2 Contents

2 Piecewise Functions

The description of fields by piecewise polynomials is used extensively in finite element analysis and computer graphics. Indeed the two areas overlap principally because they both attempt to describe continuous quantities using finite linear spaces. However, despite the potential benefits of the methods used in one discipline for the other, there appears to have been little cross fertilisation. In this paper we present a formalism applicable to both areas which seeks to place them on a common footing.

First we formalise the concept of field description by piecewise parametric functions, extending the treatment of Oden (1972). A primary requirement is that the field must be linear in the ensemble or global parameters.

Secondly we discuss alternative basis representations at the element level. We restrict our attention to polynomial bases and consider requirements for the conversion from one basis to another. It is noted that where a conversion is possible the mapping may be described by a finite dimensional linear transformation of the element parameters. Such linear mappings have been considered in the context of computer graphics by Timmer (1980) and Lewis (1981). We also discuss the relative efficiency and degree of conditioning of field evaluations with the different basis representations.

Next we consider the constraints required for interelement continuity of fields. In finite element analysis the element basis functions are usually formulated in a way which avoids the need for explicit continuity constraints across element boundaries. Moreover, each element parameter is identified with a unique ensemble or global parameter. To avoid these rather restrictive assumptions we introduce continuity constraints as linear relations between element parameters of contiguous elements. These conditions have been discussed previously with reference to computer graphics by Barsky and Beatty (1983) for cubic splines and Kahmann (1982) for Bézier patches. The imposition of such constraints may involve linear transformations of element coordinates and bases in order to equate element parameters of adjacent elements.

In the following section we introduce a new formalism which allows for general linear mappings between the ensemble field parameters and the element field parameters. This linear mapping contrasts with Oden's (1972) formalism which allows only Boolean maps. Since this mapping is linear it can now accommodate both basis transformations and continuity conditions. We are thus free to choose an element basis from a wide class of functions. The general linear mapping can be viewed as shifting information content from the element basis functions to the ensemble to element parameter map. As a result of this considerable savings may be made in field evaluations at the element level. Continuity of field tangents across interelement boundaries may be specified without requiring continuity of element coordinate density if we relax the Boolean map concept to allow real valued coefficients. In this case there is still a unique ensemble parameter associated with each element parameter. This idea is illustrated using one dimensional C^1 cubic Hermite bases. It will be argued, given the freedom to choose any polynomial element basis, that the monomial basis form is a natural candidate where efficiency and robustness of evaluation are important issues. A new formulation of the one dimensional non-uniform quadratic spline expressed with respect to the monomial basis is provided as an illustration of the power of the general linear ensemble to element parameter map.

Next we consider the ensemble basis function. The relationship between the degree of the basis, the interelement continuity, the number of ensemble parameters used to define the field, and the support associated with each ensemble basis function is discussed.

Finally we discuss how finite element equations may be expressed using the new formalism and we illustrate this with the Poisson equation. The use of monomial bases is investigated with attention being paid to savings possible in the evaluation of outer products. In finite element analysis we show that if only the monomial form for the element basis functions is used savings can be made in the storage requirements for basis functions evaluated at quadrature points.

2.1 Definitions

We consider the method by which a field **U** is defined over a region Ω^E by a finite number of piecewise functions. Much of the notation is based upon Oden (1972) and Oden and Reddy (1976). Throughout this thesis the Einstein summation convention is used: where repeated indices appear within a product a summation is implied over their range.

(1) The region of interest Ω is approximated by another region Ω^E which is partitioned into a finite number of disconnected open subregions ω^ε called elements. It is desirable that Ω^E should approximate Ω as closely as possible since the field will be defined only on Ω^E and its boundary.

(2) The collection of element closures $\overline{\omega}^\varepsilon$ need not be disconnected since contiguous elements will share portions of their boundaries. Thus $\{\overline{\omega}^\varepsilon\}$ is a covering of Ω^E while $\{\omega^\varepsilon\}$ is not.

(3) The union of element closures defines the region over which the field is specified and is called the ensemble, $\Omega^E = \cup_\varepsilon \overline{\omega}^\varepsilon$.

(4) Associated with each element of dimension L is a homeomorphism ξ_ε which maps ω^ε onto an open space in $\mathbb{R}^L$. The coordinates $(x^1,x^2,...x^L)$ of the image $\xi_\varepsilon(\mathbf{x}) \in \mathbb{R}^L$ of the point $\mathbf{x} \in \omega^\varepsilon$ are called the element coordinates of **x**.

(5) Within each element ε the field $\mathbf{u}_{(\varepsilon)}$ is characterised by a finite number of independent parameters $\mathbf{u}^{\beta(\varepsilon)}$ called element field parameters.

(6) The contribution of an element field parameter to the field at a position **x** within the element is given by an element basis function $\psi_{\beta(\varepsilon)}(\mathbf{x})$, $\mathbf{x} \in \omega^\varepsilon$. $\psi_{\beta(\varepsilon)}(\mathbf{x})$ is thus a function of the element position **x**. Its value represents the weight that the element field parameter $\mathbf{u}^{\beta(\varepsilon)}$ contributes to the field at **x**.

(7) At any point within the element the field is a linear combination of these parameters, $\mathbf{u}_{(\varepsilon)}(\mathbf{x}) = \psi_{\beta(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{\beta(\varepsilon)}$. Restricting $\mathbf{u}_{(\varepsilon)}(\mathbf{x})$ to be a linear combination of $\mathbf{u}^{\beta(\varepsilon)}$ is a practical constraint. It enables the direct solution of large classes of approximation problems since the resulting systems of equations are linear. As a consequence of this restriction $\{\mathbf{u}_{(\varepsilon)}(\mathbf{x})\}$ is a finite dimensional linear space on ω^{ε}.

(8) The field over the region covered by the elements is given by the union of fields within each element, $\mathbf{U} = \cup_{\varepsilon}\, \mathbf{u}_{(\varepsilon)}$.

(9) In general the ensemble field will be required to satisfy continuity conditions. These conditions are manifested as linear relations between the element field parameters of contiguous elements. Note that the relations are linear since $\mathbf{u}_{(\varepsilon)}(\mathbf{x})$ is a linear combination of $\mathbf{u}^{\beta(\varepsilon)}$ with coefficients $\psi_{\beta(\varepsilon)}(\mathbf{x})$. The continuity conditions will in general reduce the number of linearly independent parameters required to characterise the field.

(10) The linearly independent parameters used to define the ensemble field are called the ensemble field parameters, $\mathbf{U}^{B}$.

(11) The linear mapping, $\upsilon^{\beta(\varepsilon)}{}_{B}$, between the ensemble field parameters and the element field parameters is called the ensemble to element parameter map, $\mathbf{u}^{\beta(\varepsilon)} = \upsilon^{\beta(\varepsilon)}{}_{B}\ \mathbf{U}^{B}$. Here again restricting $\upsilon^{\beta(\varepsilon)}{}_{B}$ to be a linear map is a practical constraint.

(12) Associated with each ensemble parameter is an ensemble basis function, $\Psi_{B}(\mathbf{X}) = \cup_{\varepsilon}\, \psi_{\beta(\varepsilon)}(\mathbf{x})\ \upsilon^{\beta(\varepsilon)}{}_{B}$, where $\mathbf{X} \in \Omega^{E}$, $\mathbf{x} \in \omega^{\varepsilon}$. Each ensemble basis function represents the weight that the associated ensemble field parameter $\mathbf{U}^{B}$ contributes to the field at $\mathbf{X}$. The collection of ensemble basis functions represents a basis for fields defined on Ω^{E}.

2.2 Element Basis Functions

Any set of linearly independent functions $\{\psi_{\beta(\varepsilon)}(\mathbf{x})\}$ defined over an element ε may be used as a basis for a linear space on ε. Most finite element and graphics applications have used polynomials or rational polynomials, but non-integer power series or transcendental functions are sometimes useful. Fix (1969), for example, in considering the convergence of Rayleigh-Ritz approximations to elliptic boundary value problems found that convergence rates were improved considerably by the inclusion of appropriate singular basis functions in elements near the solution singularity. Such schemes perform best where the form of the singularity is known *a priori*. Fortunately these may often be deduced analytically from local asymptotic expansions about the singular region (Lehman 1959). It should be noted, however, that on computers transcendental functions are usually approximated by polynomials or rational polynomials (Abramowitz and Stegun 1964) (IBM 1981).

We shall confine our attention here to polynomial bases of finite order which are easy to add, multiply, differentiate and integrate and will serve to illustrate the concepts discussed in this section. Polynomial basis functions are expressed as powers of the element coordinates x^l, with the degree of the polynomial in the l^{th} coordinate direction being equal to the highest power of x^l appearing in the basis. For convenience and with no loss of generality we restrict x^l to lie in the range [0,1], noting that the degree of the polynomial will in general remain unchanged under a linear transformation in x^l.

Let $L=\dim(\omega\varepsilon)$ and let P_l be the degree of the polynomial in the direction of $\mathbf{x}^l$. If $\mathbf{x}=(x^1,x^2,...x^L)$ and $\mathbf{P}=(P_1,P_2,...P_L)$ then let $\wp[\mathbf{x},\mathbf{P}]$ represent the space of polynomials of degree P_l in each x^l. $\wp[\mathbf{x},\mathbf{P}]$ forms a linear space of dimension $\prod_{l=1...L}(P_l+1)$. Thus for a set of basis functions to span $\wp[\mathbf{x},\mathbf{P}]$ they must consist of $\prod_{l=1...L}(P_l+1)$ linearly independent polynomials in $\wp[\mathbf{x},\mathbf{P}]$ over $\omega\varepsilon$.

One basis spanning $\wp[\mathbf{x},\mathbf{P}]$ is simply the $\prod_{l=1...L}(P_l+1)$ functions defined by taking the monomials or individual powers of $\mathbf{x}$. Where we wish the basis functions to be interpreted as vectors in $\wp[\mathbf{x},\mathbf{P}]$ we shall denote them by boldface type.

For L=1 the monomial basis functions are:

$\mathbf{e}_1 = 1$ $\quad \mathbf{e}_2 = x^1$ $\quad \cdots \quad \mathbf{e}_{(P_1+1)} = (x^1)^{P_1}$

For L=2:

$\mathbf{e}_1 = 1$	$\mathbf{e}_2 = x^1$	$\cdots$	$\mathbf{e}_{(P_1+1)} = (x^1)^{P_1}$
$\mathbf{e}_{(P_1+2)} = x^2$	$\mathbf{e}_{(P_1+3)} = x^1 x^2$	$\cdots$	$\mathbf{e}_{(2P_1+2)} = (x^1)^{P_1} x^2$
:	:		:
$\mathbf{e}_{(P_1P_2+P_2+1)} = (x^2)^{P_2}$	$\mathbf{e}_{(P_1P_2+P_2+2)} = x^1 (x^2)^{P_2}$	$\cdots$	$\mathbf{e}_{((P_1+1)(P_2+1))} = (x^1)^{P_1} (x^2)^{P_2}$

Since we shall be expressing all polynomial basis functions in terms of monomials we shall refer to the monomial basis $\{\mathbf{e}_p(\mathbf{x})\}$ as the natural basis for polynomials, i.e.

$\psi_\beta(\mathbf{x}) = \psi_{\beta p}\, \mathbf{e}_p(\mathbf{x})$

where $\psi_{\beta p}$ are the basis function coefficients of $\psi_\beta(\mathbf{x})$ with respect to the monomial basis $\{\mathbf{e}_p(\mathbf{x})\}$.

There are at least two sources of error affecting the accuracy of polynomials. The first, the condition of the form, relates to the form of the polynomial while the second, the stability or the condition of the algorithm, is dependent upon the algorithm used to evaluate it. The accuracy to which a polynomial may be evaluated is clearly dependent upon both.

Consider a polynomial, $\mathbf{p}(\mathbf{x})$, expressed with respect to some basis $\{\mathbf{p}^p(\mathbf{x})\}$. Defining this polynomial are a finite set of coefficients $\{a_p\}$. In the case where $\mathbf{p}(\mathbf{x})$ is a linear combination of $\{\mathbf{p}^p(\mathbf{x})\}$ the polynomial is defined by:

$\mathbf{p}(\mathbf{x}) = \Sigma_{p=0\ldots P}\, a_p\, \mathbf{p}^p(\mathbf{x})$

Now consider a perturbation or error, Δa_p, associated with the p^{th} polynomal coefficient. This perturbation in the coefficient will give rise to a corresponding change, $\Delta\mathbf{p}(\mathbf{x},\Delta a_p)$, in the value of the polynomial. Where $\{\Delta\mathbf{p}(\mathbf{x},\Delta a_p)\}$ are small for small $\{\Delta a_p\}$ we say that the corresponding polynomial is well-conditioned. Conversely, where $\{\Delta\mathbf{p}(\mathbf{x},\Delta a_p)\}$ are large for small $\{\Delta a_p\}$ we say that the corresponding polynomial is ill-conditioned. Notice that we have made no reference to the algorithm used to evaluate the polynomial, the condition of a polynomial is dependent only upon its coefficients and its form.

More formally, we can define a condition number for a polynomial map as follows:

$\text{cond}(\mathbf{p}(\mathbf{x})) = ||\partial \mathbf{p}(\mathbf{x})/\partial a_p||\ ||a_p||/||\mathbf{p}(\mathbf{x})||$

The condition number is a measure of the maximum amount by which $\mathbf{p}(\mathbf{x})$ is changed by a perturbation in a_p. Maps with a large condition number are called ill-conditioned.

The stability of an algorithm is a measure of the degree to which perturbations in the input values, a_p and x, affect the computed values. An unstable algorithm will produce a growth in both roundoff errors and errors in the input values. If C_a represents the condition number for the algorithm, C_p represents the condition number for the polynomial form, δ_f is the error in the floating point representation of the input and δ_i is the relative error of the input then the bound for the relative error in the output δ_o is given by (Dahlquist *et al* 1974):

$\delta_o \leq C_p\ (\delta_i + C_a\ \delta_f)$

If one evaluates a polynomial expressed with respect to different bases using finite accuracy arithmetic quite different values may be obtained. Such a result arises because different bases propagate numerical inaccuracies associated with the abscissa and coefficients to the function value at different rates. In essence this is simply a consequence of the combined condition numbers of the polynomial form and the evaluation algorithm differing from one form and algorithm to another.

In the following sections we shall consider some issues associated with representing and evaluating element bases using polynomials. Any polynomial contained within the span of a polynomial basis may be expressed exactly with respect to that basis. Indeed, since the space of polynomials is linear we may readily convert a polynomial from one basis to another, which spans the first, using a linear transformation. Just how one determines the the coefficients of this transformation is addressed in the following section. Subsequent sections consider a number of one dimensional polynomial bases, their condition, and the coefficients of their linear transformations to and from the monomial basis. The final sections deal with some algorithms used for evaluating polynomials, their operation counts and their condition.

2.2.1 Transformation Of Element Basis Functions

Polynomial bases may assume different forms. Common one dimensional representations include monomials, Lagrange, Hermite, and Chebyshev functions. The list of bases expands significantly for two and three dimensional representations. Where one basis is contained in the space of another we wish to be able to transform a field expressed in terms of the first basis to an equivalent field expressed in terms of the second. Where the first basis is not contained in the space of the second an exact transformation is in general impossible. However if we define an inner product on a space spanning both bases we can determine the projection of the first basis upon the second. This section attempts to address this problem. While we restrict ourselves here to one dimensional polynomial bases, the following analysis may also be applied to higher dimensional and non-polynomial bases.

Polynomials of order less than or equal to P may be viewed as a P+1 dimensional linear space, denoted $\wp[\mathbf{x},\mathbf{P}]$. A natural basis covering $\wp$ is the set of P+1 monomials $\{\mathbf{e}_p(\mathbf{x})\}$ up to order P.

