Integration by Table Look-Up for p-version Finite Elements on Curved Tetrahedra

Saikat Dey[*], Joseph E. Flaherty,
Toshiro K. Ohsumi[†], and Mark S. Shephard

Scientific Computation Research Center
Rensselaer Polytechnic Institute, Troy, NY 12180–3590

Abstract

We develop a method of evaluating element level matrices and vectors associated with p-version finite element computations on tetrahedral meshes with curved boundaries. The procedure uses optimal interpolation formula [7, 8] for non-polynomial portions of integrands followed by exact integration. Exact integrals of product combinations of polynomial interpolants and shape functions are stored so that different integrals may be efficiently evaluated by a table look up.

We present error-analysis for our scheme applied to curvilinear tetrahedral meshes with exact geometric representation of the boundary of the domain as defined in a geometric-modelling system. Numerical results are presented for problems involving the Helmholtz equation.

*Corresponding author: SFA Inc./NRL, 2200 Defense Highway, Suite 405, Crofton, MD, 21114.

†Currently with Colgate University, Hamilton, NY 13346.

1 Introduction

Efficient implementation of p- and hp-refinement finite element methods on serial and parallel computers has been the focus of recent research [11, 12, 17, 26, 27]. Issues related to element-level computation on curved domains involve

1. the choice of variable-order shape functions,
2. the construction of geometric approximations on large elements, and
3. the efficient evaluation of integrals appearing in element matrices and vectors.

Dey et al. [13, 14, 29] address the first two issues and present a framework for specifying and evaluating variable-order shape functions on conforming, unstructured meshes containing mixed topology elements. In part, they also discuss numerical integration [14].

Traditional Gaussian integration techniques [1] on tetrahedral elements are unavailable for integration orders beyond eight [9]. Although higher order integration methods exploiting tensor products are available [15, 19, 30], they are not efficient when used with curvilinear elements or elements not admitting tensor-product shape functions. If the integrand is a polynomial then the necessary integrals can be precomputed and stored. This idea has been used with hierarchical finite elements for quadratic functionals on two-dimensional domains [28]. Precomputed matrices for triangular elements for second-order elliptic partial differential equations on planar domains are also available [31]. Atkins and Shu [2] used this approach with the discontinuous Galerkin method for hyperbolic equations. Symbolic computation employs a similar idea [25, 24, 35].

In curvilinear domains, the integrands are not polynomials even when coefficients and bases are; hence, the precomputed integral tables cannot be applied directly to the entire integrand. Here, we develop integral tables for three-dimensional curvilinear finite elements associated with p-refinement by interpolating non-polynomial parts of the integrand by polynomials to an order sufficient to ensure the optimal convergence rate of the finite element solution and evaluating the resulting element level polynomial integrals exactly using precomputed tables. Symmetries in the polynomial integral entries are identified using the concept of parametric coordinate permutations [13] to reduce the number of unique nonzero entries that require precomputation and storage.

The discretization error of this procedure may be reduced by a proper choice of the interpolation scheme. Chen and Babuška [7, 8] construct an interplant on tetrahedra where the error constant has a linear growth with polynomial degree. The points are symmetrically disposed on tetrahedra and are the minimal number necessary to interpolate a complete polynomial of a given degree.

The rest of this paper is organised as follows: Several methods of integration are reviewed (§2). They all exhibit some deficiencies that are addressed by the table look-up method (§3). Next, techniques for optimizing the implementation of the table look-up method are presented (§4) followed by theoretical bounds on integration error (§5). Finally, a numerical example comparing common methods and the table look-up method is given (§6).

2 Elemental Integrals and Their Approximation

A typical contribution to an element-level matrix, denoted by $\mathbf{K}^e$, of a second-order linear partial differential equation has the form [20]

$$K_{ij}^e = \int_{\Omega_e} \frac{\partial^\alpha N_i}{\partial x_k^\alpha} c(\mathbf{x}) \frac{\partial^\beta N_j}{\partial x_\ell^\beta} d|\mathbf{x}|, \qquad \alpha, \beta = 0, 1, \qquad k, l = 1, 2, 3, \tag{1}$$

where N_i is a shape function, $\mathbf{x} = [x_1, x_2, x_3]^T$ is a position vector, $d|\mathbf{x}|$ is a volume infinitesimal, $c(\mathbf{x})$ is a coefficient of the differential system, and Ω_e is the domain of element e. With $\alpha = \beta = 1$, the entries (1) correspond to those of a typical element stiffness matrix $\mathbf{K}^e$; $\alpha = \beta = 0$ yields entries of an element mass matrix $\mathbf{M}^e$; $\alpha = 0$, $\beta = 1$ yields entries of a convection matrix $\mathbf{C}^e$; and $\alpha = 0$ and omission of the second shape function yields terms of a load vector $\mathbf{f}^e$. Indeed, integrals arising from natural boundary conditions also have this form on domains Γ_e which is the boundary of Ω_e [20].

