf{0}))^2 \\
&+ \frac{19h_x h_z^{11}}{213,373,440h_y}(u^{(0,0,5)}(\mathbf{0}))^2 + O(H^{12}), \\
\|e_z\|_0^2 &= \frac{11h_x h_y h_z^9}{9,525,600}(u^{(0,0,5)}(\mathbf{0}))^2 + \frac{19h_x^{11}h_y}{213,373,440h_z}(u^{(5,0,0)}(\mathbf{0}))^2 \\
&+ \frac{19h_x h_y^{11}}{213,373,440h_z}(u^{(0,5,0)}(\mathbf{0}))^2 + O(H^{12}).
\end{aligned} \tag{74}$$

Case 4:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{43h_x^9 h_y h_z}{12,700,800}(u^{(5,0,0)}(\mathbf{0}))^2 + \frac{19h_y h_z^{11}}{213,373,440h_x}(u^{(0,0,5)}(\mathbf{0}))^2 + O(H^{12}), \\
\|e_y\|_0^2 &= \frac{43h_x h_y^9 h_z}{12,700,800}(u^{(0,5,0)}(\mathbf{0}))^2 + \frac{19h_x h_z^{11}}{213,373,440h_y}(u^{(0,0,5)}(\mathbf{0}))^2 + O(H^{12}), \\
\|e_z\|_0^2 &= \frac{11h_x h_y h_z^9}{9,525,600}(u^{(0,0,5)}(\mathbf{0}))^2 + \frac{19h_x^{11}h_y}{71,124,480h_z}(u^{(5,0,0)}(\mathbf{0}))^2 + \frac{19h_x h_y^{11}}{71,124,480h_z}(u^{(0,5,0)}(\mathbf{0}))^2 \\
&- \frac{h_x^6 h_y^6}{12,700,800h_z}(u^{(5,0,0)}(\mathbf{0}))(u^{(0,5,0)}(\mathbf{0})) + O(H^{12}).
\end{aligned} \tag{75}$$

Case 5:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{43h_x^9 h_y h_z}{12,700,800}(u^{(5,0,0)}(\mathbf{0}))^2 + \frac{19h_y^{11}h_z}{71,124,480h_x}(u^{(0,5,0)}(\mathbf{0}))^2 \\
&+ \frac{19h_y h_z^{11}}{213,373,400h_x}(u^{(0,0,5)}(\mathbf{0}))^2 - \frac{h_y^6 h_z^6}{12,700,800h_x}(u^{(0,5,0)}(\mathbf{0}))(u^{(0,0,5)}(\mathbf{0})) + O(H^{12}), \\
\|e_y\|_0^2 &= \frac{43h_x h_y^9 h_z}{12,700,800}(u^{(0,5,0)}(\mathbf{0}))^2 + \frac{19h_x^{11}h_z}{71,124,480h_y}(u^{(5,0,0)}(\mathbf{0}))^2 \\
&+ \frac{19h_x h_z^{11}}{213,373,400h_y}(u^{(0,0,5)}(\mathbf{0}))^2 + \frac{h_x^6 h_z^6}{12,700,800h_y}(u^{(5,0,0)}(\mathbf{0}))(u^{(0,0,5)}(\mathbf{0})) + O(H^{12}), \\
\|e_z\|_0^2 &= \frac{941h_x h_y h_z^9}{152,409,600}(u^{(0,0,5)}(\mathbf{0}))^2 + O(H^{12}).
\end{aligned} \tag{76}$$