Let $\wp^*$ represent the dual linear space to $\wp$. The elements of $\wp$ are called contravariant vectors while elements of $\wp^*$ are called covariant vectors. Given a basis $\{\mathbf{e}_p\}$ for $\wp$ we can define a dual basis $\{\mathbf{e}^p\}$ for $\wp^*$ by:

$\langle \mathbf{e}^p , \mathbf{e}_{p'} \rangle = \delta^p_{p'}$

where $\langle , \rangle : \wp^* \times \wp \rightarrow \mathbb{R}$ denotes the duality between $\wp^*$ and $\wp$.

Let $\mathbf{p} \in \wp$ be expanded in terms of $\{\mathbf{e}_p\}$:

$\mathbf{p} = p^p \mathbf{e}_p$

The contravariant components, p^p, of $\mathbf{p}$ are given by the contraction of $\mathbf{e}^p$ onto $\mathbf{p}$:

$$\begin{aligned}\langle \mathbf{e}^p , \mathbf{p} \rangle &= \langle \mathbf{e}^p , p^{p'} \mathbf{e}_{p'} \rangle \\ &= p^{p'} \langle \mathbf{e}^p , \mathbf{e}_{p'} \rangle \\ &= p^{p'} \delta^p_{p'} \\ &= p^p\end{aligned}$$

Similarly, let $\mathbf{p}^* \in \wp^*$ be expanded in terms of $\{\mathbf{e}^p\}$:

$\mathbf{p}^* = p_p \, \mathbf{e}^p$

The covariant components, p_p, of $\mathbf{p}^*$ are given by the contraction of $\mathbf{p}^*$ onto $\mathbf{e}_p$:

$\langle \mathbf{p}^*, \mathbf{e}_p \rangle = \langle p_{p'} \, \mathbf{e}^{p'}, \mathbf{e}_p \rangle$

$= p_p$

We now introduce a metric tensor $\mathbf{g}$ on $\wp$ such that:

(1) $\mathbf{g}$ is symmetric

$\mathbf{g}(\mathbf{p},\mathbf{p}') = \mathbf{g}(\mathbf{p}',\mathbf{p}) \qquad \forall \, \mathbf{p},\mathbf{p}' \in \wp$

(2) $\mathbf{g}$ is non-degenerate and positive definite

$\mathbf{g}(\mathbf{p},\mathbf{p}') = 0 \qquad \forall \, \mathbf{p} \in \wp \Leftrightarrow \mathbf{p} = \mathbf{0}$

$\mathbf{g}(\mathbf{p},\mathbf{p}) > 0 \qquad \forall \, \mathbf{p} \in \wp \qquad \mathbf{p} \neq \mathbf{0}$

This metric tensor endows the space $\wp$ with an inner product:

$\mathbf{g}(\mathbf{p},\mathbf{p}') = (\mathbf{p} \mid \mathbf{p}')$

The components of $\mathbf{g}$ expressed with respect to $\{\mathbf{e}_p\}$ are given by:

$g_{pp'} = \mathbf{g}(\mathbf{e}_p,\mathbf{e}_{p'})$

$= (\mathbf{e}_p \mid \mathbf{e}_{p'})$

An inner product on any vector space defines a natural isomorphism between the space, $\wp$, and its dual, $\wp^*$. Let $\mathbf{p} \in \wp$, and $\mathbf{p}^* \in \wp^*$ be the dual of $\mathbf{p}$. The natural isomorphism is defined by:

$\langle \mathbf{p}^*, \mathbf{p}' \rangle \equiv (\mathbf{p} \mid \mathbf{p}') \qquad \forall \, \mathbf{p}' \in \wp$

In general the same symbol, $\mathbf{p}$, is used to denote the vector, $\mathbf{p}$, and its associated dual vector, $\mathbf{p}^*$.

The natural isomorphism relates the contravariant components, p^p, of the vector $\mathbf{p}$ to the covariant components, p_p, of its corresponding dual vector $\mathbf{p}^*$:

$p_p = \langle \mathbf{p}^*, \mathbf{e}_p \rangle$

$\equiv (\mathbf{p} \mid \mathbf{e}_p)$

$= p^{p'} \, (\mathbf{e}_{p'} \mid \mathbf{e}_p)$

$= p^{p'} \, g_{p'p}$

Let $g^{pp'}$ be the components of the matrix inverse of $[g_{pp'}]$:

$g^{pp'} g_{p'p''} = \delta^{p}_{p''}$

Then the metric, **g**, also relates the covariant components of **p*** to its contravariant components:

$p^{p} = g^{pp'} p_{p'}$

An inner product is also defined on $\wp^*$ by the natural isomorphism:

$$
\begin{aligned}
(\mathbf{p}^* \mid \mathbf{p}'^*) &= (\mathbf{p} \mid \mathbf{p}') \\
&= (p^{p} \mathbf{e}_{p} \mid p'^{p'} \mathbf{e}_{p'}) \\
&= p^{p} p'^{p'} (\mathbf{e}_{p} \mid \mathbf{e}_{p'}) \\
&= p^{p} p'^{p'} g_{pp'} \\
&= p^{p} p'_{p}
\end{aligned}
$$

Now consider a subspace of $\wp$ spanned by a finite number of basis vectors $\{\psi_{\beta}(\mathbf{x})\}$ of order less than or equal to P. The metric of $\wp$ induces a metric on the subspace:

$$
\begin{aligned}
g_{\beta\beta'} &= g(\psi_{\beta}(\mathbf{x}), \psi_{\beta'}(\mathbf{x})) \\
&= (\psi_{\beta}(\mathbf{x}) \mid \psi_{\beta'}(\mathbf{x})) \\
&= \psi_{\beta}{}^{p} \; \psi_{\beta'}{}^{p'} \; (\mathbf{e}_{p} \mid \mathbf{e}_{p'}) \\
&= \psi_{\beta}{}^{p} \; \psi_{\beta'}{}^{p'} \; \mathbf{g}_{pp'}
\end{aligned}
$$

Letting $g^{\beta\beta'}$ be the components of the matrix inverse of $[g_{\beta\beta'}]$ we can construct the dual basis $\{\psi^{\beta}(\mathbf{x})\}$ to $\{\psi_{\beta}(\mathbf{x})\}$:

$\psi^{\beta}(\mathbf{x}) = g^{\beta\beta'} \; \psi_{\beta'}(\mathbf{x})$

We now introduce a second subspace of $\wp$ spanned by the basis vectors $\{\psi'_{\beta}(\mathbf{x})\}$ of order less than or equal to P. We would like to express each $\psi_{\beta}(\mathbf{x})$ in terms of $\{\psi'_{\beta}(\mathbf{x})\}$. Recall that the coefficients of each basis function represented in terms of the monomial basis are simply the contraction of that function onto each dual basis function. In a similar manner we can evaluate the mapping coefficients to express a basis function $\psi_{\beta}(\mathbf{x})$ in terms of another basis $\{\psi'_{\beta}(\mathbf{x})\}$ by taking the contraction of that function with the dual of each function in the new basis

$$
\begin{aligned}
\langle \psi'^{\beta'} , \psi_{\beta} \rangle &= \psi'^{\beta' p'} \; \psi_{\beta}{}^{p} \; g_{p'p} \\
&= \psi_{\beta}{}^{p} \; \psi'^{\beta' p'} \; (\mathbf{e}_{p} \mid \mathbf{e}_{p'})
\end{aligned}
$$

It should be noted that if $\psi_\beta(\mathbf{x})$ is not contained in the span of $\{\psi'_\beta(\mathbf{x})\}$ then in general the function cannot be represented exactly in terms of $\{\psi'_\beta(\mathbf{x})\}$. In this case the mapping coefficients will only provide the projection of $\psi_\beta(\mathbf{x})$ onto $\{\psi'_\beta(\mathbf{x})\}$. In converting from one basis to another, having determined the mapping coefficients, one should try and construct the original basis function by taking the linear combination of the second basis with the coefficients as weights. If the original basis function is contained in the span of the second basis then it will be represented exactly by this combination.

If

$\mathbf{u} = \psi_\beta(\mathbf{x})\, u^\beta$

$= \psi'_{\beta'}(\mathbf{x})\, u'^{\beta'}$

Let $\psi_\beta(\mathbf{x}) \perp \{\psi'_\beta(\mathbf{x})\}$ denote the projection of $\psi_\beta(\mathbf{x})$ on $\{\psi'_\beta(\mathbf{x})\}$. Then

$$\begin{aligned}\psi_\beta(\mathbf{x}) \perp \{\psi'_\beta(\mathbf{x})\} &= (\psi_\beta{}^\rho\, \psi'^{\beta'\rho'}\, (\mathbf{e}_\rho \mid \mathbf{e}_{\rho'}))\, \psi'_{\beta'}(\mathbf{x}) \\ &= (\psi_\beta{}^\rho\, \psi'^{\beta'\rho'}\, g_{\rho\rho'})\, \psi'_{\beta'}(\mathbf{x}) \\ &= (\psi_{\beta\rho'}\, \psi'^{\beta'\rho'})\, \psi'_{\beta'}(\mathbf{x})\end{aligned}$$

Replacing $\psi_\beta(\mathbf{x})$ in the expression for $\mathbf{u}$ with the projected components of $\psi_\beta(\mathbf{x})$ on $\{\psi'_\beta(\mathbf{x})\}$:

$(\psi_{\beta\rho'}\, \psi'^{\beta'\rho'})\, \psi'_{\beta'}(\mathbf{x})\, u^\beta = \psi'_{\beta'}(\mathbf{x})\, u'^{\beta'}$

or

$(\psi_{\beta\rho'}\, \psi'^{\beta'\rho'})\, u^\beta = u'^{\beta'}$

since $\psi'_{\beta'}(\mathbf{x})$ is an arbitrary function of $\mathbf{x}$. We are thus able to transform the element field parameters of one basis into those of another using the basis transformation matrix $(\psi_{\beta\rho'}\, \psi'^{\beta'\rho'})$.

These ideas are illustrated below in a simple example. To facilitate the following computations the monomials are defined as forming an orthonormal basis:

$(\mathbf{e}_p \mid \mathbf{e}_{p'}) = \delta_{pp'}$

Let

$\psi_1(\mathbf{x}) = 1 + (x)^3$

$\psi_2(\mathbf{x}) = (x)^2$

and

$\psi'_1(\mathbf{x}) = 1 + (x)^2 + (x)^3$

$\psi'_2(\mathbf{x}) = x$

$\psi'_3(\mathbf{x}) = x - (x)^2$

i.e.

$\psi_\beta(\mathbf{x}) = \psi_\beta{}^p \, \mathbf{e}_p(\mathbf{x})$:

$$\begin{pmatrix} \psi_1(\mathbf{x}) \\ \psi_2(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix}$$

$\psi'_\beta(\mathbf{x}) = \psi'_\beta{}^p \, \mathbf{e}_p(\mathbf{x})$:

$$\begin{pmatrix} \psi'_1(\mathbf{x}) \\ \psi'_2(\mathbf{x}) \\ \psi'_3(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & -1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix}$$

Then $g'_{\beta\beta'} = \psi'_\beta{}^p \, \psi'_{\beta',p'} \, \delta_{pp'}$

$$= \begin{pmatrix} 3 & 0 & -1 \\ 0 & 1 & 1 \\ -1 & 1 & 2 \end{pmatrix}$$

$$\therefore \; g'^{\beta\beta'} = 1/2 \begin{pmatrix} 1 & -1 & 1 \\ -1 & 5 & -3 \\ 1 & -3 & 3 \end{pmatrix}$$

$\psi'^{\beta}(\mathbf{x}) = \psi'_{\beta'}(\mathbf{x})\, g'^{\beta\beta'}$

so

$$\psi'^{\beta p} = 1/2 \begin{pmatrix} 1 & 0 & 0 & 1 \\ -1 & 2 & 2 & -1 \\ 1 & 0 & -2 & 1 \end{pmatrix} \begin{matrix} \downarrow \beta \\ \\ \\ p \rightarrow \end{matrix}$$

$$\langle \psi'^{\beta'}, \psi_{\beta} \rangle = \begin{pmatrix} 1 & 0 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{matrix} \downarrow \beta' \\ \\ \\ \beta \rightarrow \end{matrix}$$

Thus

$$(\psi_1(\mathbf{x}), \psi_2(\mathbf{x})) = (\psi'_1(\mathbf{x}), \psi'_2(\mathbf{x}), \psi'_3(\mathbf{x})) \begin{pmatrix} 1 & 0 \\ -1 & 1 \\ 1 & -1 \end{pmatrix}$$

and

$$\begin{pmatrix} u'^1 \\ u'^2 \\ u'^3 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} u^1 \\ u^2 \end{pmatrix}$$

2.2.2 Polynomial Basis Functions

Polynomial bases enjoy a privileged status in the representation of fields by piecewise functions. Their properties have been the subject of much investigation and are generally well understood. Much of this attention is a consequence of the attractive features of polynomials. These features include an ability to determine analytically the error properties of the different representations as well as the relative ease with which one can manipulate polynomials both numerically and algebraically.

As was mentioned in section 2.2 $\wp[\mathbf{x},\mathbf{P}]$ forms a linear space over the domain of **x**. One is thus able to select any set of polynomials spanning $\wp[\mathbf{x},\mathbf{P}]$ as a basis for the space. In practice a remarkably large number of polynomial families have been used as bases for piecewise polynomial representations of fields. Catalogues of the more popular bases are available in Oden (1972), Zienkiewicz (1977), and Zienkiewicz and Morgan (1982) for finite elements and Foley and van Dam (1983), Barsky (1984), and Farouki and Hinds (1985) for computer graphics. As a consequence of the linearity of $\wp[\mathbf{x},\mathbf{P}]$ one is able to transform a field basis using a linear transformation of the field parameters.

In the following sections we consider the properties of a number of one dimensional polynomial bases commonly used in the representation of fields by piecewise functions. It should be noted that the construction of higher dimensional bases from lower ones may be readily achieved using tensor product or blending methods (Ferguson 1964) (Coons 1967) (Gordon 1971). The one dimensional representations considered include the monomial, Lagrange, Hermite, Bézier, and Chebyshev bases.

2.2.2.1 Monomial Basis

The monomial basis of degree P consists of the P+1 powers of x from 0 to P:

$\psi_\beta(x) = (x)^{\beta-1}$ $\beta=1...P+1$

The powers of x may be considered a natural basis for the space of polynomials. When dealing with other polynomial bases in subsequent sections we shall give the linear transformations between that basis and the monomial basis. The monomial basis is sometimes referred to as the power basis or the algebraic polynomial basis.

The coefficients of a field expressed with respect to the monomial basis lack a clear physical interpretation. In general, polynomials possessing large coefficients for high order monomials will appear less smooth than those with smaller coefficients. The order of the polynomial is, however, immediately apparent from the highest power of x possessing a nonzero coefficient. This feature is sometimes useful since It may be difficult to determine the order of a polynomial expressed with respect to other bases where it has degenerated to one of lower degree.

The condition of the monomial basis has been studied by Gautschi (1978) for x defined over an interval symmetric about the origin, $[-x^*,x^*]$, as well as x defined over $[0,x^*]$. In both cases the condition grows exponentially with the power of the polynomial, the asymptotic growth rate being smallest, $(1+\sqrt{2})^P$ and $(1+\sqrt{2})^{2P}$ respectively, when $x^*=1$.

2.2.2.2 Lagrange Basis

Given a set of P+1 control points or nodes defined at unique positions within an element the Lagrange basis of order P is constructed so that each of the P+1 basis functions is unity at one and only one node and zero at all others. At the position of a node a field expressed in terms of the Lagrange basis assumes the value of the coefficient associated with that node. Since all the other basis functions are zero none of the other coefficients contribute to the field value at that nodal position.