Transforming, as usual [5, 20, 32], (1) to an integral on a canonical element T yields an integral of the form

$$I = \int_T \Theta(\boldsymbol{\xi}) \Upsilon(\boldsymbol{\xi}) d|\boldsymbol{\xi}| \tag{2a}$$

where

$$\Theta(\boldsymbol{\xi}) = \frac{\partial^\alpha N_i}{\partial \xi_m^\alpha} \frac{\partial^\beta N_j}{\partial \xi_n^\beta}, \qquad \Upsilon(\boldsymbol{\xi}) = c(\mathbf{x}(\boldsymbol{\xi})) \frac{\partial^\alpha \xi_m}{\partial x_k^\alpha} \frac{\partial^\alpha \xi_n}{\partial x_\ell^\beta} J(\boldsymbol{\xi}), \qquad \alpha, \beta = 0, 1, \quad m, n, k, l = 1, 2, 3, \tag{2b}$$

and $\mathbf{x}(\boldsymbol{\xi})$ is a mapping of Ω_e in physical $\mathbf{x}$ space to T in computational $\boldsymbol{\xi}$ space having the Jacobian

$$J(\boldsymbol{\xi}) = \det \begin{bmatrix} \frac{\partial x_1}{\partial \xi_1} & \cdots & \frac{\partial x_1}{\partial \xi_3} \\ \vdots & \ddots & \vdots \\ \frac{\partial x_3}{\partial \xi_1} & \cdots & \frac{\partial x_3}{\partial \xi_3} \end{bmatrix}. \tag{3}$$

The function $\Theta(\boldsymbol{\xi})$ is a polynomial with polynomial shape functions and, hence, is easily integrated. The metrics $\partial \xi_m / \partial x_k$ are functions of $1/J(\boldsymbol{\xi})$ and are generally not polynomials even when $\mathbf{x}(\boldsymbol{\xi})$ is polynomial. An exception occurs with linear transformations where J is constant. Even in this case, $c(\mathbf{x}(\boldsymbol{\xi}))$ need not be a polynomial; therefore, $\Upsilon(\boldsymbol{\xi})$ is generally not a polynomial and, typically, will not be integrable by symbolic means.

Without exact integration, (2a) is approximated in a manner consistent with the accuracy of the finite element interpolation of the exact solution of the partial differential system. If the basis contains complete polynomials of at most degree p, then $\Theta(\boldsymbol{\xi})$ is a complete polynomial of at most degree $2p$.

The complexity of quadrature procedures is appraised in terms of the number of times the integrand is evaluated. For the purposes of computing and comparing the complexity of various methods, we assume the integrand must be calculated to order ϱ accuracy, i.e., it must be exact for all polynomial integrands of order ϱ or less.

2.1 Symmetric Gaussian Quadrature

Symmetric Gaussian quadrature [10] provides an approximation of (2a) having the form

$$I \approx \sum_{\nu=1}^{n_s(\varrho)} w_\nu \Theta(\boldsymbol{\xi}^{(\nu)}) \Upsilon(\boldsymbol{\xi}^{(\nu)}) \tag{4}$$

where w_ν and $\boldsymbol{\xi}^{(\nu)}$, $\nu = 1, 2, \ldots, n_s(\varrho)$, are the Christoffel weights and evaluation points, respectively, and $n_s(\varrho)$ is the number of evaluation points needed for order ϱ. The evaluation points are symmetrically placed within T.

Symmetric Gaussian quadrature has been the preferred integration rule for use with finite element methods since rules with a (nearly) minimal number of function evaluations for a given order are known for low orders [9]. The complexity of an evaluation is $O(\varrho^3)$ [10, 22] for a method of order ϱ. Unfortunately, the points and weights are only known to order eight for integration on tetrahedra [9].

2.2 Generalized Product Rule (GPR) Quadrature

By Mapping T to a $2 \times 2 \times 2$ cube, (2a) can be expressed as a product of three one-dimensional integrals [13]. Using Gaussian quadrature, the complexity in each coordinate direction is $O(\varrho)$; thus, the total complexity is $O(\varrho^3)$. This method of integration is known as a *Generalized Product Rule* (GPR) [10] and is preferred to the symmetric product rules of Grundmann and Möller [18] since the GPR weights are positive [23]; thus, accuracy is not lost due to round-off error. Since neither the shape nor the symmetry of the tetrahedral element is used, GPR integration will use more points than symmetric Gaussian quadrature for a given order; however, the rules are available for all ϱ.

2.3 Sum Factorization and Vector Quadrature

If $\Theta(\boldsymbol{\xi})\Upsilon(\boldsymbol{\xi})$ can be factored as $F_1(\xi_1)F_2(\xi_2)F_3(\xi_3)$ then (2a) is evaluated in $O(\varrho)$ operations as [15]

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} F_1(\xi_1)F_2(\xi_2)F_3(\xi_3)d\xi_1 d\xi_2 d\xi_3 = \int_{-1}^{1} F_1 d\xi_1 \int_{-1}^{1} F_2 d\xi_2 \int_{-1}^{1} F_3 d\xi_3. \tag{5}$$

This method of quadrature is known as sum factorization.

However, for curved domains, $\Theta(\boldsymbol{\xi})\Upsilon(\boldsymbol{\xi})$ cannot be factored exactly. Therefore, $\Theta(\boldsymbol{\xi})\Upsilon(\boldsymbol{\xi})$ must be interpolated to a tensor-product form, requiring $O(\varrho^3)$ operations. A serious difficulty in interpolating the integrand in this manner is that the basis is not complete for order ϱ. Thus, if F_ν, $\nu = 1, 2, 3$, are polynomials whose degrees sum to ϱ or less, then the functional factorization cannot represent all polynomials of three variables of order ϱ or less. For example, the function $1 + \xi_2 + 2\xi_3 + 3\xi_2\xi_3$ cannot be exactly represented as a prodcut $F_1(\xi_1)F_2(\xi_2)F(\xi_3)$ of polynomials. Thus, a higher-order interpolation would be needed to achieve the same order of accuracy as Gaussian quadrature, making sum factorization inefficient.