Case 6:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{941h_x^9h_yh_z}{152,409,600}(u^{(5,0,0)}(\mathbf{0}))^2+\frac{19h_y^{11}h_z}{213,373,440h_x}(u^{(0,5,0)}(\mathbf{0}))^2\\
&+ \frac{19h_yh_z^{11}}{213,373,440h_x}(u^{(0,0,5)}(\mathbf{0}))^2-\frac{h_y^6h_z^6}{12,700,800h_x}(u^{(0,5,0)}(\mathbf{0}))(u^{(0,0,5)}(\mathbf{0}))+O(H^{12}),\\
\|e_y\|_0^2 &= \frac{941h_xh_y^9h_z}{152,409,600}(u^{(0,5,0)}(\mathbf{0}))^2+\frac{19h_x^{11}h_z}{213,373,440h_y}(u^{(5,0,0)}(\mathbf{0}))^2\\
&+ \frac{19h_xh_z^{11}}{213,373,440h_y}(u^{(0,0,5)}(\mathbf{0}))^2+\frac{h_x^6h_z^6}{12,700,800h_y}(u^{(5,0,0)}(\mathbf{0}))(u^{(0,0,5)}(\mathbf{0}))+O(H^{12}),\\
\|e_z\|_0^2 &= \frac{941h_xh_yh_z^9}{152,409,600}(u^{(0,0,5)}(\mathbf{0}))^2+\frac{19h_x^{11}h_y}{213,373,440h_z}(u^{(5,0,0)}(\mathbf{0}))^2\\
&+ \frac{19h_xh_y^{11}}{213,373,440h_z}(u^{(0,5,0)}(\mathbf{0}))^2+\frac{h_x^6h_y^6}{12,700,800h_z}(u^{(5,0,0)}(\mathbf{0}))(u^{(0,5,0)}(\mathbf{0}))+O(H^{12}).
\end{aligned}$$

(77)

p = 5

Case 1:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^{11}h_yh_z}{10,059,033,600}(u^{(6,0,0)}(\mathbf{0}))^2+\frac{53h_y^{13}h_z}{90,531,302,400h_x}(u^{(0,6,0)}(\mathbf{0}))^2+O(H^{14}),\\
\|e_y\|_0^2 &= \frac{43h_xh_y^{11}h_z}{3,772,137,600}(u^{(0,6,0)}(\mathbf{0}))^2+O(H^{14}),\\
\|e_z\|_0^2 &= \frac{h_xh_yh_z^{11}}{10,059,033,600}(u^{(0,0,6)}(\mathbf{0}))^2+\frac{53h_xh_y^{13}}{90,531,302,400h_z}(u^{(0,6,0)}(\mathbf{0}))^2+O(H^{14}).
\end{aligned}$$

(78)

Case 2:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{43h_x^{11}h_yh_z}{3,772,137,600}(u^{(6,0,0)}(\mathbf{0}))^2+\frac{53h_y^{13}h_z}{90,531,302,400h_x}(u^{(0,6,0)}(\mathbf{0}))^2\\
&+ \frac{h_x^5h_y^7h_z}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0}))+O(H^{14}),\\
\|e_y\|_0^2 &= \frac{43h_xh_y^{11}h_z}{3,772,137,600}(u^{(0,6,0)}(\mathbf{0}))^2+\frac{53h_x^{13}h_z}{90,531,302,400h_y}(u^{(6,0,0)}(\mathbf{0}))^2\\
&+ \frac{h_x^7h_y^5h_z}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0}))+O(H^{14}),\\
\|e_z\|_0^2 &= \frac{h_xh_yh_z^{11}}{10,059,033,600}(u^{(0,0,6)}(\mathbf{0}))^2+\frac{53h_x^{13}h_y}{90,531,302,400h_z}(u^{(6,0,0)}(\mathbf{0}))^2\\
&+ \frac{53h_xh_y^{13}}{90,531,302,400h_z}(u^{(0,6,0)}(\mathbf{0}))^2+\frac{h_x^7h_y^7}{65,840,947,200h_z}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0}))\\
&+ O(H^{14}).
\end{aligned}$$

(79)

Case 3a:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{h_x^{11}h_yh_z}{10,059,033,600}(u^{(6,0,0)}(\mathbf{0}))^2+\frac{53h_y^{13}h_z}{30,177,100,800h_x}(u^{(0,6,0)}(\mathbf{0}))^2\\
&+ \frac{53h_yh_z^{13}}{30,177,100,800h_x}(u^{(0,0,6)}(\mathbf{0}))^2+O(H^{14}),\\
\|e_y\|_0^2 &= \frac{19h_xh_y^{11}h_z}{558,835,200}(u^{(0,6,0)}(\mathbf{0}))^2+O(H^{14}),\\
\|e_z\|_0^2 &= \frac{19h_xh_yh_z^{11}}{558,835,200}(u^{(0,0,6)}(\mathbf{0}))^2+O(H^{14}).
\end{aligned}$$