The P+1 Lagrange basis functions of order P may be constructed as follows. Given P+1 unique nodal positions $\{x_1, x_2, x_3, \ldots x_{P+1}\}$ the basis function $\psi_\beta(x)$ is defined by:

$\psi_\beta(x) = \prod_{p=1..P+1,\, p\neq\beta} (x - x_\beta)/(x_p - x_\beta)$

From this definition it can be seen that unless the nodal postions are unique the basis functions associated with the non-unique nodes will be undefined.

Lagrange bases are usually defined with equally spaced nodes, the first and last nodes lying on the boundary of the element. This choice of nodal positioning may, however, be ill-conditioned (de Boor 1978). Rack and Reimer (1982) demonstrated that good conditioning is obtained with the Lagrange nodes placed at the Fekete points (roots of the Jacobi polynomials $P^{(-1,-1)}{}_n$) or the Chebyshev points (roots of the Chebyshev polynomials T_n). Gautschi (1978) cites a private communication with de Boor claiming that the basis consisting of the Lagrange polynomials for Chebyshev nodes is optimally conditioned among all polynomial bases, having an asymptotic condition number of $2\log_2(P)/\pi$.

In the following we give the transformations between the monomial and Lagrange bases of order 1, 2 and 3 defined for $x \in [0,1]$ with equally spaced nodes.

Let

$\mathbf{u}(x) = \mathbf{e}_p(x)\, \mathbf{u}^p$

where $\mathbf{e}_p(x)$ are one dimensional monomial basis functions and

$\mathbf{u}(x) = \psi_\beta(x)\, \mathbf{u}^\beta$

where $\psi_\beta(x)$ are Lagrange basis functions.

Linear Lagrange:

$$\begin{pmatrix} \psi_1 \\ \psi_2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \end{pmatrix}$$

Quadratic Lagrange:

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{pmatrix} = \begin{pmatrix} 1 & -3 & 2 \\ 0 & 4 & -4 \\ 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \end{pmatrix} = 1/4 \begin{pmatrix} 4 & 4 & 4 \\ 0 & 2 & 4 \\ 0 & 1 & 4 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \end{pmatrix}$$

Cubic Lagrange:

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix} = 1/2 \begin{pmatrix} 2 & -11 & 18 & -9 \\ 0 & 18 & -45 & 27 \\ 0 & -9 & 36 & -27 \\ 0 & 2 & -9 & 9 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix} = 1/27 \begin{pmatrix} 27 & 27 & 27 & 27 \\ 0 & 9 & 18 & 27 \\ 0 & 3 & 12 & 27 \\ 0 & 1 & 8 & 27 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix}$$

2.2.2.3 Hermite Basis

The Hermite polynomials of odd order P are constructed from the field values and derivatives up to order (P–1)/2 at the element boundaries. By specifying the field values and derivatives at the boundary it is a relatively simple procedure to ensure (P–1)/2th order field continuity between elements by identifying the element parameters at the boundary of one element with those of the next. Efficient use of the field parameters is made in this case since the field values and derivatives are shared between contiguous elements.

Let

$\mathbf{u}(x) = \mathbf{e}_p(x)\, \mathbf{u}^p$

where $\mathbf{e}_p(x)$ are one dimensional monomial basis functions and

$\mathbf{u}(x) = \psi_\beta(x)\, \mathbf{u}^\beta$

where $\psi_\beta(x)$ are Hermite basis functions with $x \in [0,1]$.

Linear Hermite:

$$\begin{pmatrix} \psi_1 \\ \psi_2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \end{pmatrix}$$

Cubic Hermite:

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 & -3 & 2 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 3 & -2 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 1 & 3 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix}$$

Quintic Hermite:

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \psi_5 \\ \psi_6 \end{pmatrix} = 1/2 \begin{pmatrix} 2 & 0 & 0 & -20 & 30 & -12 \\ 0 & 2 & 0 & -12 & 16 & -6 \\ 0 & 0 & 1 & -3 & 3 & -1 \\ 0 & 0 & 0 & 20 & -30 & 12 \\ 0 & 0 & 0 & -8 & 14 & -6 \\ 0 & 0 & 0 & 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \\ (x)^4 \\ (x)^5 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \\ (x)^4 \\ (x)^5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 2 & 1 & 2 & 2 \\ 0 & 0 & 0 & 1 & 3 & 6 \\ 0 & 0 & 0 & 1 & 4 & 12 \\ 0 & 0 & 0 & 1 & 5 & 20 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \\ \psi_5 \\ \psi_6 \end{pmatrix}$$

2.2.2.4 Bézier Basis

The Bézier polynomials are similar in form to the Hermite family. The boundary field values for both families are equal to the field values associated with the endpoint nodes or control points. However, in the Hermite family the boundary derivatives are defined directly while in the Bézier form they are determined by the field values associated with the boundary and internal control points.

Two properties of the Bézier form make it attractive for use in geometric modelling in interactive graphics applications. Firstly, the control of fields by manipulating the positions of discrete points is intuitively appealing. Secondly, the curve so generated lies within the convex hull formed by the control points.

Let

$\mathbf{u}(x) = \mathbf{e}_p(x)\ \mathbf{u}^p$

where $\mathbf{e}_p(x)$ are one dimensional monomial basis functions and

$\mathbf{u}(x) = \psi_\beta(x)\ \mathbf{u}^\beta$

where $\psi_\beta(x)$ are Bézier basis functions (Bernstein polynomials) with $x \in [0,1]$.

Cubic Bézier:

The cubic Bézier curve uses four points ($\mathbf{u}^1$, $\mathbf{u}^2$, $\mathbf{u}^3$, $\mathbf{u}^4$) to control the field. Control points $\mathbf{u}^1$ and $\mathbf{u}^4$ are equal to the field values at the element boundaries while the difference between each central control point, $\mathbf{u}^2$ and $\mathbf{u}^3$, and its nearest boundary point determines the field derivative at the boundaries:

$\mathbf{u}(0) = \mathbf{u}^1$

$\mathbf{u}(1) = \mathbf{u}^4$

$\partial\mathbf{u}(0)/\partial x = 3(\mathbf{u}^2 - \mathbf{u}^1)$

$\partial\mathbf{u}(1)/\partial x = 3(\mathbf{u}^4 - \mathbf{u}^3)$

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix} = \begin{pmatrix} 1 & -3 & 3 & -1 \\ 0 & 3 & -6 & 3 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix} = 1/3 \begin{pmatrix} 3 & 3 & 3 & 3 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 3 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix}$$

Cubic Bézier (Timmer 1980):

Timmer (1980) proposed a modification to the cubic Bézier to provide better control of the field. By moving the two central control points closer to the boundaries the field at the centre of the element could be made equal to the average of the central point values. A more intuitive control of the field was obtained at the expense of the convex hull property of the normal Bézier curve.

$\mathbf{u}(0) = \mathbf{u}^1$

$\mathbf{u}(1) = \mathbf{u}^4$

$\partial \mathbf{u}(0)/\partial x = 4(\mathbf{u}^2 - \mathbf{u}^1)$

$\partial \mathbf{u}(1)/\partial x = 4(\mathbf{u}^4 - \mathbf{u}^3)$

$$\begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix} = \begin{pmatrix} 1 & -4 & 5 & -2 \\ 0 & 4 & -8 & 4 \\ 0 & 0 & 4 & -4 \\ 0 & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \end{pmatrix} = 1/4 \begin{pmatrix} 4 & 4 & 4 & 4 \\ 0 & 1 & 3 & 4 \\ 0 & 0 & 2 & 4 \\ 0 & 0 & 1 & 4 \end{pmatrix} \begin{pmatrix} \psi_1 \\ \psi_2 \\ \psi_3 \\ \psi_4 \end{pmatrix}$$

2.2.2.5 Chebyshev Basis

The Chebyshev polynomials of order P have P+1 extreme values of equal magnitude on the interval [–1,1]. These extrema are located at $x = \cos(\pi p/P)$, $p = 0...P$, and alternate in value between –1 and 1. The P real roots of the P^{th} order Chebyshev polynomials, also called the Chebyshev points, are located at $x = \cos(2\pi(p+1)/2P)$, $p = 0...P-1$. It can be shown that for a function defined over the domain [–1,1] a P^{th} order polynomial which passes through that function at the P+1 Chebyshev points gives a much better approximation to the function than a polynomial which passes through P+1 equidistant points.

Another important property of the orthogonal polynomials is their relatively good conditioning. Gautschi (1978) found that the condition of orthogonal bases exhibit only polynomial growth in P. For Chebyshev polynomials defined on [–1,1] the condition number is less than or equal to $P\sqrt{2}$, a marked improvement over the monomial basis. An extensive study the properties of Chebyshev polynomials may be found in Rivlin (1974).

Let

$\mathbf{u}(x) = \mathbf{e}_p(x)\, \mathbf{u}^p$

where $\mathbf{e}_p(x)$ are one dimensional monomial basis functions and

$\mathbf{u}(x) = \psi_\beta(x)\, \mathbf{u}^\beta$

where $\psi_\beta(x) = T_{\beta-1}(x)$ are Chebyshev basis functions defined on the domain [–1,1].

$$\begin{pmatrix} T_0 \\ T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 2 & 0 & 0 & 0 \\ 0 & -3 & 0 & 4 & 0 & 0 \\ 1 & 0 & -8 & 0 & 8 & 0 \\ 0 & 5 & 0 & -20 & 0 & 16 \end{pmatrix} \begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \\ (x)^4 \\ (x)^5 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ x \\ (x)^2 \\ (x)^3 \\ (x)^4 \\ (x)^5 \end{pmatrix} = 1/16 \begin{pmatrix} 16 & 0 & 0 & 0 & 0 & 0 \\ 0 & 16 & 0 & 0 & 0 & 0 \\ 8 & 0 & 8 & 0 & 0 & 0 \\ 0 & 12 & 0 & 4 & 0 & 0 \\ 6 & 0 & 8 & 0 & 2 & 0 \\ 0 & 10 & 0 & 5 & 0 & 1 \end{pmatrix} \begin{pmatrix} T_0 \\ T_1 \\ T_2 \\ T_3 \\ T_4 \\ T_5 \end{pmatrix}$$

Coefficients for the transformation to and from the monomial basis for Chebyshev polynomials up to order 12 may be found in Abramowitz and Stegun (1964).

2.2.3 Algorithms For Evaluating Polynomials

We examine here the efficiency and accuracy of some polynomial algorithms. Our attention will be confined to polynomials of one variable. Polynomials of more than one variable may be evaluated recursively using one variable algorithms. For example, consider the two variable polynomial represented in monomial form by

$p(\mathbf{x}) = \Sigma_{p_2=0..P_2}(\Sigma_{p_1=0..P_1} a_{p_1p_2}(x^1)^{p_1}) (x^2)^{p_2}$

Note that the sum over p_1 is a polynomial of one variable, x^1, with coefficients $a_{p_1p_2}$(p_2 fixed). The sum over p_2 is again a polynomial of one variable, x^2, with coefficients

$(\Sigma_{p_1=0..P_1} a_{p_1p_2}(x^1)^{p_1})$

and may thus be evaluated using the one variable polynomial algorithm.

Basis functions of more than one dimension are often constructed by taking tensor products of one dimensional polynomials. Significant savings can be made in evaluating such functions although the class of functions which can be represented by tensor product methods is restricted. For example, bilinear basis functions are often given as:

$\psi(\mathbf{x}) = (b_{10} + (b_{11}\, x^1))\, (b_{20} + (b_{21}\, x^2))$

which can be evaluated in 3 multiplications and 2 additions. Compare this with the monomial form

$\psi(\mathbf{x}) = a_{00} + (a_{10}\, x^1) + (a_{01}\, x^2) + (a_{11}\, x^1\, x^2)$

which requires 4 multiplications and 3 additions using Horner's algorithm. Any tensor product bilinear basis function can be represented in monomial form using the transformation:

$a_{00} = b_{10}\, b_{20}$

$a_{10} = b_{11}\, b_{20}$

$a_{01} = b_{10}\, b_{21}$

$a_{11} = b_{11}\, b_{21}$

Note, however, that this transformation is not in general invertible. For instance

$\psi(\mathbf{x}) = 1 + x^1 + x^2$

is not representable in the tensor product form.

It should be noted that in the calculation of a finite element stiffness matrix the basis functions and their derivatives are often evaluated at quadrature points once only, and stored, so the efficiency of basis evaluation is not critical. However, in graphics applications including post-processing of solutions from finite element analysis, where the field must be evaluated at arbitrary points within an element, computational efficiency needs to be considered.

We begin this section by stating a theorem due to Belaga (1961):

> Any algorithm for evaluating a polynomial of degree P without preprocessing of the coefficients requires at least P multiplications and P additions.

The caveat on preprocessing the coefficients is important. Motzkin (1955) showed that the operation count may be reduced to below P multiplications and/or additions if the polynomial coefficients are rearranged. This preprocessing of the coefficients involves further arithmetic operations but if the polynomial is evaluated at many points an overall savings on the arithmetic operations can be made. To determine the minimum number of operations required to evaluate a polynomial with preprocessed coefficients we turn to another theorem due to Belaga (1958):

> For any polynomial of degree P there exists an algorithm to evaluate it with $\lfloor (P+1)/2 \rfloor + 1$ multiplications and P+1 additions. No algorithm exists with less than $\lfloor (P+1)/2 \rfloor$ multiplications or with less than P additions.

In the following we shall consider a number of basic algorithms for evaluating polynomials. These include Horner's algorithm, Clenshaw's algorithm, the root product algorithm, algorithms involving preprocessed coefficients, and the forward difference algorithm used to evaluate polynomials at equally spaced intervals. Each of these schemes is designed to efficiently evaluate particular classes of polynomials: Horner's algorithm assumes that the polynomial is expanded in terms of the monomial basis; Clenshaw's algorithm requires that the polynomial is expressed in terms of orthogonal polynomials. Clearly, where the algorithm used and polynomial type do not match, the coefficients must be transformed to the appropriate basis. The methods considered are not intended to be exhaustive, they are merely a sample of commonly used algorithms for evaluating low order polynomials. Some

methods may be modified to evaluate polynomial derivatives. Where efficient modifications are known the algorithm for evaluating the derivatives will be discussed also.

In the above we have implicitly assumed that the algorithms would be implemented on a classic von Neumann single instruction single data (SISD) machine. There has been a growing interest in algorithms designed for multiple instruction multiple data (MIMD) machines. The study of the complexity of algorithms designed for MIMD computers is relatively new area. In the analysis one usually assumes that all of the processors are identical, the cost of interprocess communication is negligible, an unlimited number of processors is available, an unlimited amount of common memory is available with no access conflicts. Recent papers indicate that under the above conditions the time required to evaluate polynomials may be reduced from that required for SISD machines. For example, Knuth (1969) reports that the fastest serial algorithm for evaluating x^n requires $\log_2(n)+(\log_2(n)/\log_2(\log_2(n)))$ multiplications. Kung (1976) demonstrated a method for evaluating x^n in 2 parallel divisions and $\lceil\log_2(n)\rceil+2$ parallel additions thus obtaining an asymptotic speedup of the ratio of the time required for a serial multiplication divided by the time required for one addition. Lozinskii (1983) showed how the cost of evaluating x^n may be further reduced to $O((\log_2(n))^{1/2})$ multiplications using $O(n)$ processors and certain preprocessed data.

Kung (1976), in fact, presented a number of parallel algorithms to evaluate x^N, $\{ x,...x^N\}$, $\Pi_{n=1...N}\ (x + a_n)$, and $\Sigma_{n=1...N}\ a_n x$ on SIMD and MIMD machines using fewer equivalent multiplications than any known sequential algorithm. In particular $\Pi_{n=1...N}\ (x + a_n)$ could be evaluated in $\lceil\log_2(n)\rceil + 1$ parallel additions and 2 parallel divisions while $\Sigma_{n=1...N}\ a_n x^n$ required $2\lceil\log_2(n)\rceil + 6$ parallel additions and 6 parallel multiplications. These algorithms are all faster than known sequential algorithms when multiplication or division take more time than addition or subtraction.