Suppose that $\Theta(\boldsymbol{\xi})$ and $\Upsilon(\boldsymbol{\xi})$ are approximated as

$$\Theta(\boldsymbol{\xi}) \approx \sum_{\nu=1}^{n_v(\varrho)} a_\nu \varphi_\nu(\boldsymbol{\xi}), \qquad \Upsilon(\boldsymbol{\xi}) \approx \sum_{\nu=1}^{n_v(\varrho)} b_\nu \varphi_\nu(\boldsymbol{\xi}) \tag{6}$$

with a basis $\{\varphi_\nu(\boldsymbol{\xi})\}_{\nu=1}^{n_v(\varrho)}$ that is orthogonal in $L^2(T)$. Then (2a) is approximated as

$$I \approx \sum_{\nu=1}^{n_v(\varrho)} a_\nu b_\nu. \tag{7}$$

This method is called vector factorization. Evaluation of the sum and interpolation has $O(\varrho^3)$ complexity. In comparison, the table look-up method (§3), will require only the interpolation of Υ, thus, is less expensive than vector quadrature.

Thus, sum factorization and vector quadrature both require the solution of an interpolation problem on curved domains. Due to the cited inefficiencies, these quadrature methods are not considered further.

3 Integration by Table Look-Up

As an alternative to the quadrature techniques of §2, we describe a table look-up procedure where the non-polynomial portion Υ of (2a) is approximated by the polynomial of degree q

$$\Upsilon(\boldsymbol{\xi}) \approx \Upsilon^*(\boldsymbol{\xi}) = \sum_{\nu=1}^{n_t(q)} a_\nu \phi_\nu(\boldsymbol{\xi}), \tag{8a}$$

where

$$n_t(q) = \frac{(q+1)(q+2)(q+3)}{6} \tag{8b}$$

for a complete polynomial of degree q, and the result is integrated exactly to yield

$$I \approx \int_T \Theta(\boldsymbol{\xi})\Upsilon^*(\boldsymbol{\xi})\, d|\boldsymbol{\xi}| = \sum_{\nu=1}^{n_t(q)} a_\nu \int_T \Theta(\boldsymbol{\xi})\phi_\nu(\boldsymbol{\xi})\, d|\boldsymbol{\xi}|. \tag{9}$$

While any basis may be used in (8a), it seems expedient to use $\phi_\nu = N_\nu$, $\nu = 1, 2, \ldots, n_t(q)$. If the coefficients a_ν, $\nu = 1, 2, \ldots, n_t(q)$, are to be determined by interpolation, then Chen and Babuška [7, 8] provide a "nearly optimal" set of points which limit the growth of the constant of the interpolation error on tetrahedra to $O(q)$. The points are symmetrically disposed on T and are the minimal number to interpolate a polynomial of degree q. Since Υ depends only on the partial differential system and the mesh geometry (cf. (2b)), it requires only one interpolation for all combinations of i and j in (2b).

The determination of a_ν, $\nu = 1, 2, \ldots, n_t(q)$, in (8a) by interpolation requires the solution of the linear algebraic system

$$\mathbf{\Phi a} = \mathbf{d} \tag{10a}$$

where

$$\Phi_{mn} = N_n(\boldsymbol{\xi}^{(m)}), \qquad d_m = \Upsilon(\boldsymbol{\xi}^{(m)}), \qquad m, n = 1, 2, \ldots, n_t(q), \tag{10b}$$

with $\boldsymbol{\xi}^{(m)}$, $m = 1, 2, \ldots n_t(q)$, being the Chen and Babuška [7, 8] interpolation points. The algebraic complexity of solving (10) depends on the finite element basis. Use of a Lagrangian basis [20] renders it diagonal. The situation is more complex with the hierarchical bases commonly used for p-refinement [6, 33]. Elements of the hierarchical basis are associated with a specific mesh entity (region, face, edge, or vertex) and they vanish, to maintain continuity, on the boundary of all regions (finite elements) not containing that entity. The Chen and Babuška interpolation points are also on vertices, edges, faces, and element interiors. Thus, an interpolation at a vertex will only involve the (linear) shape function that is identified with that vertex. All other shape functions vanish there, and the coefficient in (10) that is associated with the vertex will be explicitly determined. Likewise, when interpolation occurs at a point on an edge, the only nontrivial coefficients of (10) are those identified with the edge and the two vertices bounding the edge. With the vertex coefficients determined, the edge coefficients may determined by block forward substitution. Determination of coefficients associated with faces and finite elements follow in similar fashion. Thus, (10) may be put in a block lower triangular form with coefficients and interpolation points ordered and determined by vertices, edges, faces, and regions. Unfortunately, there are $O(q^3)$ points and unknowns within element interiors, so this ordering leaves a large block to be determined when q is large. Assuming a direct solution procedure saving the factorization of $\mathbf{\Phi}$, the algebraic complexity of the block forward substitution would still be $O(q^6)$, as it would be for an arbitrary ordering. In determining the computational cost, we assume that integrand evaluations incur a much larger expense than algebraic operations; thus, we neglect the $O(q^6)$ cost involved with forward substitution. Moreover, since the interpolation is split over mesh entities, the coefficient bounding the cost is small ($< 1/6$); thus, for moderate q's used in practical problems, neglecting the forward substitution cost is justified.

In addition to this algebraic cost, there is also the complexity associated with evaluating Υ in (10) at $O(q^3)$ points (cf. (8b)). Other costs associated with accessing memory to retrieve the value of an integral (2a) for a particular choice of α, β, etc. are assumed to be negligible. We show (§5) that $q \geq p$ is required to maintain the optimal $O(h^p)$ convergence rate of the finite element method with exact integration; thus, the complexity of table look-up has the same $O(p^3)$ complexity associated with other quadrature methods (§2). If the integrand were polynomial, table look-up would be exact with an $O(1)$ complexity.

The table look-up method has several advantages:

1. The topology-dependent basis and interpolation points use fewer evaluations than, *e.g.*, product formulations.

2. A hierarchical basis with the Chen and Babuška [7, 8] interpolation points renders the interpolation problem (10) block lower triangular, and provides computational savings for moderate values of q.