(80)

Case 3b:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{43h_x^{11}h_yh_z}{3,772,137,600}(u^{(6,0,0)}(\mathbf{0}))^2 + \frac{53h_y^{13}h_z}{90,531,302,400h_x}(u^{(0,6,0)}(\mathbf{0}))^2 \\
&+ \frac{53h_yh_z^{13}}{90,531,302,400h_x}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{h_x^5h_y^7h_z}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) \\
&+ \frac{h_x^5h_yh_z^7}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + \frac{h_y^7h_z^7}{65,840,947,200h_x}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) \\
&+ O(H^{14}), \\
\|e_y\|_0^2 &= \frac{43h_xh_y^{11}h_z}{3,772,137,600}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{53h_x^{13}h_z}{90,531,302,400h_y}(u^{(6,0,0)}(\mathbf{0}))^2 \\
&+ \frac{53h_xh_z^{13}}{90,531,302,400h_y}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{h_x^7h_y^5h_z}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) \\
&+ \frac{h_xh_y^5h_z^7}{8,230,118,400}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + \frac{h_x^7h_z^7}{65,840,947,200h_y}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) \\
&+ O(H^{14}), \\
\|e_z\|_0^2 &= \frac{43h_xh_yh_z^{11}}{3,772,137,600}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{53h_x^{13}h_y}{90,531,302,400h_z}(u^{(6,0,0)}(\mathbf{0}))^2 \\
&+ \frac{53h_xh_y^{13}}{90,531,302,400h_z}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{h_x^7h_yh_z^5}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) \\
&+ \frac{h_xh_y^7h_z^5}{8,230,118,400}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + \frac{h_x^7h_y^7}{65,840,947,200h_z}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) \\
&+ O(H^{14}).
\end{aligned} \tag{81}$$

Case 4:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{19h_x^{11}h_yh_z}{558,835,200}(u^{(6,0,0)}(\mathbf{0}))^2 + \frac{53h_yh_z^{13}}{90,531,302,400h_x}(u^{(0,0,6)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5h_yh_z^7}{5,486,745,600}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + O(H^{14}), \\
\|e_y\|_0^2 &= \frac{19h_xh_y^{11}h_z}{558,835,200}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{53h_xh_z^{13}}{90,531,302,400h_y}(u^{(0,0,6)}(\mathbf{0}))^2 \\
&+ \frac{h_xh_y^5h_z^7}{5,486,745,600}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + O(H^{14}), \\
\|e_z\|_0^2 &= \frac{43h_xh_yh_z^{11}}{3,772,137,600}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{53h_x^{13}h_y}{30,177,100,800h_z}(u^{(6,0,0)}(\mathbf{0}))^2 \\
&+ \frac{53h_xh_y^{13}}{30,177,100,800h_z}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{h_x^7h_yh_z^5}{5,486,745,600}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) \\
&+ \frac{h_xh_y^7h_z^5}{5,486,745,600}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + O(H^{14}).
\end{aligned} \tag{82}$$

Case 5:

$$\begin{aligned}
\|e_x\|_0^2 &= \frac{19h_x^{11}h_yh_z}{558,835,200}(u^{(6,0,0)}(\mathbf{0}))^2 + \frac{h_x^5h_y^7h_z}{2,743,372,800}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) \\
&+ \frac{53h_y^{13}h_z}{30,177,100,800h_x}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{53h_yh_z^{13}}{90,531,302,400h_x}(u^{(0,0,6)}(\mathbf{0}))^2 + O(H^{14}), \\
\|e_y\|_0^2 &= \frac{19h_xh_y^{11}h_z}{558,835,200}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{h_x^7h_y^5h_z}{2,743,372,800}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) \\
&+ \frac{53h_x^{13}h_z}{30,177,100,800h_y}(u^{(6,0,0)}(\mathbf{0}))^2 + \frac{53h_xh_z^{13}}{90,531,302,400h_y}(u^{(0,0,6)}(\mathbf{0}))^2 + O(H^{14}), \\
\|e_z\|_0^2 &= \frac{3757h_xh_yh_z^{11}}{60,354,201,600}(u^{(0,0,6)}(\mathbf{0}))^2 + O(H^{14}).
\end{aligned} \tag{83}$$