In what follows we shall only consider the complexity and condition of a few sequential algorithms.

2.2.3.1 Horner's Algorithm

Let a polynomial of order P be expressed in monomial form

$p(x) = a_0 + a_1x + a_2(x)^2 + \ldots a_P(x)^P$

This polynomial may be written in Horner's nested form as

$p(x) = a_0 + x(a_1 + x(a_2 + \ldots x(a_P) \ldots))$

The corresponding algorithm is given by:

```
begin
    y := aP;
    for p in reverse 0..P–1 loop
        y := y * x + ap;
    end loop;
    p(x) := y;
end;
```

To evaluate p(x) at an arbitrary point x in this form requires only P multiplications and P additions. Borodin (1971) has shown that Horner's algorithm is unique in that it is the only algorithm without preprocessing of the coefficients which can evaluate $a_0 + a_1x + a_2(x)^2 + \ldots a_P(x)^P$ in P multiplications or divisions and P additions or subtractions.

Horner's algorithm clearly requires that the polynomial coefficients be expressed with respect to the monomial basis.

Rice (1965) performed numerical experments to determine the combined condition of Horner's algorithm and the monomial form. He found that under certain conditions Horner's algorithm may exhibit moderately severe ill-conditioning in comparison with Clenshaw's algorithm and the root product algorithm. Newbery (1974) compared the condition of Horner's and Clenshaw's algorithms using both the theoretical analysis of round-off error propagation and numerical experiments. In both cases it was found that the accuracy of the Horner scheme was highly sensitive to the magnitude of x and that no improvement can be

expected by reformulating the problem in terms of Chebyshev polynomials if the monomial coefficients are of constant or strictly alternating sign.

In order to minimise the above effects of ill-conditioning Newbery (1975) suggested several modifications to Horner's algorithm. These included a rescaling of x so that its maximum magnitude is always less than or equal to 1/2 then evaluating the even and odd powers of x separately using the Horner algorithm. A theoretical analysis of Newbery's algorithm indicated that the error bounds were consistently lower than Horner's. However, numerical experiments performed in the same paper did not bear these predictions out, apparently because the transformed coefficients needed to be computed in higher precision arithmetic (Oliver 1979). The above modifications require an extra multiplication and addition over Horner's algorithm for an equivalent polynomial.

Horner's algorithm may be extended to evaluate the first D derivatives of a polynomial of order P in (P–D/2)(D+1) multiplications and additions:

```
begin
    for d in 0..min(D,P) loop
        y^d_P := a_P;
        for p in reverse d..P–1 loop
            if d=0 then
                y^d_p := y^d_{p+1}x + a_p;
            else
                y^d_p := y^d_{p+1}x + y^{d–1}_p;
            end if;
        end loop;
        d^d/dx^d (p(x)) := d! y^d_d;
    end loop;
end;
```

Shaw and Traub (1974) proposed an algorithm to evaluate all the normalised derivatives, $d^d/dx^d\ (p(x))/d!$, of a polynomial of order P in 3P–2 multiplications and divisions and

P(P+1)/2 additions and subtractions. Woznaikowski (1974) showed that this algorithm is stable.

2.2.3.2 Clenshaw's Algorithm

Let $p(x) = \sum_{p=1...P} a_p q^p(x)$ where $q^p(x)$ are orthogonal polynomials of order p. A system of polynomials, $\{q^p(x)\}$, defined on the interval $x \in [x_0,x_1]$ is called orthogonal with respect to the weight function w(x) if

$\int_{x=x_1,x_2} q^p(x)\, q^{p'}(x)\, w(x)\, dx = 0 \qquad p \neq p'$

Orthogonal polynomials satisfy the linear recurrence relation:

$q^{p+1}(x) = (\alpha^p{}_1 x - \alpha^p{}_2)\, q^p(x) - \alpha^p{}_3\, q^{p-1}(x)$

where $q^{-1}(x)=0$ and $q^0(x)=1$. It should be noted that the $\alpha^p{}_i$ are dependent only upon the polynomial order and not upon x. Ignoring the cost of evaluating the $\alpha^p{}_i$ it can be seen that this algorithm requires 3 multiplications and 2 additions to evaluate $q^{P+1}(x)$ from $q^P(x)$ and $q^{P-1}(x)$. Clenshaw (1955) has shown that the sum of a series of functions generated by such linear recurrence relations can be evaluated in 3P–1 multiplications and additions using the following algorithm:

begin

$y_P := a_P$;

$y_{P-1} := (\alpha^{P-1}{}_1\, x - \alpha^{P-1}{}_2)\, y_P + a_{P-1}$;

for p **in reverse** 0..P–2 **loop**

$y_p := (\alpha^p{}_1\, x - \alpha^p{}_2)\, y_{p+1} + \alpha^{p+1}{}_3\, y_{p+2} + a_{P-1}$;

end loop;

$p(x) := y_0$;

end;

A short list of orthogonal polynomials, the domain of definition, the weighting functions w(x), and the coefficients $\alpha^p{}_i$ in the linear recurrence relation are given below. A more extensive list may be found in Abramowitz and Stegun (1964).

Name:	Domain:	w(x):	$\alpha^p{}_1$:	$\alpha^p{}_2$:	$\alpha^p{}_3$:
Legendre	[–1,1]	1	(2p+1)/(p+1)	0	p/(p+1)
Chebyshev	[–1,1]	$(1-(x)^2)^{-1/2}$	2	0	1
Laguerre	[0,∞)	e^{-x}	–1/(p+1)	(2p+1)/(p+1)	p/(p+1)
Hermite	(–∞,∞)	$e^{-(x)2}$	2	0	2p

Smith (1965) extended Clenshaw's algorithm to evaluate derivatives of orthogonal polynomial series by noting that

$d^d/dx^d\ (p(x)) = \Sigma_{p=0...P}\ a_p\ d^d/dx^d\ (q^p(x))$

The recurrence relation for $d^d/dx^d\ (q^p(x))$ is given by:

$$d^d/dx^d\ (q^{p+1}(x)) = (\alpha^p{}_1 x - \alpha^p{}_2)\ d^d/dx^d\ (q^p(x)) - \alpha^p{}_3\ d^d/dx^d\ (q^{p-1}(x)) + \alpha^p{}_1\ d^{d-1}/dx^{d-1}\ (q^p(x))$$

with $d^d/dx^d\ (q^{d-2}(x)) = 0$ and $d^d/dx^d\ (q^{d-1}(x)) = 0$

Let D be the maximum derivative of p(x) required. Smith's algorithm is given as:

```
begin
    for d in 0..min(D,P) loop
        y^d_{P+1} := 0·0;
        y^d_P := 0·0;
        for p in reverse d..P loop
            y^d_p := (α^{p-d}_1 x − α^{p-d}_2) y^d_{p+1} − α^{p-d+1}_3 y^d_{p+2};
            if d=0 then
                y^d_p := y^d_p + a_p;
            else
                y^d_p := y^d_p + α^{p-d}_1 y^{d-1}_p;
            end if;
        end loop;
    d^d/dx^d (p(x)) := d! y^d_d;
    end loop;
end;
```

Some simplifications may be made to the above algorithms if we restrict ourselves to the evaluation of Chebyshev polynomials. Since the recurrence coefficients $\alpha^p{}_2$ and $\alpha^p{}_3$ are zero and one respectively, the number of multiplications and additions for Clenshaw's algorithm can be reduced to P+1 and 2P, respectively.

Rice (1965) offered an heuristic argument to show that the Chebyshev form evaluated using Clenshaw's algorithm is never ill-conditioned on the interval [–1,1]. Newbery (1974) in comparing Horner's and Clenshaw's algorithms using numerical experiments found that, like the Horner scheme, the conditioning of Clenshaw's was best for low values of |x| but deteriorated rapidly as |x| approached 1. He concluded that while the Clenshaw algorithm was generally superior to Horner's this was not the case where the polynomial coefficients have uniform or strictly alternating sign.

2.2.3.3 Root Product Algorithm

A polynomial of order P expressed as a root product has the form

$p(x) = a_0 \Pi_{p=1...P} (x - x_p)$ where $p(x_p) = 0$

The root product algorithm is given by:

```
begin
    y := a0;
    for p in reverse 0..P–1 loop
        y := (x – xp) * y;
    end loop;
    p(x) := y;
end;
```

To evaluate p(x) at an arbitrary x requires only P multiplications and P additions. It thus has the same operation count as Horner's algorithm. Kronsjö (1979) in an analysis of floating point error propagation has shown that the root product algorithm is completely stable.

It should be noted that while any polynomial may be transformed into the root product form, finding the roots may be an expensive and unstable process. In such cases the transformation would be of limited practical value. Since the Lagrange family of polynomials are often defined by the positions of their roots, the root product algorithm is thus a natural method to evaluate them.

2.2.3.4 Preprocessed Coefficients

As was mentioned earlier with reference to the theorems of Belaga, it is possible to reduce the complexity of polynomial evaluations if one is permitted to preprocess the coefficients. Belaga (1958) proposed such an algorithm to evaluate polynomials of degree greater than three using $\lfloor (P+1)/2 \rfloor + 1$ multiplications and $2\lfloor P/2 \rfloor + 1$ additions. The preprocessed coefficients are, however, not unique and may even be complex for real polynomials. Pan (1959) suggested an algorithm to evaluate polynomials of degree greater than four in $\lfloor P/2 \rfloor + 2$ multiplications and P+1 additions when P is odd, and the same number of operations as Belaga's scheme when P is even. Here again the preprocessed coefficients are not unique although they are all real for real polynomials. Rabin and Winograd (1971) demonstrated how to achieve $\lfloor P/2 \rfloor + O(\log_2(n))$ multiplication or division operations and $P + O(n)$ additions using rational preprocessing of coefficients. With all of the above schemes the cost of preprocessing may be high since evaluating the preprocessed coefficients may be ill-conditioned and complex.

It should be noted that schemes involving preprocessed coefficients possess no advantage over Horner's algorithm in evaluating polynomials of degree less than 4 since
$\lfloor (P+1)/2 \rfloor + 1 \geq P \quad \text{for} \quad P < 4$

Rice (1965) performed numerical experiments to determine the conditioning of both the Belaga and Pan algorithms. He found that both of these algorithms may be severely ill-conditioned. These results have been confirmed in more rigorous studies of round-off error effects by Kronsjö (1979).

2.2.3.5 Forward Difference Algorithm

Where one requires a polynomial to be evaluated at a number of equally spaced points any of the previous algorithms considered may be used to calculate the polynomial value at each point individually. However, a more efficient method for repeatedly evaluating polynomials at constant increments is to use the forward difference algorithm.

Consider a polynomial of order P expressed with respect to the monomial basis:

$p(x) = a_0 + a_1x + a_2(x)^2 + \ldots a_P(x)^P$

$= \Sigma_{p=0\ldots P}\, a_p\,(x)^p$

We now take the first forward difference of this polynomial:

$\Delta p(x) = p(x + \delta x) - p(x)$

The value of the polynomial at a point $x + \delta x$ can thus be expressed in terms of its value at x and its first forward difference:

$p(x + \delta x) = p(x) + \Delta p(x)$

Expanding $\Delta p(x)$ in terms of its powers gives:

$\Delta p(x) = \Sigma_{p=0\ldots P}\, a_p\,((x + \delta x)^p - (x)^p)$

Using the binomial formula we can also expand $(x + \delta x)^p$ in powers of x:

$(x + \delta x)^p = (p_0)\,(\delta x)^p + (p_1)\,x\,(\delta x)^{p-1} + \ldots (p_p)\,(x)^p$

$= \Sigma_{q=0\ldots p}\,(p_q)\,(x)^q\,(\delta x)^{p-q}$

where (p_q) represent the binomial coefficients. Substituting the binomial expansion for $(x + \delta x)^p$ into the expression for $\Delta p(x)$ we obtain:

$\Delta p(x) = \Sigma_{p=0\ldots P}\, a_p\,((\Sigma_{q=0\ldots p}\,(p_q)\,(x)^q\,(\delta x)^{p-q}) - (x)^p)$

$= \Sigma_{p=0\ldots P}\, a_p\,(\Sigma_{q=0\ldots p-1}\,(p_q)\,(x)^q\,(\delta x)^{p-q})$

Notice that the first forward difference of p(x) is a polynomial of order P–1. The second forward difference $\Delta^2 p(x)$ is a polynomial of order P–2 and may be evaluated from the first forward difference in a similar manner. For a polynomial of order P the P^{th} forward difference is constant for all values of x.

Having evaluated the P forward differences of $p(x)$ we can now calculate $p(x + \delta x)$ and its P forward differences since:

$p(x + \delta x) = p(x) + \Delta p(x)$

$\Delta p(x + \delta x) = \Delta p(x) + \Delta^2 p(x)$

:

$\Delta^{P-1} p(x + \delta x) = \Delta^{P-1} p(x) + \Delta^P p(x)$

$\Delta^P p(x + \delta x) = \Delta^P p(x)$

Once the forward differences for a P^{th} order polynomial have been found the polynomial may thus be evaluated at subsequent equispaced points in only P additions. Where the time to execute one addition is lower than that for a multiplication, and the polynomial is to be evaluated at a sufficient number of points to offset the cost of calculating the forward differences, savings can be achieved over Horner's algorithm using the above method. More detailed descriptions of the forward difference algorithm applied to cubic polynomials may be found in Foley and van Dam (1983) and Newman and Sproull (1979).

2.3 Continuity Conditions

In this section we consider the conditions required to ensure continuity of field value and derivatives across the common boundaries of contiguous elements ε and ε'. In all cases we shall endeavour to express these conditions as linear constraints on the element field parameters.

Initially we deal with C^0 fields. Since continuity of field value across elements is the simplest case to deal with, both conceptually and mathematically, we use it to illustrate some of the problems associated with the mismatching of element coordinates of contiguous elements.

In the next section we deal with C^1 fields. The major problem arising here concerns the parameter to which the field derivatives are referred. If the ensemble field derivatives are taken with respect to the global coordinate system, problems may arise with the derivative field parameters assuming infinite values. If the ensemble field derivatives are taken with respect to the coordinates of either element an asymmetry is introduced with the coordinates of one element assuming special significance over the other. An elegant solution to this problem involves the introduction of a patch coordinate system at the interelement boundary. This patch coordinate does not have to be defined explicitly nor does it require specific values to be known at any point. It is simply used to relate the field derivatives at the boundary of one element to those at the same point in the other. Indeed, unless some of the element field parameters represent field derivatives at the element boundary, the patch coordinate system may remain anonymous.

C^2 fields are dealt with in the following section in much the same manner as C^1 fields. Here, as with the first derivatives, the patch coordinate concept facilitates the expression of second derivative continuity conditions as linear constraints on the element field parameters.

For the derivative continuous fields some simplification of expression and an increase in the efficiency of evaluation can be achieved through the imposition of constraints on the alignment of the patch coordinates with the element coordinates.