3. The method presents a general approach to high-order integration on geometrically complex regions. The basis need not satisfy any special properties.

4. When $\Upsilon(\boldsymbol{\xi})$ is constant, table look-up provides exact integration in $O(1)$ time.

5. Table look-up may be used with a non-polynomial but integrable basis [3].

Relative to other quadrature methods, table look-up requires the solution of an interpolation problem on each finite element; however, this problem must only be solved once for each differential equation and mesh.

4 Implementation and Optimization of Table Look-Up

The integration table contains values of

$$\int_T \frac{\partial^\alpha N_i(\boldsymbol{\xi})}{\partial \xi_m^\alpha} \frac{\partial^\beta N_j(\boldsymbol{\xi})}{\partial \xi_n^\beta} N_k(\boldsymbol{\xi})\, d|\boldsymbol{\xi}|. \tag{11}$$

With shape functions of degree p and interpolation of degree q, the tables for $\mathbf{K}^e$ or $\mathbf{M}^e$ would contain $O(p^6 q^3)$ entries for second-order, three-dimensional problems. Various symmetry and orthogonality considerations can reduce the size of these tables, and we describe an approach based on the shape functions of Carnevali *et al.* [6].

Repetitive zero and nonzero entries in the tables need not be stored. Instead, memory may be reduced by using integers to point to the unique (double-precision) floating point entries. This is done by using two arrays: one array holds unique floating point results of the integrals in a *result table*, and a second *index table* stores integers pointing to the result. A mapping converts the indices $i, j, k, m, n, \alpha, \beta$ of an integral (11) to a unique entry in the index table which gives the appropriate answer in the result table. The tables are represented as one-dimensional arrays to reduce the number of pointers. Entries in the result table were generated using the symbolic integration system PARI [4], which manipulates polynomials in factors of 5 to 100 times faster than interpreted symbolic packages, such as Maple [16].

A reduction of the dimension of the index table is obtained from the symmetry of integration under cyclic permutation of the parametric coordinates, since integration is invariant under coordinate rotations. A further reduction in size is obtained by exploiting the symmetry or anti-symmetry of the shape functions under a permutation of variables. For example, for shape functions used here [6], parameter permutations of the edge modes $N_i(\xi, \eta)$, are related by $N_i(\xi, \eta) = s(N_i) \times N_i(\eta, \xi)$ where $s(N_i)$ is 1 if the degree of N_i is even and -1 otherwise.

Table 1 shows the reduction in size of the index table by exploiting symmetries present in the shape functions of Carnevali *et al.* [6] with $p \leq 6$. The asymptotic entries in Table 1 assume that the elemental matrices and vectors are dense and asymmetric. The shape functions of Carnevali *et al.* [6] also satisfy an orthogonality condition when the difference in degrees of the product of two shape function derivatives in $\mathbf{K}^e$ exceeds four. This sparsity,

	$\mathbf{K}^e$		$\mathbf{M}^e$		$\mathbf{f}^e$	
	Asymp.	Compact	Asymp.	Compact	Asymp.	Compact
Entries (Millions)	0.5	0.05	0.5	0.04	0.007	0.0005
Size (MB)	4	0.4	4	0.32	0.06	0.004

Table 1: Unique nonzero precomputed entries for $p = 6$ compared to asymptotic estimates for full dense matrices and vectors.

however, has not been exploited. Furthermore, if the element matrices are symmetric, this allows a further reduction of the number of index entries. A description of the asymptotic size of the precomputed coefficients and various of parametric permutations is given in [13].

5 Integration Error Estimates

We examine the errors due to approximate integration for a Galerkin problem of the form: find $u \in H_0^1$ such that

$$A(v, u) = (v, f), \qquad \forall v \in H_0^1, \tag{12}$$

where $A(v, u)$ is a bilinear form satisfying the conditions of continuity and coercivity on H^1

$$|A(v, w)| \leq C_1 \|v\|_{1,\Omega} \|w\|_{1,\Omega}, \qquad A(v, v) \geq C_2 \|v\|_{1,\Omega}^2, \qquad \forall v, w \in H_0^1(\Omega). \tag{13}$$

The problem domain Ω satisfies

$$\Omega = \cup_{e=1}^{N_\Delta} \Omega_e$$

where N_Δ is the number of finite elements. The linear form (v, f) denotes the usual L^2 inner product, functions in the Sobolev space $H^k(\Omega)$, $k \geq 0$, have square integrable derivatives through order k on Ω, $\|\cdot\|_{k,\Omega}$ denotes a norm on $H^k(\Omega)$, and a subscript 0 on H^1 restricts functions to satisfy trivial Dirichlet data on $\partial\Omega$. We also let $|\cdot|_{k,\Omega}$, $k \geq 1$, denote a seminorm involving derivatives of order k on $H^k(\Omega)$.

With exact integration and exact representation of Ω, a finite element solution $U \in S_0^N \subset H_0^1$ converges as $O(h^p)$ in H^1, where h is the maximum length of an element edge. The software used in §6 employs an exact mapping of Ω_e to the canonical element T [14]; thus, perturbations of this rate are only due to approximate integration. Towards quantifying this effect, let the strain energy and load potential in (12) be approximated as $A_*(v,u)$ and $(v,f)_*$. If $A_*(U,U)$ is coercive, then the finite element solution $U^* \in S^N_{*,0}$ with inexact integration satisfies [34]

$$\begin{aligned}\|u-U^*\|_{1,\Omega} \leq & C \inf_{V\in S_0^N}\left\{\|u-V\|_{1,\Omega} + \sup_{W\in S_0^N}\frac{|A(V,W)-A_*(V,W)|}{\|W\|_{1,\Omega}}\right. \\ & \left. + \sup_{W\in S_0^N}\frac{|(W,f)-(W,f)_*|}{\|W\|_{1,\Omega}}\right\}.\end{aligned} \tag{14}$$

In order to maintain the optimal rate of convergence, the two perturbations in (14) must also converge as $O(h^p)$. A regularity assumption is necessary for our perturbation analysis.