Case 6:

$$
\begin{aligned}
\|e_x\|_0^2 &= \frac{3757h_x^{11}h_yh_z}{60,354,201,600}(u^{(6,0,0)}(\mathbf{0}))^2 + \frac{53h_y^{13}h_z}{90,531,302,400h_x}(u^{(0,6,0)}(\mathbf{0}))^2 \\
&+ \frac{h_x^5h_yh_z^7}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + \frac{h_y^7h_z^7}{65,840,947,200h_x}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) \\
&+ \frac{53h_yh_z^{13}}{90,531,302,400h_x}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{h_x^5h_y^7h_z}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) + O(H^{14}), \\
\|e_y\|_0^2 &= \frac{3757h_xh_y^{11}h_z}{60,354,201,600}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{53h_x^{13}h_z}{90,531,302,400h_y}(u^{(6,0,0)}(\mathbf{0}))^2 \\
&+ \frac{h_xh_y^5h_z^7}{8,230,118,400}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + \frac{h_x^7h_z^7}{65,840,947,200h_y}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) \\
&+ \frac{53h_xh_z^{13}}{90,531,302,400h_y}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{h_x^7h_y^5h_z}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) + O(H^{14}), \\
\|e_z\|_0^2 &= \frac{3757h_xh_yh_z^{11}}{60,354,201,600}(u^{(0,0,6)}(\mathbf{0}))^2 + \frac{53h_x^{13}h_y}{90,531,302,400h_z}(u^{(6,0,0)}(\mathbf{0}))^2 \\
&+ \frac{h_xh_y^7h_z^5}{8,230,118,400}(u^{(0,6,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + \frac{h_x^7h_y^7}{65,840,947,200h_z}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,6,0)}(\mathbf{0})) \\
&+ \frac{53h_xh_y^{13}}{90,531,302,400h_z}(u^{(0,6,0)}(\mathbf{0}))^2 + \frac{h_x^7h_yh_z^5}{8,230,118,400}(u^{(6,0,0)}(\mathbf{0}))(u^{(0,0,6)}(\mathbf{0})) + O(H^{14}).
\end{aligned}
\tag{84}
$$

References

[1] S. Adjerid, I. Babuška, and J.E. Flaherty, *A posteriori* error estimation for the finite element method-of-lines solution of parabolic problems, *Math. Model. Meth. Appl. Sci.* 9, (1999), 261-286.

[2] S. Adjerid, B. Belguendouz, and J.E. Flaherty, *A posteriori* error estimation for diffusion systems, *SIAM J. Sci. Comput.* 21, (1999), 728-746.

[3] M. Ainsworth, The influence and selection of subspaces for *a posteriori* error estimators, *Numer. Math.* 73, (1996), 399-418.

[4] M. Ainsworth and J.T. Oden, *A Posteriori Error Estimation in Finite Element Analysis*, Wiley/Interscience, New York, 2000.

[5] I. Babuška, T. Strouboulis, C.S. Upadhyay and S.K. Gangaraj, A posteriori estimation and adaptive control of the pollution error in the *h*-version of the finite element method, *Intern. J. Numer. Meth. Engrg.* 38, (1995), 4207-4235.

[6] I. Babuška, J.E. Flaherty, W.D. Henshaw, J.E. Hopcroft, J.E. Oliger and T. Tezduyar, Eds., *Modeling, Mesh Generation, and Adaptive Numerical Methods for Partial Differential Equations*, Springer-Verlag, New York, 1995.

[7] I. Babuška and W.C. Rheinboldt, Error estimates for adaptive finite element computations, *SIAM J. Numer. Anal.* 15, (1978), 736-754.