Consider two contiguous elements ε and ε'. Since ε and ε' are disconnected we introduce a patch coordinate system $\mathbf{w}$ at the interelement boundary in order to compare field values and derivatives between the elements. The field derivatives up to order c with respect to the element coordinate systems $\mathbf{x}$ and $\mathbf{x}'$ at the common boundary may be transformed to field derivatives with respect to the patch coordinate system if we demand that the patch coordinates are C^c at $\partial\omega^\varepsilon \cap \partial\omega^{\varepsilon'}$. Let $\mathbf{w}$ be a point on the common boundary of elements ε and ε'. Let $\chi_{(\varepsilon)}(\mathbf{w}) = \mathbf{x}$, $\mathbf{x} \in \partial\omega^\varepsilon$ be an homeomorphism mapping the patch coordinates to the element coordinates of the same point on the boundary of element ε. Let $\chi'_{(\varepsilon')}(\mathbf{w}) = \mathbf{x}'$, $\mathbf{x}' \in \partial\omega^{\varepsilon'}$ be a similar mapping to the boundary of element ε'. Assuming that the field is C^{c-1}, to enforce C^c we require that the field derivatives of order c with respect to the patch coordinates be continuous across the interelement boundary.

i.e. $\partial^c \mathbf{u}(\mathbf{x})/\partial \mathbf{w}^c = \partial^c \mathbf{u}(\mathbf{x}')/\partial \mathbf{w}^c$

where $\mathbf{x} = \chi_{(\varepsilon)}(\mathbf{w})$

and $\mathbf{x}' = \chi'_{(\varepsilon')}(\mathbf{w})$

$\forall \mathbf{w} \in \partial\omega^\varepsilon \cap \partial\omega^{\varepsilon'}$.

2.3.1 C^0 Fields

For C^0 fields the field value must be continuous across the common element boundary.

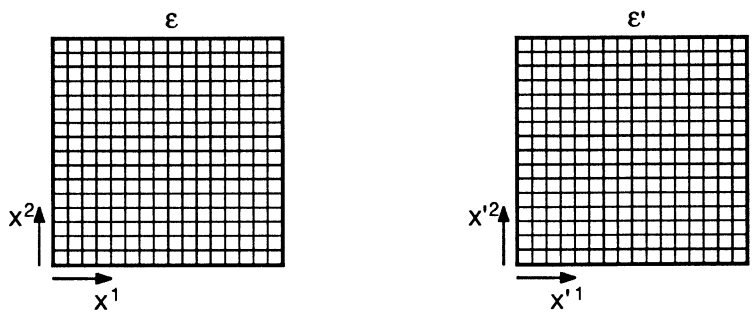
i.e. $\mathbf{u}(\mathbf{x}) = \mathbf{u}(\mathbf{x}')$

where $\mathbf{x} = \chi_{(\varepsilon)}(\mathbf{w})$

and $\mathbf{x}' = \chi'_{(\varepsilon')}(\mathbf{w})$

$\forall\ \mathbf{w} \in \partial\omega^{\varepsilon} \cap \partial\omega^{\varepsilon'}$

In this section we shall look at those C^0 continuity conditions which are able to be satisfied by linear constraints upon the ensemble field parameters. Continuity conditions at the bounds of one dimensional elements are usually trivial to formulate. Our attention will thus be concentrated upon two dimensional elements:



The simplest and most widely used method is to ensure that the field basis restricted to the common boundary is the same in both elements. The element field parameters associated with the non-trivial boundary basis functions can then be equated in both elements.

i.e. $\forall\ \beta\ \exists\ \beta' \mid \psi^{\beta(\varepsilon)}(\mathbf{x}) = \psi^{\beta'(\varepsilon')}(\mathbf{x}')$

where $\mathbf{x} = \chi_{(\varepsilon)}(\mathbf{w})$

and $\mathbf{x}' = \chi_{(\varepsilon')}(\mathbf{w})$

$\forall\ \mathbf{w} \in \partial\omega^{\varepsilon} \cap \partial\omega^{\varepsilon'}$

e.g. The matching of two elements each of which has a linear Lagrange basis along their common boundary:

for $\psi^{1(\varepsilon)}(\mathbf{x}) = 1 - x$ $\qquad$ $\psi^{1'(\varepsilon')}(\mathbf{x}') = 1 - x'$

$\psi^{2(\varepsilon)}(\mathbf{x}) = x$ $\qquad$ $\psi^{2'(\varepsilon')}(\mathbf{x}') = x'$

then $\mathbf{u}_{1(\varepsilon)} = \mathbf{u}_{1'(\varepsilon')}$

$\mathbf{u}_{2(\varepsilon)} = \mathbf{u}_{2'(\varepsilon')}$

It is not necessary to demand that the field bases match at the boundary. If a linear transformation can be made from one basis to the other we can realise the continuity constraints or linear relations between the element parameters.

e.g. The matching of elements with linear Lagrange and linear monomial bases along their common boundary:

for $\psi^{1(\varepsilon)}(\mathbf{x}) = 1 - x$ $\psi^{1'(\varepsilon')}(\mathbf{x}') = 1$

$\psi^{2(\varepsilon)}(\mathbf{x}) = x$ $\psi^{2'(\varepsilon')}(\mathbf{x}') = x'$

then $\mathbf{u}_{1(\varepsilon)} = \mathbf{u}_{1'(\varepsilon')}$

$\mathbf{u}_{2(\varepsilon)} = \mathbf{u}_{1'(\varepsilon')} + \mathbf{u}_{2'(\varepsilon')}$

Where one basis is of different order to the other at the boundary we must restrict the bases to a common one of lower order.

e.g. The matching of elements with quadratic monomial and linear Lagrange bases along their common boundary:

for $\psi^{1(\varepsilon)}(\mathbf{x}) = 1$ $\psi^{1'(\varepsilon')}(\mathbf{x}') = 1 - x'$

$\psi^{2(\varepsilon)}(\mathbf{x}) = x$ $\psi^{2'(\varepsilon')}(\mathbf{x}') = x'$

$\psi^{3(\varepsilon)}(\mathbf{x}) = (x)^2$

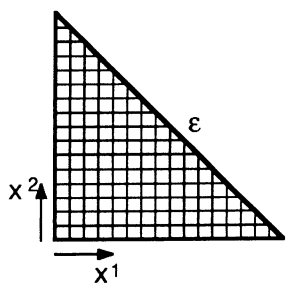
then $\mathbf{u}_{1(\varepsilon)} = \mathbf{u}_{1'(\varepsilon')}$

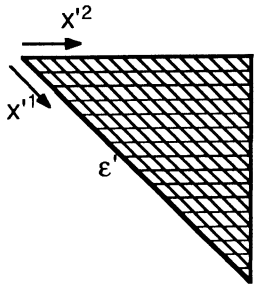
$\mathbf{u}_{2(\varepsilon)} = -\mathbf{u}_{1'(\varepsilon')} + \mathbf{u}_{2'(\varepsilon')}$

$\mathbf{u}_{3(\varepsilon)} = 0$

Where the elements don't have matching element coordinates at the boundary we must be able to transform from one to another.

e.g. The matching of two bilinear simplex elements:



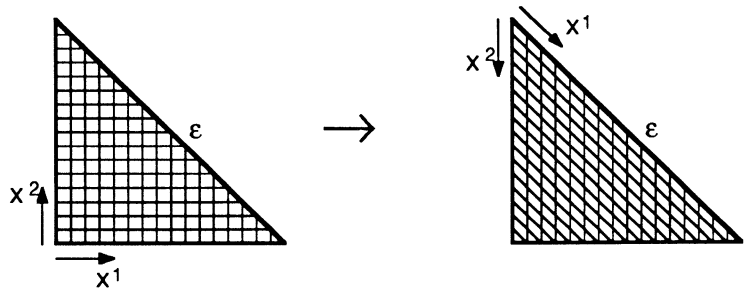


for $\psi^{1(\varepsilon)}(\mathbf{x}) = 1 - x^1 - x^2$ $\psi^{1'(\varepsilon')}(\mathbf{x}') = 1 - x'^1 - x'^2$

$\psi^{2(\varepsilon)}(\mathbf{x}) = x^1$ $\psi^{2'(\varepsilon')}(\mathbf{x}') = x'^1$

$\psi^{3(\varepsilon)}(\mathbf{x}) = x^2$ $\psi^{3'(\varepsilon')}(\mathbf{x}') = x'^2$

In order to match the coordinates along the $x^1 + x^2 = 1$ boundary in ε with the $x'^2 = 0$ boundary in ε' we can use a linear transformation of the element coordinates in ε.



$x^1 \leftarrow x^1$

$x^2 \leftarrow 1 - x^1 - x^2$

Then $\psi^{1(\varepsilon)}(\mathbf{x}) = x^2$

$\psi^{2(\varepsilon)}(\mathbf{x}) = x^1$

$\psi^{3(\varepsilon)}(\mathbf{x}) = 1 - x^1 - x^2$

On the boundary of element ε shared with element ε', denoted $\partial\varepsilon\varepsilon'$, we have $x^2 = 0$

Thus $\psi^{1(\partial\varepsilon\varepsilon')}(\mathbf{x}) = 0$

$\psi^{2(\partial\varepsilon\varepsilon')}(\mathbf{x}) = x^1$

$\psi^{3(\partial\varepsilon\varepsilon')}(\mathbf{x}) = 1 - x^1$

We can now directly map x^1 and x'^1 along this boundary. The C^0 conditions thus become

$\mathbf{u}^2 = \mathbf{u}'^2$

$\mathbf{u}^3 = \mathbf{u}'^1$

While the above examples have been rather simple, the idea of manifesting continuity conditions as linear relations or constraints upon the ensemble parameters may easily be carried over to more complicated bases. In general the sequence of operations is as follows:

(1) Transform the element coordinates to ensure that those coordinates which vary along the interelement boundary match between contiguous elements.

(2) Transform the element basis functions so that a one to one association exists between the nontrivial basis functions restricted to the common boundary in one element and those in the other.

(3) Identify the element field parameters associated with the nontrivial boundary basis functions in one element with the associated element field parameters in the other element, constraining all the unpaired element field parameters to zero.

2.3.2 C^1 Fields

For C^1 fields, as well as satisfying C^0 conditions, the first derivatives of the field with respect to the patch coordinates **w** must be continuous across the common element boundary.

i.e. $\partial \mathbf{u}(\mathbf{x})/\partial w^k = \partial \mathbf{u}(\mathbf{x}')/\partial w^k$

where $\mathbf{x} = \chi_{(\varepsilon)}(\mathbf{w})$

and $\mathbf{x}' = \chi'_{(\varepsilon')}(\mathbf{w})$

$\forall \mathbf{w} \in \partial\omega^{\varepsilon} \cap \partial\omega^{\varepsilon'}$.

The field **u** is defined within an element ε as a function of the element coordinates **x**.

Where the field derivatives are given explicitly with respect to the patch coordinates, **w**, they must first be transformed into derivatives with respect to the element coordinates:

$\partial \mathbf{u}(\mathbf{x})/\partial x^l = \partial \mathbf{u}(\mathbf{x})/\partial w^k \; \partial w^k/\partial x^l$

The term $\partial w^k/\partial x^l$ describes the relative density of the patch coordinates with respect to the element coordinates at the boundary. Where the patch coordinates are identified with the element coordinates this parameter will be equal to the delta function. If the parameter is reduced to a value below unity the element coordinates will be biassed toward the boundary.

Where we don't have $\partial \mathbf{u}(\mathbf{x})/\partial w^k$ defined explicitly at the common boundary we must express this first order continuity condition in terms of derivatives with respect to the element coordinates. While the patch coordinates are used implicitly to formulate the continuity conditions, all references to **w** must ultimately be eliminated:

$$\partial \mathbf{u}(\mathbf{x})/\partial w^k = \partial \mathbf{u}(\mathbf{x}')/\partial w^k$$

$$\partial \mathbf{u}(\mathbf{x})/\partial x^l \; \partial x^l/\partial w^k = \partial \mathbf{u}(\mathbf{x}')/\partial x'^{l'} \; \partial x'^{l'}/\partial w^k$$

or $$\partial \mathbf{u}(\mathbf{x})/\partial x^l = \partial \mathbf{u}(\mathbf{x}')/\partial x'^{l'} \; \partial x'^{l'}/\partial x^l$$

The term $\partial x'^{l'}/\partial x^l$ describes the relative density of the element coordinates in element ε' with respect to the element coordinates in element ε. Decreasing the value of this parameter biasses the element coordinates in element ε toward the boundary and those in element ε' away from the boundary.

The use of patch coordinates to ensure continuity of fields across element boundaries is a new concept. Traditionally all field derivatives have been expressed with respect to the element coordinates. Implicit in the traditional formulation is the assumption that the densities of element coordinates are continuous across interelement boundaries. This poses severe restrictions on the relative sizes of contiguous elements in order to avoid singularities in the element coordinate systems. The patch coordinate concept removes these restrictions while providing a greater degree of control of the ensemble field.

It is often useful to associate the element coordinates with a global coordinate system identified with a physically meaningful quantity. Where possible, the arc length of global coordinate curves following the element coordinates provide a practical parameter with which to determine the different element coordinate densities. de Boor (1978) has argued that parameterising with respect to arc length usually leads to good results. While he was concerned only with one dimensional spline curves this strategy may be readily extended to higher dimensional representations and other bases.

2.3.3 C^2 Fields

A C^2 field must be C^1 and in addition the second derivative of the field with respect to the patch coordinates must be continuous across the boundary.

i.e. $\partial^2\mathbf{u}(\mathbf{x})/\partial w^{k_1}\partial w^{k_2} = \partial^2\mathbf{u}(\mathbf{x}')/\partial w^{k_1}\partial w^{k_2}$

where $\mathbf{x} = \chi_{(\varepsilon)}(\mathbf{w})$

and $\mathbf{x}' = \chi'_{(\varepsilon')}(\mathbf{w})$

$\forall \mathbf{w} \in \partial\omega^{\varepsilon} \cap \partial\omega^{\varepsilon'}$.

We shall firstly consider the second derivative continuity conditions, where the derivatives are expressed explicitly with respect to the patch coordinates, **w**. The derivatives with respect to the patch coordinates must be transformed into derivatives with respect to the element coordinates:

$$\partial^2\mathbf{u}(\mathbf{x})/\partial x^{l_1}\partial x^{l_2} = \partial^2 w^k/\partial x^{l_1}\partial x^{l_2}\ \partial\mathbf{u}(\mathbf{x})/\partial w^k + \partial w^{k_1}/\partial x^{l_1}\ \partial w^{k_2}/\partial x^{l_2}\ \partial^2\mathbf{u}(\mathbf{x})/\partial w^{k_1}\partial w^{k_2}$$

The terms $\partial w^k/\partial x^l$ describe the relative density of the patch coordinates with respect to the element coordinates. Decreasing the value of this parameter biases the element coordinates toward the boundary. The term $\partial^2 w^k/\partial x^{l_1}\partial x^{l_2}$ describes the rate of change of the density of patch coordinates with respect to the element coordinates. Altering this parameter, called the tension, changes the rate of change of the bias.

Where the field derivatives are not expressed explicitly with respect to patch coordinates the second order continuity conditions can be deduced by firstly adopting an implicit patch coordinate system, **w**:

$$\partial^2\mathbf{u}(\mathbf{x})/\partial w^{k_1}\partial w^{k_2} = \partial^2\mathbf{u}(\mathbf{x}')/\partial w^{k_1}\partial w^{k_2}$$

Rearranging the above to eliminate all references to **w** we obtain:

$$\partial^2\mathbf{u}(\mathbf{x})/\partial x^{l_1}\partial x^{l_2} = \partial^2\mathbf{u}(\mathbf{x}')/\partial x'^{l'_1}\partial x'^{l'_2}\ \partial x'^{l'_1}/\partial x^{l_1}\ \partial x'^{l'_2}/\partial x^{l_2} + \partial\mathbf{u}(\mathbf{x}')/\partial x'^{l'_3}\ \partial^2 x'^{l'_3}/\partial x^{l_1}\partial x^{l_2}$$

The terms $\partial x'^{l'}/\partial x^l$ describe the relative density of the element coordinates in element ε' with respect to the element coordinates in element ε. Decreasing the value of this parameter biases the element coordinates in element ε toward the boundary and those in element ε' away from the boundary. The term $\partial^2 x'^{l'_3}/\partial x^{l_1}\partial x^{l_2}$ describes the rate of change of the density

of the element coordinates in element ε' with respect to the element coordinates in element ε. Altering this tension parameter changes the rate of change of the the bias across the interelement boundary. Bias and tension parameters in the implicit patch coordinate formulation have been discussed in Barsky and Beatty (1983) with reference to cubic splines. The bias parameter will be considered further in sections 2.4.2 and 2.4.3 when we deal with the ensemble to element parameter map.