Definition 1. Let $v \in H^s(T)$, $s \geq 0$, then the transformation from T to Ω_e is *regular* if the Jacobian J is bounded and there exists a constant $C > 0$ such that

$$|v|_{s,T} \leq C\left\{\inf_{\boldsymbol{\xi}\in T} J(\boldsymbol{\xi})\right\}^{-\frac{1}{2}} h_e^s \|v\|_{s,\Omega_e} \tag{15}$$

where h_e is the length of the longest edge of Ω_e.

As noted, the Jacobian J of the transformation is not necessarily a polynomial; however, if it is then optimal convergence results are available as indicated in the following theorem.

Theorem 1. *Assume $J(\boldsymbol{\xi})$, $\boldsymbol{\xi} \in T$, is a polynomial of degree q satisfying (15), then the perturbation errors in (14) are $O(h^p)$ when integrals (2a) are evaluated by Gaussian quadrature of order $2p-1+q$.*

Proof. Wait and Mitchell [34] establish the result when $A(v,u)$ corresponds to a Laplacian-like operator. The results may easily be generalized to situations where terms of the form Cvu, $C > 0$, are present in $A(v,u)$. □

For integration by table look up, we concentrate on a single element Ω_e and establish that

$$|A_e(V,W) - A_{*,e}(V,W)| \leq C h_e^p \|V\|_{p+1,\Omega_e}\|W\|_{1,\Omega_e}, \tag{16}$$

where A_e is the portion of A on Ω_e. Having done this, it is clear that the bounds in (14) on Ω will be satisfied. Bounds for the perturbation involving (W, f) proceed along similar lines and will not be presented.

Proposition 1. *Assume $\Upsilon \in C^\infty(T)$. Then*

$$\|\Upsilon - \Upsilon^*\|_{\infty,T} \leq C_1(C_2 + \lambda_q(T))\frac{|T|^{q-2}}{(q/3)!}|\Upsilon|_{q,T}, \qquad q > 2, \tag{17}$$

where $\lambda_q(T)$ is the Lebesgue constant and $|T|$ is the volume of the canonical tetrahedron ($= 1/6$ for a right unit tetrahedron).

Proof. Let $P_q(T)$ be the space of polynomials of degree q or less on T, then [7]

$$\|\Upsilon - \Upsilon^*\|_{\infty,T} \leq (C_2 + \lambda_q(T)) \inf_{\rho \in P_q(T)} \left\{ \sup_{\boldsymbol{\xi} \in T} |\rho(\boldsymbol{\xi}) - \Upsilon(\boldsymbol{\xi})| \right\} \tag{18}$$

Bounding the least deviation by an interpolation error yields (17) [5]. $\square$

Let us apply this result to a single term in the strain energy.

Theorem 2. *Assume (15) holds, interpolation is performed at the points of Chen and Babuška [7], and $\Upsilon \in C^\infty(T)$, then there exists a constant $C_1 > 0$ such that*

$$\left| \int_T \Theta(\Upsilon - \Upsilon^*)\, d|\boldsymbol{\xi}| \right| \leq C_1(C_2+q)\frac{|T|^{q-2}}{(q/3)!}\|V\|_{p+1,\Omega_e}\|W\|_{1,\Omega_e}\|\Upsilon\|_{q,\Omega_e}h_e^q, \qquad q > 2. \tag{19}$$

Proof. Adding the squares of (15) with $s = 0, 1$ yields

$$\|v\|_{1,T} \leq C \left\{ \inf_{\boldsymbol{\xi} \in T} J(\boldsymbol{\xi}) \right\}^{-\frac{1}{2}} \|v\|_{1,\Omega_e}. \tag{20}$$

Let N_i and N_j in (2b) be designated as V and W for consistency with (16) and use the generalized Hölder inequality and (17) to obtain

$$\left| \int_T \Theta(\Upsilon - \Upsilon^*)\, d|\boldsymbol{\xi}| \right| \leq C_1(C_2 + \lambda_q(T))\frac{|T|^{q-2}}{(q/3)!}\|V\|_{1,T}\|W\|_{1,T}|\Upsilon|_{q,T}. \tag{21}$$

Using the regularity conditions (15) and (20) yields

$$\left|\int_T \Theta(\Upsilon - \Upsilon^*)\, d|\boldsymbol{\xi}|\right| \le C_1(C_2 + \lambda_q(T))\frac{|T|^{q-2}}{(q/3)!}\|V\|_{1,\Omega_e}\|W\|_{1,\Omega_e}\|\Upsilon\|_{q,\Omega_e} h_e^q. \tag{22}$$

The Chen and Babuška [7] interpolation points satisfy $\lambda_q(T) = O(q)$. This with replacement of $\|V\|_{1,\Omega_e}$ by $\|V\|_{p+1,\Omega_e}$ gives (19). □

Corollary 1. *Under the conditions of Theorem 2 then (16) is satisfied with* $q = p$, $p > 2$.