[8] I. Babuška and W.C. Rheinboldt, Reliable error estimation and mesh adaptation for the finite element method, in *Computational Methods in Nonlinear Mechanics*, North-Holland, New York, (1980), 67-108.

[9] R.E. Bank, The efficient implementation of local mesh refinement algorithms, in *Adaptive Computational Methods for Partial Differential Equations*, Eds., I. Babuska, J. Chandra, and J.E. Flaherty, SIAM, Philadelphia, (1983), 74-81.

[10] S.C. Brenner and L. Ridgway Scott, *The Mathematical Theory of Finite Element Methods*, Springer-Verlag, New York, 1994.

[11] B. Cockburn, G.E. Karniadakis, C.-W. Shu, Eds., *Discontinuous Galerkin Methods*, Springer-Verlag, Berlin, 2000.

[12] J.E. Flaherty and P.K. Moore, Integrated space-time adaptive *hp*-refinement methods for parabolic systems, *Appl. Numer. Math.* 16, (1995), 317-341.

[13] G.E. Karniadakis and S.J. Sherwin, *Spectral/hp Element Methods for CFD*, Oxford University Press, New York, 1999.

[14] R.B. Lehoucq, D.C. Sorensen and C. Yang, *ARPACK User's Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM, Philadelphia, 1998.
[15] P.K. Moore, Finite difference methods and spatial *a posteriori* error estimates for solving parabolic equations in three space dimensions on grids with irregular nodes, *SIAM J. Numer. Anal.* 36, (1999), 1044-1064.
[16] P.K. Moore, An adaptive finite element method for parabolic differential systems: some algorithmic considerations in solving in three space dimensions, *SIAM J. Sci. Comput.* 21, (2000), 1567-1586.
[17] P.K. Moore, Interpolation error-based *a posteriori* error estimation in three dimensions: a first step, *SMU Tech. Report* 01-08, (2001), 1-33.
[18] P.K. Moore, Applications of Lobatto polynomials to an adaptive finite element method: *a posteriori* error estimates for hp-adaptivity and grid-to-grid interpolation. *Numer. Math.* 94, (2003), 367-401.
[19] P.K. Moore, Implicit interpolation error-based error estimation for reaction-diffusion equations in two space dimensions, *Comp. Meth. Appl. Mech. Engrg.* 192, (2003), 4379-4401.
[20] P.K. Moore, An incomplete assembly with thresholding algorithm for systems of reaction-diffusion equations in three space dimensions: IAT for reaction-diffusion systems, *J. Comp. Phys.* 189, (2003), 130-158.
[21] P.K. Moore, An hb-refinement strategy for solving reaction-diffusion systems in three space dimensions, in preparation.
[22] Y. Saad, ILUT: A dual threshold incomplete LU factorization, *Numer. Linear Algebra Appl.* 1, (1994), 387-402.
[23] Y. Saad and M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Stat. Comput.*, 7 (1986), 856-869.
[24] B. Szabó and I. Babuška, *Finite Element Analysis*, Wiley/Interscience, New York, 1991.
[25] R. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, Wiley-Teubner, Chichester, 1996.
[26] R. Wait and A.R. Mitchell, *Finite Element Analysis and Applications*, John Wiley & Sons, Chichester, 1985.
[27] A. Weiser, *Local-mesh, local-order, adaptive finite element methods with a posteriori error estimators for elliptic partial differential equations*, Tech. Rep. 213, Dept. Computer Science, Yale University, New Haven, (1981).
[28] D. Yu, Asymptotically exact a-posteriori error estimator for elements of bi-even degree, *Mathematica Numerica Sinica* 19, (1991), 89-101.
[29] A. Ženíšek, *Nonlinear Elliptic and Evolution Problems and Their Finite Element Approximations*, Academic Press, London, 1990.
[30] O.C. Zienkiewicz and R.L. Taylor, *The Finite Element Method, Volume 1 The Basis, Fifth Edition*, Butterworth-Heinemann, Oxford, 2000.

Department of Mathematics, Southern Methodist University, Dallas, TX 75275, USA
E-mail: pmoore@smu.edu
URL: http://faculty.smu.edu/pmoore