2.4 Ensemble To Element Parameter Map

The ensemble to element parameter map associates each element parameter, $\mathbf{u}^{\beta}$, with a linear combination of ensemble parameters, $\mathbf{U}^{B}$.

$$\mathbf{u}^{\beta(\varepsilon)} = \upsilon^{\beta(\varepsilon)}{}_{B}\, \mathbf{U}^{B}$$

Traditionally this association has been viewed as a Boolean map where each element parameter is logically associated with one and only one ensemble parameter. This view is conceptually simple but is too restrictive. It works well in describing C^0 fields by simple ensembles using a single basis type but fails to accommodate general fields with higher order continuity or ensembles constructed from dissimilar bases.

We propose that the ensemble to element parameter map, $\upsilon^{\beta(\varepsilon)}{}_{B}$, be extended to general linear maps between $\mathbf{u}^{\beta}$ and $\mathbf{U}^{B}$. Several interesting consequences arise from firstly allowing real valued coefficients for $\upsilon^{\beta(\varepsilon)}{}_{B}$ and secondly allowing a many to many mapping between $\mathbf{u}^{\beta}$ and $\mathbf{U}^{B}$.

If the ensemble to element parameter map is allowed to have real coefficients, fields with higher order continuity are more readily dealt with. The field derivatives at the element boundaries may now be defined with respect to a patch coordinate aligned to the element coordinates. The density of element coordinates may now change across the interelement boundary without sacrificing continuity of field derivatives.

When the ensemble to element parameter map is free to be a general linear map we are freed from requiring that each element parameter be associated with only one ensemble parameter. Field derivatives at element boundaries may be described with respect to general patch coordinates, no longer constrained to lie along the directions of the element coordinates. Ensembles constructed from dissimilar bases may also be accommodated using this approach.

In the following we shall consider the Boolean map, the real valued map and the general linear map in more detail.

2.4.1 Boolean Map

We consider here the case where the ensemble to element parameter map is viewed as a Boolean map. In other words $\upsilon^{\beta(\varepsilon)_B}$ characterises the connection between the ensemble field parameters, $\mathbf{U}^B$, and the element field parameters, $\mathbf{u}^{\beta(\varepsilon)}$, as a purely logical relation.

This view of the ensemble to element parameter map prevails because the ensemble field parameters, $\mathbf{U}^B$,are usually assigned to specific positions within the ensemble, termed ensemble nodes. Similarly the element field parameters are assumed to reside at specific element positions called element nodes.

A one to one mapping is established between each element basis function and each element node as well as between each ensemble basis function and each ensemble node.

Each element node is then logically associated with a unique ensemble node. This association is isomorphic to the ensemble to element parameter map, $\upsilon^{\beta(\varepsilon)_B}$. Note that element nodes from distinct elements may be associated with the same ensemble node but that, under this scheme, each element node may be associated with one and only one ensemble node.

The basis functions are usually defined as having a value of unity at a unique node and to vanish at all other nodes. In this case the field value at the node will be equal to the field parameter associated with that node. Simplex bases, serendipity bases and basis functions defined as tensor products of Lagrange polynomials fit naturally into this scheme.

The Boolean map associates each element node with its unique ensemble node thus establishing both the ensemble topology (for any element, those elements contiguous with it) as well as the unique ensemble field parameter connected to each element field parameter.

Several criticisms may be levelled at this approach:

(1) The field parameters should not be assigned to unique points. The effective region for each field parameter is by definition the support of the associated basis function;

(2) Characterisation of the ensemble topology should be separated from the field definition;

(3) The ensemble to element parameter map should not be restricted to one-to-many Boolean functions. The essential quality of retaining linearity of the field with respect to the ensemble field parameters may be retained while using more general linear maps.

Consider the case where the ensemble to element map is characterised by identifying those element nodes coincident with ensemble nodes. Unless the element nodes are allowed to lie outside the element the support of the ensemble basis function, associated with its unique ensemble field parameter resident at a node, may only extend over those elements coincident at that node. In this case basis functions with support greater than two elements on a simple one dimensional mesh are unable to be accommodated.

2.4.2 Real Valued Map

Real valued coefficients in the ensemble to element parameter map, $\upsilon^{\beta(\varepsilon)}{}_{B}$, allow each of the element field parameters, $\mathbf{u}^{\beta(\varepsilon)}$, to vary linearly with its associated ensemble field parameter, $\mathbf{U}^{B}$. This is particularly useful where $\mathbf{U}^{B}$ represents derivative information about the field.

Consider now a one dimensional cubic Hermite mesh. The field within a element is characterised by specifying the field value and derivatives with respect to the element coordinate, **x**, at each element boundary. It is possible to construct a basis of four cubic polynomials where for each function one of the function values and derivatives at the two boundaries is non zero, taking the value of unity. The field within the element is thus given by :

$$\mathbf{u}(\mathbf{x}) = \psi_{\beta(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{\beta(\varepsilon)}$$

$$= \psi_{1(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{1(\varepsilon)} + \psi_{2(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{2(\varepsilon)} + \psi_{3(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{3(\varepsilon)} + \psi_{4(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{4(\varepsilon)}$$

where $\mathbf{u}^{1(\varepsilon)} = \mathbf{u}^{(\varepsilon)}|_{x=0}$

$\mathbf{u}^{2(\varepsilon)} = \partial\mathbf{u}^{(\varepsilon)}/\partial x|_{x=0}$

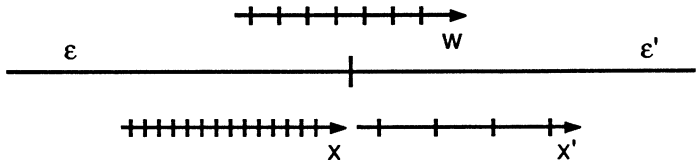
$\mathbf{u}^{3(\varepsilon)} = \mathbf{u}^{(\varepsilon)}|_{x=1}$

$\mathbf{u}^{4(\varepsilon)} = \partial\mathbf{u}^{(\varepsilon)}/\partial x|_{x=1}$

Note that the element field parameters associated with the two derivative basis functions represent derivatives with respect to the element coordinate **x**.

The motivation for using cubic Hermite bases is to ensure first order continuity of the ensemble field across interelement boundaries. This is usually achieved by requiring that the field derivatives with respect to the element coordinates at the common boundary of contiguous elements be equal. However, this condition is too restrictive since we may retain first order continuity of the ensemble field without demanding that the field derivatives with respect to the element coordinates be continuous. We do, in fact, require that the field derivative with respect to the element coordinates multiplied by the density of element coordinates be continuous across the interelement boundaries. In this case the density of element coordinates may be discontinuous across the interelement boundaries.

These derivative scaling factors free us from expressing the ensemble derivative parameters with respect to the element coordinates, allowing them to be described with respect to some auxilary or patch coordinate **w**. The ensemble derivative parameters are thus expressed as $\partial \mathbf{U}^{B}/\partial \mathbf{w}$ and are transformed to the element derivative parameters $\partial \mathbf{u}^{\beta(\varepsilon)}/\partial \mathbf{x}$ via the ensemble to element parameter map. In this case $\upsilon^{\beta(\varepsilon)}{}_{B}$, where $\beta(\varepsilon)$ and B are indices for derivative parameters, represents $\partial \mathbf{w}/\partial \mathbf{x}$.



It is sometimes useful to associate the patch coordinate **w** with a physically meaningful quantity such as the arc length of a curve. Notice that **w** need never be known explicitly since only **dw** is used. It is possible to transform **dw** to be equal to an increment of arc length through a simple rescaling of the derivative ensemble field parameters by $(g_{jj'}\, \partial y^{j}/\partial \mathbf{w}\, \partial y^{j'}/\partial \mathbf{w})^{-1/2}$ and the corresponding components of the ensemble to element parameter map by $(g_{jj'}\, \partial y^{j}/\partial \mathbf{w}\, \partial y^{j'}/\partial \mathbf{w})^{1/2}$ where y^{j} are the global curvilinear coordinates and $g_{jj'}$ are the covariant components of the associated metric tensor.

2.4.3 General Linear Map

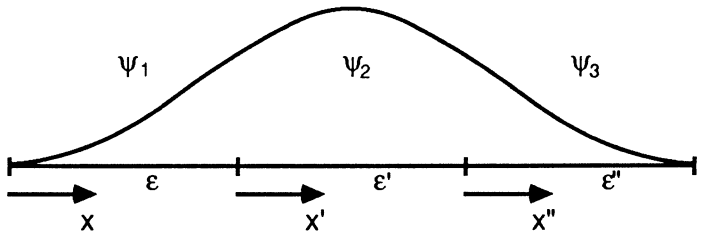
By allowing a many to many map between the element field parameters, $\mathbf{u}^{\beta(\varepsilon)}$, and the ensemble field parameters, $\mathbf{U}^{B}$, we can free ourselves from the assumption that each element parameter is dependent upon one and only one ensemble parameter. This is useful where we wish to reduce the number of independent ensemble variables for a given problem or where there may not exist a convenient basis to satisfy the field continuity conditions.

Consider now a one dimensional quadratic spline. When this spline is formulated upon an even mesh we get a single basis common to all elements:

$\psi_1(\mathbf{x}) = 1/2 - x + (x)^2/2$

$\psi_2(\mathbf{x}') = 1/2 + x' - (x')^2$

$\psi_3(\mathbf{x}'') = \quad (x'')^2/2$



However, if we wish to alter the element coordinate density across element boundaries we will no longer have a unique basis for all elements. In this case we will either have to define separate bases for each element or make use of the general linear ensemble to element parameter map.

Let $s^{\varepsilon'}_{\varepsilon} = \partial x'^{(\varepsilon')}/\partial x^{(\varepsilon)}$ represent $(\partial x'^{(\varepsilon')}/\partial w)/(\partial x^{(\varepsilon)}/\partial w)$, the ratio of element coordinate density across the common boundary of elements ε and ε'. Similarly, let $s^{\varepsilon''}_{\varepsilon'} = \partial x''^{(\varepsilon'')}/\partial x'^{(\varepsilon')}$. The basis functions for the quadratic spline may be evaluated with the aid of a symbolic manipulator (Hearn 1973) using a recurrence relation described in de Boor (1978) and Schumaker (1981):

$$\psi_1(\mathbf{x}) = (s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)) - 2x(s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)) + (x)^2(s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon))$$

$$\psi_2(\mathbf{x}) = (1/(1 + s\varepsilon'_\varepsilon)) + 2x'(s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)) - (x')^2(s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon) + 1/(1 + s\varepsilon''_{\varepsilon'}))$$

$$\psi_3(\mathbf{x}) = + (x'')^2(1/(1 + s\varepsilon''_{\varepsilon'}))$$

Note that since the polynomial coefficients for the basis functions in element ε vary with $s\varepsilon'_\varepsilon$ and $s\varepsilon''_{\varepsilon'}$ we no longer have a single basis sufficient for all elements. When there is no discontinuity of element coordinate density across the interelement boundary the parameters $s\varepsilon'_\varepsilon$ and $s\varepsilon''_{\varepsilon'}$ will both be 1·0 and the basis functions revert to those of the even mesh.

Rather than generate a separate basis for each combination of $s\varepsilon'_\varepsilon$ and $s\varepsilon''_{\varepsilon'}$ (there may be one required for every element) it is more economical to respecify our basis as being simply the powers of x up to the order of the basis polynomials. With the element bases set to the monomial basis, $\psi_1(\mathbf{x}) = 1$, $\psi_2(\mathbf{x}) = x$, $\psi_3(\mathbf{x}) = (x)^2$, the ensemble to element parameter map, $\upsilon^{\beta(\varepsilon)}{}_B$, become the coefficients in the polynomial expansion of the basis functions above. Thus

$$\upsilon^{1(\varepsilon)}{}_B = s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)$$

$$\upsilon^{2(\varepsilon)}{}_B = -2s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)$$

$$\upsilon^{3(\varepsilon)}{}_B = s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)$$

$$\upsilon^{1(\varepsilon')}{}_B = 1/(1 + s\varepsilon'_\varepsilon)$$

$$\upsilon^{2(\varepsilon')}{}_B = 2s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon)$$

$$\upsilon^{3(\varepsilon')}{}_B = -(s\varepsilon'_\varepsilon/(1 + s\varepsilon'_\varepsilon) + 1/(1 + s\varepsilon''_{\varepsilon'}))$$

$$\upsilon^{1(\varepsilon'')}{}_B = 0$$

$$\upsilon^{2(\varepsilon'')}{}_B = 0$$

$$\upsilon^{3(\varepsilon'')}{}_B = 1/(1 + s\varepsilon''_{\varepsilon'})$$

These coefficients will, in general, be different for each element since $s\varepsilon'_\varepsilon$ and $s\varepsilon''_{\varepsilon'}$ may vary from element to element. The coefficients of the ensemble to element parameter map for higher order splines can be derived in a similar manner.

Let us now consider the use of derivative scaling factors for elements of dimension greater than one. Unless we have a degenerate element the number of patch coordinates, **w**, associated with an element, ε, will be greater than or equal to the element dimension,

dim(ω^ε). In this case **w** can be viewed as a local coordinate system defined implicitly about the neighbourhood of a point on the ensemble.

First derivatives with respect to **w** at the ensemble level may be readily transformed into derivatives with respect to **x** at the element level using the chain rule:

$\partial/\partial x^l = \partial w^k/\partial x^l \; \partial/\partial w^k$

If the derivative ensemble parameters are carried as derivatives with respect to **w**

i.e. $\mathbf{U}^B = \partial \mathbf{u}/\partial w^k|_{\mathbf{x}}$, $\mathbf{x} \in \omega^\varepsilon$, for some B

then since $\mathbf{u}^{\beta(\varepsilon)} = \upsilon^{\beta(\varepsilon)}{}_B \, \mathbf{U}^B$ we can identify $\upsilon^{\beta(\varepsilon)}{}_B$ with $\partial w^k/\partial x^l$.

Second derivatives with respect to **w** may be similarly transformed into derivatives with respect to **x**:

$\partial^2/\partial x^l \partial x^{l'} = \partial^2 w^k/\partial x^l \partial x^{l'} \; \partial/(\partial w^k)^2 + \partial w^k/\partial x^l \; \partial w^k/\partial x^{l'} \; \partial^2/\partial w^k \partial w^{k'}$

Notice that all of the first and second derivatives of **w** with respect to **x** are required to transform derivatives of a field with respect to **w** into second derivatives with respect to **x**.

Higher derivatives may be evaluated in a similar manner.