Proof. The energy difference $|A_e(V,W) - A_{*,e}(V,W)|$ is a linear combination of terms proportional to the left side of (19). Since the term

$$(C_2 + q)\frac{|T|^{q-2}}{(q/3)!}\|\Upsilon\|_{q,\Omega_e} \le C \tag{23}$$

for fixed q, then (19) is bounded by $O(h_e^q)$ terms. Thus, perturbations due to quadrature are $O(h^p)$ if $q = p$. □

6 Numerical Examples

Our numerical examples involve solutions of the Helmholtz equation

$$\Delta u(\mathbf{x}) + \kappa^2 u(\mathbf{x}) = 0, \qquad \mathbf{x} \in \Omega, \tag{24}$$

subject to the Neumann data

$$\frac{\partial u}{\partial r} = \frac{\partial}{\partial r}\left(\frac{e^{i\kappa r}}{r}\right), \qquad \mathbf{x} \in \partial\Omega, \tag{25}$$

where r is a spherical distance. Problems are solved on various three-dimensional domains Ω having planar and curvilinear boundaries. With the specified boundary conditions, the solution of the boundary value problem (24, 25) is

$$u = \frac{e^{i\kappa r}}{r} \tag{26}$$

regardless of the shape of Ω.

Results are obtained with wave number $\kappa = 3$ by uniform p-refinement of an initial mesh using integration by table look-up, symmetric Gaussian quadrature, and GPR. Results with symmetric Gaussian quadrature are only presented for $p \leq 4$ since higher-order methods are unavailable. The integrals leading to the stiffness and mass matrices for this problem are of the form (1) with $\alpha = \beta$ and $c(\boldsymbol{x})$ constant. The integrals are computed to the minimum order of Gauss quadrature or table look-up interpolation to maintain optimal convergence as discussed in §5. The times to compute $\mathbf{K}^e$ and $\mathbf{M}^e$ are compared for the various quadrature rules. Since the cost of doing the polynomial interpolation of $\Upsilon(\boldsymbol{\xi})$ terms is $O(1)$ for domains with planar boundaries, it demonstrates the savings that can be realized using the table look-up method in elemental computations for elements that do not have any curvilinear boundary entities.

Example 1. We solve (24, 25) with Ω an octant of the region between two concentric cubes centered at the origin and having edge lengths one and two. The mesh describing Ω has 1261 elements, all having planar faces.

In Figures 1 and 2 we present the global error and the computational time as a function of the degrees of freedom (DOF) for computations performed with p ranging from 1 to 5. All integration methods are exact, so the finite element solution errors are decreasing at an exponential rate with p (Figure 1). Results with table look-up (with $q = 0$ for this example) are between five and ten times faster than symmetric Gaussian quadrature and between ten and thirty seven times faster than GPR. Even though the table look-up method has a running time of $O(1)$, the time per element increases with order in Figure 2 since the number of integrals calculated per element increases.

Example 2. We solve (24, 25) with Ω being the upper half of a $\pi/6$ radian sector between two concentric spheres of radius one and three centered at the origin.

From Figure 3, the table look-up method is competitive with the other methods. The lack of an exponential convergence rate at higher orders is explained in the next section.

Results in Figure 4, likewise, indicate the advantages of table look-up at higher orders. Symmetric Gaussian quadrature was faster at $p = 4$ but higher-order formulas are unavailable. For methods which are available at higher order, clearly, the table look-up method is faster. The results show the theory presented in §5 holds even for problems where $A_*(U, U)$ is not coercive. However, we mention that we have obtained similar results [13] for the Poisson equation for which the theory holds.

Note that there is a lack of optimal convergence at higher orders in Figure 3. This can be explained for Gauss quadrature by the theorem

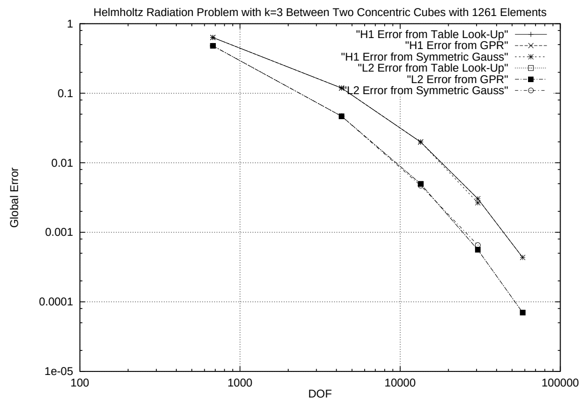

Figure 1: Global error in L^2 and H^1 as a function of DOF for Example 1.

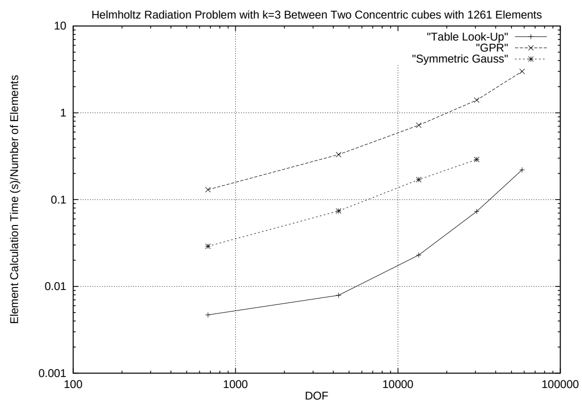

Figure 2: Time to evaluate elemental integrals as a function of DOF for Example 1.

Theorem 3. *Let $I_n(f)$ be the n-th order quadrature interpolant of f in a d-dimensional domain Ω_e. Then*

$$\left| \int_{\Omega} f(x) d\Omega_e - I_n(f) \right| \leq \frac{C}{(n+1)!} \|f\|_{\infty,\Omega_e}, \tag{27}$$

where C contains a factor of $\max \left\{ |\Omega_e| , |\Omega_e|^{n+1} \right\}$.