Some savings can be made if the patch coordinates are aligned along the elements. Let $\{\pi(l)\}$ be a permutation of $\{l\}$ such that $w^{\pi(l)}$ is aligned along x^l. Then $\partial w^k/\partial x^l$ is non-zero only when $k = \pi(l)$. The first derivative transformation reduces to

$\partial/\partial x^l = \partial w^{\pi(l)}/\partial x^l \; \partial/\partial w^{\pi(l)}$ (no sum on $\pi(l)$)

Similarly the second derivative transformation becomes

$$\partial^2/\partial x^l \partial x^{l'} = \begin{cases} \partial^2 w^{\pi(l)}/\partial x^l \partial x^{l'} \; \partial/\partial w^{\pi(l)} + (\partial w^{\pi(l)}/\partial x^l)^2 \; \partial^2/(\partial w^{\pi(l)})^2 & l = l' \text{ (no sum on } \pi(l)) \\ \partial w^{\pi(l)}/\partial x^l \; \partial w^{\pi(l')}/\partial x^{l'} \; \partial^2/\partial w^{\pi(l)} \partial w^{\pi(l')} & l \neq l' \text{ (no sum on } \pi(l) \text{ or } \pi(l')) \end{cases}$$

The first term of the transformation for $l \neq l'$ is absent since

$\partial/\partial x^{l'}(\partial w^{\pi(l)}/\partial x^l) = 0 \qquad l \neq l'$

2.5 Ensemble Basis Functions

In this section we consider the effect, upon the ensemble basis functions, of introducing interelement continuity conditions in the form of linear constraints between the field parameters. Recall that associated with each ensemble parameter is an ensemble basis function, $\Psi_B(\mathbf{X}) = \cup_\varepsilon \psi_{\beta(\varepsilon)}(\mathbf{x})\, \upsilon^{\beta(\varepsilon)}{}_B$, where $\mathbf{X} \in \Omega^E$, $\mathbf{x} \in \omega^\varepsilon$. Each ensemble basis function represents the weight that the associated ensemble field parameter $\mathbf{U}^B$ contributes to the field at the ensemble position, $\mathbf{X}$. The collection of ensemble basis functions thus represents a basis for fields defined on Ω^E.

Let us now consider an ensemble with a fixed number of elements. Each element has a fixed number of element field parameters associated with it, although the actual number may vary from element to element. Note that since for each element the number of element field parameters, $\Sigma_{\beta(\varepsilon)}1$, is constant and since the number of elements in the ensemble is fixed then $\Sigma_\varepsilon\, \Sigma_{\beta(\varepsilon)}1$ is also constant, being equal to the number of degrees of freedom used to characterise an uncoupled field. Since each continuity condition reduces the number of ensemble field parameters, and thus the number of ensemble basis functions, by one we have the relation:

$$\Sigma_B 1 = \Sigma_\varepsilon\, \Sigma_{\beta(\varepsilon)}1 - \Sigma_c 1$$

where $\Sigma_c 1$ represents the total number of linear constraints on the ensemble field.

Each interelement continuity condition effectively increases the support of one of the ensemble basis functions by one. This is a global phenomenon in the sense that the support of individual basis functions may increase or decrease but

$$\Sigma_B \text{supp}(\Psi_B) = \Sigma_\varepsilon\, \Sigma_{\beta(\varepsilon)} \text{supp}(\psi_{\beta(\varepsilon)})$$
$$= \Sigma_\varepsilon\, \Sigma_{\beta(\varepsilon)}1$$

remains constant. It can thus be seen that the introduction of continuity conditions upon a field results in a decrease in the number of ensemble field parameters and an increase in the support of some of the ensemble functions.

If there are no interelement continuity conditions the element field parameters of one element will be independent of any others in the ensemble. In this case the number of

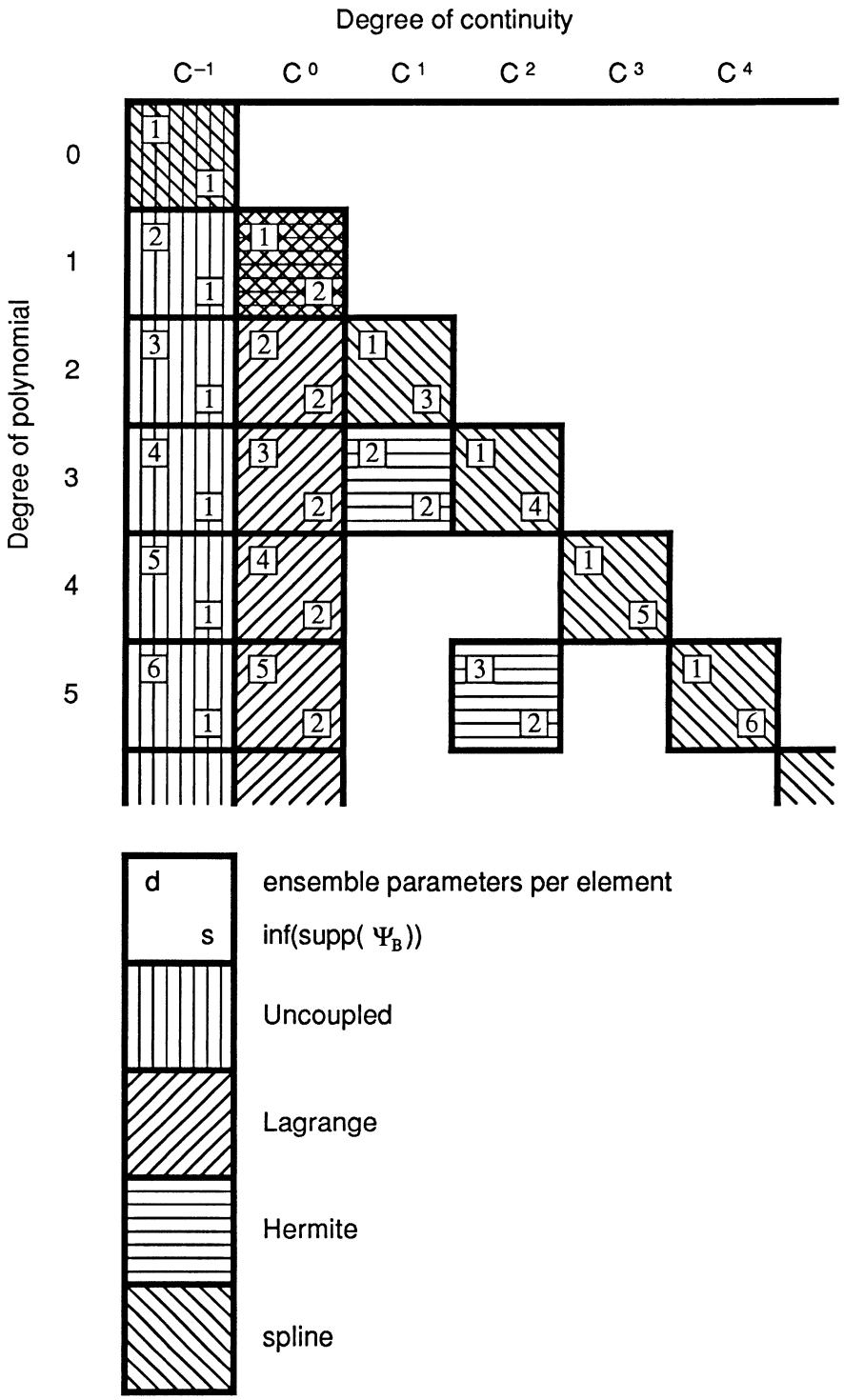
independent ensemble parameters will be $\Sigma_\varepsilon \Sigma_{\beta(\varepsilon)}1$. For a completely uncoupled mesh supp(Ψ_B) will be one element for all ensemble field parameters, B. In general, for a one dimensional mesh with no interelement continuity conditions the greatest lower bound for supp(Ψ_B), inf(supp(Ψ_B)), is one element.

If we require our field to be C^0 we must ensure that the field defined in one element is equal to the field defined in contiguous elements at their common boundaries. We shall assume here that any interelement continuity condition may be realised as a linear relation between the element field parameters of the elements concerned. Also, since the element field parameters are related linearly to the ensemble field parameters via the ensemble to element parameter map, $\mathbf{u}^{\beta(\varepsilon)} = \upsilon^{\beta(\varepsilon)}{}_B\, \mathbf{U}^B$, the nett effect of introducing a continuity condition is to reduce the number of independent ensemble field parameters by one. For a C^0 field defined on a simple one dimensional ensemble of E elements of order P, (E–1) degrees of freedom are required for continuity conditions leaving (PE–1) linearly independent ensemble field parameters. In this case inf(supp(Ψ_B)) is two elements although some of the basis functions may have one element support. The Lagrange family are an example of such basis functions.

The Hermite family of basis functions results from explicitly specifying boundary values and derivatives up to order (P–1)/2 for polynomials of odd degree, P. Since the field value and derivative information is shared between contiguous elements the ensemble field will be $C^{(P-1)/2}$. For a $C^{(P-1)/2}$ mesh on a simple one dimensional ensemble of E elements (E–1)(P+1)/2 degrees of freedom are required to satisfy the continuity conditions leaving (E+1)(P+1)/2 linearly independent ensemble field parameters for the mesh or (P+1)/2 ensemble parameters per element. For the Hermite family inf(supp(Ψ_B)) is two elements.

With piecewise polynomials of degree P it is possible to generate $C^{(P-1)}$ fields using the spline family. Here field values at the interelement boundary are implicitly shared. Thus (E–1)P ensemble degrees of freedom for a simple one dimensional mesh of E elements are required to satisfy the boundary conditions. The remaining E+P ensemble degrees of freedom are used to define the field. There is therefore one independent ensemble field

parameter per element. For b-splines of order P inf(supp(Ψ_B)) is (P+1) elements. The above observations are summarised in the following diagram:



2.6 Applications

In this section we consider the implications of using a general linear map for the ensemble to element field parameter map. In particular, we shall stress the advantages of using the monomial basis for all element bases.

Consider a field defined by piecewise polynomials on one dimensional elements. Let the bases in all elements be the monomial basis:

$\psi_{1(\varepsilon)}(\mathbf{x}) = 1 \qquad \psi_{2(\varepsilon)}(\mathbf{x}) = x \qquad \ldots \qquad \psi_{P+1(\varepsilon)}(\mathbf{x}) = (x)^P$

Here we identify $\upsilon^{\beta(\varepsilon)}{}_B\ \mathbf{U}^B$ with the polynomial power coefficients:

$$\mathbf{u}_{(\varepsilon)}(\mathbf{x}) = \psi_{\beta(\varepsilon)}(\mathbf{x})\ \upsilon^{\beta(\varepsilon)}{}_B\ \mathbf{U}^B$$

$$= \psi_{\beta(\varepsilon)}(\mathbf{x})\ \mathbf{u}^{\beta(\varepsilon)} \qquad \text{where } \mathbf{u}^{\beta(\varepsilon)} = \upsilon^{\beta(\varepsilon)}{}_B\ \mathbf{U}^B$$

$$= \mathbf{u}^{1(\varepsilon)} + x\,\mathbf{u}^{2(\varepsilon)} + (x)^2\,\mathbf{u}^{3(\varepsilon)} + \ldots (x)^P\,\mathbf{u}^{P+1(\varepsilon)}$$

It should be noted that once we have made the transformation from the ensemble field parameters, $\mathbf{U}^B$, to the element field parameters, $\mathbf{u}^{\beta(\varepsilon)}$, via the ensemble to element parameter map, $\upsilon^{\beta(\varepsilon)}{}_B$, we are able to efficiently evaluate the field, $\mathbf{u}_{(\varepsilon)}(\mathbf{x})$, at any number of arbitrary positions within the element using Horner's algorithm. Horner's algorithm is optimal with respect to the number of multiplications and additions for the evaluation of polynomials without preprocessed coefficients, requiring only P multiplications and P additions to evaluate a one dimensional polynomial of degree P. It should be noted that while it is expensive to transform from $\mathbf{U}^B$ to $\mathbf{u}^{\beta(\varepsilon)}$ it is cheap to evaluate $\mathbf{u}_{(\varepsilon)}(\mathbf{x})$. Clearly the greatest savings will be made where many field evaluations within each element are required.

The monomial basis possesses a further advantage over most other bases in that the product of two monomial basis functions is itself a monomial basis function. This feature may be exploited to reduce the number of multiplications required to evaluate field inner and outer products.

In the following subsections we provide a few examples of how our formalism may be applied to the evaluation of field outer products, the solution of the Poisson equation using

the finite element method, and the use of the general linear map to provide a useful strategy for mesh refinement. In the first two cases special attention is given to the monomial basis.

2.6.1 Outer Products

Having the element basis functions as individual powers of x can reduce the effort required to evaluate outer products of the basis functions or their derivatives. Consider the evaluation of the metric tensor:

$$g_{ll'} = \partial z^n/\partial x^l \;\; \partial z^n/\partial x^{l'}$$

$$= \partial z^n/\partial y^m \;\; \partial y^m/\partial x^l \;\; \partial z^n/\partial y^{m'} \;\; \partial y^{m'}/\partial x^{l'}$$

where **z** represents a global rectangular Cartesian coordinate system and **y** is a general global curvilinear coordinate system.

Now $y^m = \psi_{\beta(\varepsilon)}(x)\; y^{m\beta(\varepsilon)}$

where $y^{m\beta(\varepsilon)} = \upsilon^{\beta(\varepsilon)}{}_B \; Y^{mB}$

and $\partial y^m/\partial x^l = \psi_{\beta(\varepsilon),l}(x)\; \upsilon^{\beta(\varepsilon)}{}_B \; Y^{mB}$

so $g_{ll'} = \psi_{\beta(\varepsilon),l}\, \psi_{\beta'(\varepsilon),l'}\, y^{m\beta(\varepsilon)}\, y^{m'\beta'(\varepsilon)} \;\; \partial z^n/\partial y^m \;\; \partial z^n/\partial y^{m'}$

At this point we shall restrict ourselves to one dimensional bases, i.e. where $\dim(\omega^\varepsilon) = 1$. Notice now that if

$\psi_{\beta(\varepsilon)} = (x)^{\beta-1} \quad \beta = 1...P+1$

then

$\psi_{\beta(\varepsilon),l}\, \psi_{\beta'(\varepsilon),l'} = (\beta-1)(\beta'-1)(x)^{(\beta-2)(\beta'-2)}$

Making use of symmetry and ignoring the coefficients occurring for zero values of $(\beta-1)$ or $(\beta'-1)$ it can be seen that the evaluation of $\psi_{\beta(\varepsilon),l}\, \psi_{\beta'(\varepsilon),l'}$ can be performed in $2(P-1)$ additions and $2(P-1) + P(P-1)$ multiplications. Contrast this where $\psi_{\beta(\varepsilon)}$ is a general polynomial basis function of degree P. Here each evaluation of $\psi_{\beta(\varepsilon)}$ requires P multiplications and P additions using Horner's algorithm and $\psi_{\beta(\varepsilon),l}\, \psi_{\beta'(\varepsilon),l'}$ requires $P(P+1)/2$ basis function evaluations so the outer product requires $P^2(P+1)/2$ multiplications and additions.