Proof. cf. [21] □

and for the table look-up method by (23). In both cases, (27) and (23), the error is bounded by the n-th derivative of the integrand or Υ. If the max-

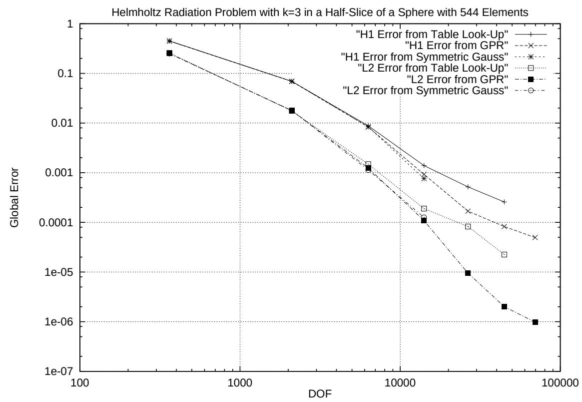

Figure 3: Global error in L^2 and H^1 as a function of DOF for Example 2.

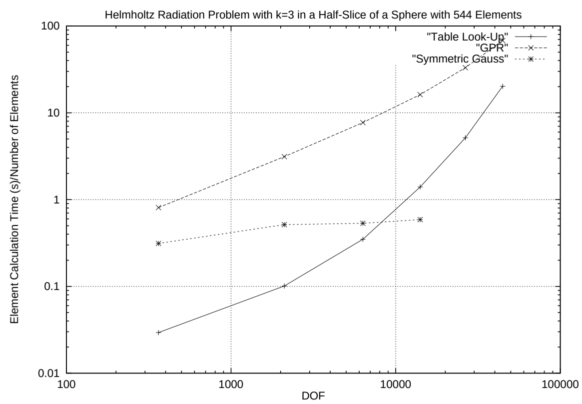

Figure 4: Time to evaluate elemental integrals as a function of DOF for Example 2.

imum of the n-th derivative increases faster than $1/(n+1)!$, then the error does not decrease with increasing integration or interpolation order. Plotting the absolute value of the maximum of $\frac{\partial^n J}{\partial r^n}/(n+1)!$ in Figure 5, we show that the ratio is increasing thus explaining the lack of optimal convergence.

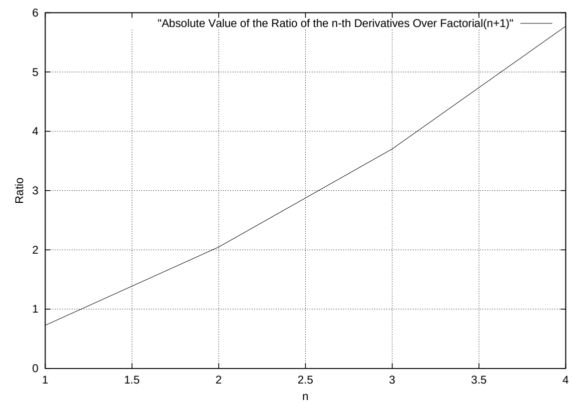

Figure 5: Ratio of n-derivative of the Jacobian over $(n+1)!$

6.1 Analysis of the Numerical Examples

In our two numerical examples, we have shown two cases where the table look-up method compared favorably with traditional methods. The examples with planar elements is the most favorable to the table look-up method which results in the exact value in $O(1)$ time.

What is more impressive is that in the worse-case problems for the table look-up, where the interpolation was of the term which has an exponentially increasing derivative, the results show the table look-up method being competitive with other methods. Since Gauss quadrature rules interpolate the entire integrand, the polynomial which multiplies the non-polynomial term reduces the rate at which the derivative of the interpolant grows.

Thus, the table look-up method, is ideal for cases where the elements are either planar or where $\Upsilon(\boldsymbol{\xi})$'s derivatives grows slowly, e.g. nearly planar elements. Even when the n-th derivatives of $\Upsilon(\boldsymbol{\xi})$'s grow faster than $1/(n+1)!$, the loss of convergence rate experienced for errors already well below limits of practical interest. In meshes with a large number of elements, most of the elements are internal and thus planar. Thus, especially for larger problems, the table look-up method is the clear choice of integration method.

7 Concluding Remarks and Future Work

A general technique for effective numerical computation of the element level integrals was presented. It is based on approximating the non-polynomial part of the integrand in terms of polynomials followed by use of precomputed values of the resulting polynomial. The table look-up method is better suited for use in general curved domains compared to other methods since it is more efficient, more flexible in what is approximated, and has fewer limitations in the maximum order of the method available. The Helmholtz equation showed the validity of these assertions.

The number of the precomputed integral entries required to be stored for various element level integrals can be reduced by investigating additional symmetries in the shape functions and their derivatives if they exist. The efficiency of using polynomial integrand approximation followed by use of precomputed values for element level integral computation needs to be evaluated for three dimensional element topologies besides tetrahedron.

Future work should include attempting the table look-up method on more complex (non-linear) problems, such as the Navier-Stokes equations for high p and many elements. In this paper, only $\Upsilon(\boldsymbol{\xi})$ was interpolated. It would also be worthwhile to consider interpolating $\theta(\boldsymbol{\xi})\Upsilon(\boldsymbol{\xi})$ where $\theta(\boldsymbol{\xi})$ is a factor of $\Theta(\boldsymbol{\xi})$. The choice of $\theta(\boldsymbol{\xi})$ would be problem dependent. The flexibility of the table look-up method provides for a number of research opportunities in exploiting problem knowledge to decrease the expense of numerical integration in the finite element method. For example, it would be worthwhile to investigate using the table look-up method with non-polynomial basis functions where traditional methods are often too computationally expensive.

8 Acknowledgments

The work described in this paper comes from the first authors doctoral research which was sponsored in part by ONR and NRL.

References

[1] M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions.* Dover Publications Inc., 1972.

[2] H. L. Atkins and C.-W. Shu. Quadrature-free implementation of discontinuous galerkin method for hyperbolic equations. *AIAA J.*, 36:775–782, 1998.

[3] I. Babuška, B. Andersson, B. Guo, J. M. Melenk, and H.-S. Oh. Finite element method for solving problems with singular solutions. *J. Comput. Appl. Math.*, 74:51–70, 1996.

[4] C. Batut, D. Bernardi, H. Cohen, and M. Olivier. *PARI*. Laboratory A2X, University of Bordeaux I, Cedex, FRANCE, 1997.

[5] S. C. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Springer-Verlag, 1994.

[6] P. Carnevali, R. B. Morris, Y. Tsuji, and G. Taylor. New basis functions and computational procedures for p-version finite element analysis. *I. J. Numer. Meths. Eng.*, 36:3759–3779, 1993.

[7] Q. Chen and I. Babuška. Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Comp. Meth. Appl. Mech. Engng.*, 128:405–417, 1995.

[8] Q. Chen and I. Babuška. Approximate optimal polynomial interpolation points in the tetrahedron. preprint, 1995.

[9] R. Cools and P. Rabinowitz. Monomial cubature since 'Stroud': a compilation. *J. Comput. Appl. Math.*, 48:309–326, 1993.

[10] P. J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic Press, Inc., second edition, 1984.

[11] L. Demkowicz, J. T. Oden, W. Rachowicz, and O. Hardy. Toward a universal h-p adaptive finite element strategy, Part 1: Constrained approximation and data structure. *Comp. Meth. Appl. Mech. Engng.*, 77:79–112, 1989.

[12] K. D. Devine and J. E. Flaherty. A parallel adaptive hp-refinement techniques for conservation laws. *Appl. Num. Math.*, 6:1–20, 1996.

[13] S. Dey. *Geometry-based three dimensional hp finite element modeling and computations*. PhD thesis, Civil Engineering, Rensselaer Polytechnic Institute, 1997.

[14] S. Dey, M. S. Shephard, and J. E. Flaherty. Geometric representation issues associated with p-version finite element computations. *Comp. Meth. Appl. Mech. Engng.*, 150:39–55, 1997.

[15] M. Dubiner. Spectral methods on triangles and other domains. *J. Sci. Comp.*, 6:345–390, 1991.

[16] M. B. Monagan *el al.* *Maple V.* Waterloo Maple Inc., 1998.

[17] J. Fish and R. Guttal. Recent advances in the p-version of the finite element method for shells. *Computing Systems in Engineering*, 6:195–211, 1995.

[18] A. Grundmann and H. M. Möller. Invariant integration formulas for the n-simplex by combinatorial methods. *SIAM J. Numer. Anal.*, 15:282–290, 1978.

[19] H. E. Hinnat. A fast method of numerical quadrature for p-version finite element matrices. *Int. J. Numer. Mech. Engng.*, 37:3723–3750, 1994.

[20] T. J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis.* Prentice Hall, 1987.

[21] E. Isaacson and H. B. Keller. *Analysis of Numerical Methods.* Dover Publications, Inc., 1994.

[22] Y. Jinyun. Symmetric gaussian quadrature formulae for tetrahedral regions. *Comp. Meth. Appl. Mech. Engng.*, 43:349–353, 1984.

[23] A. R. Krommer and C. W. Ueberhuber. *Computational Integration.* SIAM, 1998.

[24] A. K. Noor and C. M. Andersen. Computerized symbolic manipulation in nonlinear finite element analysis. *Computer and Structures*, 13:379–403, 1979.

[25] A. K. Noor and C. M. Andersen. Computerized symbolic manipulation in structural mechanics-progress and potential. *Computer and Structures*, 10:95–118, 1979.

[26] J. T. Oden. Parallel adaptive hp-finite element methods for problems in fluid and solid mechanics. In T. J. R. Hughes, E. Onate, and O. C. Zienkiewicz, editors, *Recent developments in finite element analysis*, pages 29–35, 1994.

[27] W. Rachowicz, J. T. Oden, and L. Demkowicz. Toward a universal h-p adaptive finite element strategy, part 3, design of h-p meshes. *Comp. Meth. Appl. Mech. Engng.*, 77:181–212, 1989.

[28] M. P. Rossow and I. N. Katz. Hierarchical finite elements and precomputed arrays. *Int. J. Numer. Meth. Engng.*, 12:997–999, 1978.

[29] M. S. Shepard, S. Dey, and J. E. Flaherty. A straightforward structure to construct shape functions for variable p-order meshes. *Comp. Meth. Appl. Mech. Engng.*, 147:209–233, 1997.

[30] S. J. Sherwin and G. E. Karniadakis. A new triangular and tetrahedral basis for high-order (hp) finite element methods. *Int. J. Numer. Meth. Engng.*, 38:3775–3802, 1995.

[31] P. Silvester. Construction of triangular finite element universal matrices. *Int. J. Numer. Meth. Engng.*, 12:237–244, 1978.

[32] G. Strang and G. J. Fix. *An Analysis of the Finite Element Method.* Prentice-Hall, 1973.

[33] B. A. Szabo and I. Babuška. *Finite Element Analysis.* John Wiley & Sons, Inc., 1991.

[34] R. Wait and A. R. Mitchell. *Finite Element Analysis and Applications.* John Wiley & Sons Ltd., 1985.

[35] G. Yagawa, G.-W. Ye, and S. Yoshimura. A numerical integration scheme for finite element method based on symbolic manipulation. *Int. J. Numer. Meth. Engng.*, 29:1539–1549, 1990.