P:	$P^2(P+1)/2$:	$2(P-1) + P(P-1)$:
1	1	0
2	6	4
3	18	10
4	40	18
5	75	28

2.6.2 Poisson Equation

In this section we demonstrate how the formalism that we have introduced may be applied to the finite element method. Consider the Poisson equation:

$-\nabla^2 \mathbf{u} = \mathbf{h}$ on Ω

$\partial \mathbf{u}/\partial \mathbf{n} = \mathbf{u_n}$ on $\partial\Omega$

We apply the Galerkin weighted residual method to the above equation using a weighting function v:

$$0 = \int_\Omega (\nabla^2 \mathbf{u} + \mathbf{h}) v \, \mathbf{d}\Omega$$

$$= -\int_\Omega \nabla \mathbf{u}.\nabla v \, \mathbf{d}\Omega + \int_\Omega \mathbf{h} v \, \mathbf{d}\Omega + \int_{\partial\Omega} \mathbf{u_n} \, v \, \mathbf{d}(\partial\Omega)$$

Thus

$$\int_\Omega \partial \mathbf{u}/\partial z^n \, \partial v/\partial z_n \, \mathbf{dz} = \int_\Omega \mathbf{h} v \, \mathbf{dz} + \int_{\partial\Omega} \mathbf{u_n} \, v \, \mathbf{d}(\partial\Omega)$$

Now let $\mathbf{u} = \psi_{\beta(\varepsilon)}(\mathbf{x}) \, \upsilon^{\beta(\varepsilon)}{}_B \, \mathbf{U}^B$

and $v = \psi_{\beta'(\varepsilon)}(\mathbf{x}) \, \upsilon^{\beta'(\varepsilon)}{}_{B'}$

$= \Psi_{B'}(\mathbf{X})$

so that v represents an ensemble basis function. We simplify the representation of derivatives of the element basis function, $\psi_{\beta(\varepsilon)}$, with respect to the element coordinate, x^l, setting:

$\psi_{\beta(\varepsilon),l} = \partial \psi_{\beta(\varepsilon)}/\partial x^l$

The integrals expressed with respect to the global coordinate system, **z**, must be transformed into integrals expressed with respect to the element coordinate system, **x**. Thus

$\int_\Omega \sim \mathbf{dz} = \int_\Omega \sim J \, \mathbf{dx}$

where J is the determinant of the Jacobian of the transformation from **z** to **x**:

$J = (\det(g_{ll'}))^{1/2}$

$g^{ll'} = \partial x^l/\partial z^n \, \partial x^{l'}/\partial z_n$

The integral equation becomes:

$$[\int_\Omega g^{ll'} \psi_{\beta(\varepsilon),l} \psi_{\beta'(\varepsilon),l'} J \, \mathbf{dx}] \upsilon^{\beta(\varepsilon)}{}_B \upsilon^{\beta'(\varepsilon)}{}_{B'} \mathbf{U}^B = [\int_\Omega \mathbf{h} \psi_{\beta'(\varepsilon)} J \, \mathbf{dx}] \upsilon^{\beta'(\varepsilon)}{}_{B'} + [\int_{\partial\Omega} \mathbf{u_n} \psi_{\beta'(\varepsilon)} \, \mathbf{d}(\partial\Omega)] \upsilon^{\beta'(\varepsilon)}{}_{B'}$$

The integrals may now be evaluated element by element to form the element coefficient matrix:

$\mathbf{e}_{\beta(\varepsilon)\,\beta'(\varepsilon)} = \int_\Omega g^{ll'}\, \psi_{\beta(\varepsilon),l}\, \psi_{\beta'(\varepsilon),l'}\, J\, \mathbf{dx}$

and element right hand side:

$\mathbf{f}_{\beta'(\varepsilon)} = \int_\Omega \mathbf{h}\, \psi_{\beta'(\varepsilon)}\, J\, \mathbf{dx} + \int_{\partial\Omega} \mathbf{u}_n\, \psi_{\beta'(\varepsilon)}\, \mathbf{d}(\partial\Omega)$

The element coefficient matrices and right hand sides are assembled using the ensemble to element parameter map, $\upsilon^{\beta(\varepsilon)}{}_B$:

$[\, \mathbf{e}_{\beta(\varepsilon)\,\beta'(\varepsilon)}\,]\, \upsilon^{\beta(\varepsilon)}{}_B\, \upsilon^{\beta'(\varepsilon)}{}_{B'}\, \mathbf{U}^B = [\, \mathbf{f}_{\beta'(\varepsilon)}\,]\, \upsilon^{\beta'(\varepsilon)}{}_{B'}$

to form the ensemble coefficient matrix and right hand side:

$\mathbf{E}_{B\,B'}\, \mathbf{U}^B = \mathbf{F}_{B'}$

Let us now consider the operation count in the evaluation of the outer product within the element coefficient matrix, $\mathbf{e}_{\beta(\varepsilon)\,\beta'(\varepsilon)}$. For simplicity we shall assume that the problem is one dimensional. Let the element basis functions be represented by monomials up to order P:

$\psi_{\beta(\varepsilon)}(x) = (x)^{\beta-1} \qquad \beta = 1...P+1$

$\psi_{\beta(\varepsilon),l}(x) = (\beta-1)(x)^{\beta-2}$

and

$\psi_{\beta(\varepsilon),l}\, \psi_{\beta'(\varepsilon),l'} = (\beta-1)(\beta'-1)(x)^{(\beta+\beta'-4)}$

Noting that

$\mathbf{e}_{\beta(\varepsilon)\,\beta'(\varepsilon)} = 0 \qquad$ for $\beta = 1$ or $\beta' = 1$

to form $\mathbf{e}_{\beta(\varepsilon)\,\beta'(\varepsilon)}$ we need only evaluate

$\int_\Omega g^{ll'}\, (x)^p\, J\, \mathbf{dx} \qquad p = 0...2(P-1)$

Note that with a general basis, exploiting the symmetry of $\mathbf{e}_{\beta(\varepsilon)\,\beta'(\varepsilon)}$, (P+2)(P+1)/2 derivative outer products must be evaluated at each integration point. When the monomial basis is used this operation count can be reduced to (2P–1).

P:	(P+2)(P+1)/2:	(2P–1):
1	3	1
2	6	3
3	10	5
4	15	7
5	21	9

2.6.3 Mesh Refinement

Throughout this chapter we have dealt with the approximation of fields by a finite dimensional subspace composed of piecewise polynomials. We shall now consider how one may reduce the errors associated with such approximations by increasing the dimension of the finite element subspace. There exist three general approaches to increase the accuracy of approximation:

(1) The *h*-version whereby the order of the element bases are fixed while the element sizes are reduced;

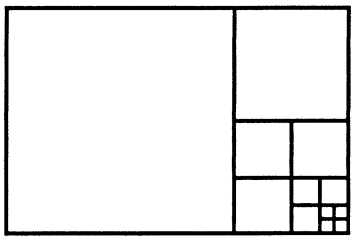
(2) The *p*-version where element sizes are fixed and the orders of the bases are increased;

(3) The *h*-*p*-version in which both the density of elements and the basis orders are increased.

The *h*-version is the traditional method for increasing the accuracy of field approximation and has been studied for many years. Most finite element texts include a discussion of the convergence rates associated with a reduction in the element size. See Oden (1972), Strang and Fix (1973), Babuska and Aziz (1972). Convergence rates for the *p*-version were considered by Babuska, Szabo and Katz (1981). They demonstrated that if the approximated field possesses a singularity at an element vertex then the rate of convergence is at least double that of the *h*-version. In both cases, however, the convergence rates are polynomial. Guo and Babuska (1986) showed that the *h*-*p*-version has an exponential rate of convergence with respect to the number of ensemble degrees of freedom.

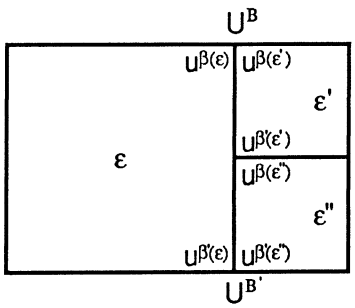
We shall now restrict our attention to the problem of generating a locally refined ensemble mesh from a relatively coarse one. Local mesh refinements are useful in the finite element analysis of elliptic problems with singularities where the form of the singularity is not known. In those problems where the form of the singularity is known *a priori* special bases with singular basis functions may be used to approximate the field (Fix 1969) (Barnhill and Whiteman 1975).

Local mesh refinement can present some difficulties where mid-boundary or 'hanging' nodes are produced:



Several authors have attempted to address this problem. Gregory *et al* (1978) demonstrated a technique for constructing an hybrid element possessing mid-boundary nodes. They firstly partitioned the element into subelements, applied appropriate constraints to the virtual nodes created on the normal boundaries, and then recombined the subelements to form the hybrid. Gupta (1978) also proposed the construction of an hybrid element using lofted interpolants from the mid-boundary node to the normal boundaries. Carey (1976) proposed that the mid-boundary nodes be handled by adding constraint equations in the element coefficient matrix and right hand side. This resulted in the formation of a fringe element, bounding the subelements, with appropriate constraints applied to the mid-boundary nodes.

We propose that the problem of mid-boundary nodes be handled by the ensemble to element parameter map. Since this map can be considered as a general linear transformation between the ensemble and the element parameters it can directly accommodate linear constraints between them. Consider for now the two dimensional problem of two smaller bilinear Lagrange elements, ε' and ε'', adjacent to a larger one, ε:



A mid-boundary node exists in element ε. Since element ε is bilinear, in order to retain zeroth order continuity, we must ensure that the field along the common boundary of elements ε, ε', and ε'' is constrained to be linear. Since we are dealing with linear Lagrange element bases there is an identity mapping between the element field parameters $\mathbf{u}^{\beta(\varepsilon)}$, $\mathbf{u}^{\beta'(\varepsilon)}$ and the corresponding ensemble field parameters $\mathbf{U}^{B}$, $\mathbf{U}^{B'}$. An identity mapping also exists between $\mathbf{u}^{\beta(\varepsilon')}$, $\mathbf{u}^{\beta'(\varepsilon'')}$ and $\mathbf{U}^{B}$, $\mathbf{U}^{B'}$. The coefficients of the ensemble to element parameter maps $\upsilon^{\beta(\varepsilon)}{}_{B}$, $\upsilon^{\beta'(\varepsilon)}{}_{B'}$, $\upsilon^{\beta(\varepsilon')}{}_{B}$, and $\upsilon^{\beta'(\varepsilon'')}{}_{B'}$ are thus all unity. Assuming that the mid-boundary node lies half way along the boundary of element ε we require that the field value at this point be the average of the value of the field at the two nearest vertices, $\mathbf{u}^{\beta(\varepsilon)}$ and $\mathbf{u}^{\beta'(\varepsilon)}$. The corresponding coefficients of the ensemble to element parameter maps $\upsilon^{\beta'(\varepsilon')}{}_{B}$, $\upsilon^{\beta'(\varepsilon')}{}_{B'}$, $\upsilon^{\beta(\varepsilon'')}{}_{B}$, and $\upsilon^{\beta(\varepsilon'')}{}_{B'}$ will thus all be 1/2.

It should be noted in the above that the continuity constraints have been handled by the ensemble to element parameter map. There was no need to construct special transition elements to accommodate the mid-boundary nodes, nor was there any need to provide constraints explicitly in the ensemble coefficient matrix and right hand side. The governing set of equations contain the required constraints implicitly thus ensuring that the number of ensemble field parameters is kept to a minimum.

While we have considered in detail only bilinear Lagrange elements having a single mid-boundary node positioned half way along the boundary the above methods are equally be applicable to higher order bases, higher dimension elements, and multiply partitioned elements. As long as the required constraints upon the element field parameters are linear they can be handled by the ensemble to element parameter map.

2 Bibliography

Abramowitz, M., Stegun, I. A. (1964) *Handbook Of Mathematical Functions With Formulas, Graphs, And Mathematical Tables,* Dover Publications Inc., New York.

Ascher, U., Pruess, S., Russell, R. D. (1983) "On spline basis selection for solving differential equations" *SIAM Journal On Numerical Analysis,* February 1983, volume 20, number 1, pages 121-142.

Babuska, I., Aziz, A. K. (1972) "Survey lectures on the mathematical foundations of the finite element method" in *The Mathematical Foundations Of The Finite Element Method With Applications To Partial Differential Equations,* edited by Aziz, A. K., Academic Press, New York, pages 5-359.

Babuska, I., Szabo, B. A., Katz, I. N. (1981) "The *p*-version of the finite element method" *SIAM Journal On Numerical Analysis,* June 1981, volume 18, number 3, pages 515-545.

Ball, A. A. (1984) "Reparameterization and its application in computer-aided geometric design" *International Journal For Numerical Methods In Engineering,* 1984, volume 20, pages 197-216.

Barnhill, R. E., Whiteman, J. R. (1975) "Error analysis of Galerkin methods for Dirichlet problems containing boundary singularities" *Journal Of The Institute Of Mathematics And Its Applications,* February 1975, volume 15, number 1, pages 121-125.

Barsky, B. A. (1984) "A description and evaluation of various 3-D models" *IEEE Computer Graphics And Applications,* January 1984, volume 4, number 1, pages 38-52.

Barsky, B. A., Beatty, J. C. (1983) "Local control of bias and tension in Beta-splines" *ACM Transactions On Graphics,* April 1983, volume 2, number 2, pages 109-134.

Belaga, E. C. (1958) "Some problems in the computation of polynomials" *Doklady Akademii Nauk SSSR,* 1958, volume 123, pages 775-777, (Russian).

Belaga, E. C. (1961) "On computing polynomials in one variable with initial conditioning of the coefficients" *Problems Of Cybernetics,* 1961, volume 5, pages 1-13.

Borodin, A. (1971) "Horners rule is uniquely optimal" in *Theory Of Machines And Computations,* edited by Kohavi, Z., Paz, A., Academic Press, New York, pages 45-58.

Böhm, W. (1984) "Efficient evaluation of splines" *Computing,* 1984, volume 33, pages 171-177.

Carey, G. F. (1976) "A mesh-refinement scheme for finite element computations" *Computer Methods In Applied Mechanics And Engineering,* 1976, volume 7, pages 93-105.

Chang, G. (1982) "Matrix formulations of Bézier technique" *Computer-Aided Design,* November 1982, volume 14, number 6, pages 345-350.

Clenshaw, C. W. (1955) "A note on the summation of Chebyshev series" *Mathematical Tables And Other Aids To Computation,* 1955, volume 9, pages 118-120.

Cohen, E., Lyche, T., Schumaker, L. L. (1985) "Algorithms for degree-raising of splines" *ACM Transactions On Graphics,* July 1985, volume 4, number 3, pages 171-181.

Coons, S. A. (1967) *Surfaces For Computer-Aided Design Of Space Forms,* Project MAC report MAC-TR-41, Massachusetts Institute of Technology, Cambridge.

Dahlquist, G., Björck, Å., Anderson, N. (1974) *Numerical Methods,* Prentice-Hall, Inc., Englewood Cliffs.

de Boor, C. (1978) *A Practical Guide To Splines,* Springer-Verlag, New York.

Farouki, R. T., Hinds, J. K. (1985) "A hierarchy of geometric forms" *IEEE Computer Graphics And Applications,* May 1985, volume 5, number 5, pages 51-78.

Ferguson, J. (1964) "Multivariable curve interpolation" *Journal Of The Association For Computing Machinery,* April 1964, volume 11, number 2, pages 221-228.

Fix, G. (1969) "Higher-order Rayleigh-Ritz approximations" *Journal Of Mathematics And Mechanics,* 1969, volume 18, number 7, pages 645-657.

Foley, J. D., van Dam, A. (1983) *Fundamentals Of Interactive Computer Graphics,* Addison-Wesley Publishing Company, Inc., Reading.

Frind, E. O. (1977) "An isoparametric Hermitian element for the solution of field problems" *International Journal For Numerical Methods In Engineering,* 1977, volume 11, pages 945-962.

Gautschi, W. (1978) "Questions of numerical condition related to polynomials" in *Recent Advances In Numerical Analysis,* edited by de Boor, C., Golub, G., Academic Press, pages 45-72.

Gordon, W. J. (1971) "Blending-function methods of bivariate and multivariate interpolation and approximation" *SIAM Journal On Numerical Analysis,* March 1971, volume 8, number 1, pages 158-177.

Gregory, J. A., Fishelov, D., Schiff, B., Whiteman, J. R. (1978) "Local mesh refinement with finite elements for elliptic problems" *Journal Of Computational Physics,* October 1978, volume 29, pages 133-140.

Guo, B., Babuska, I. (1986) "The *h-p*-version of the finite element method. Part 1: The basic approximation results" *Computational Mechanics,* 1986, volume 1, pages 21-41.

Gupta, A. K. (1978) "A finite element for transition from a fine to a coarse grid" *International Journal For Numerical Methods In Engineering,* 1978, volume 12, pages 35-45.

Hearn, A. C. (1973) *REDUCE 2 User's Manual,* University of Utah report UCP-19, University of Utah, Salt Lake City.

IBM (1981) *VS FORTRAN Application Programming: Library Reference,* SC26-3989, International Business Machines Corporation.

Kahmann, J. (1983) "Continuity of curvature between adjacent Bézier patches" in *Surfaces In Computer Aided Geometric Design,* edited by Barnhill, R. E., Boehm, W., North-Holland Publishing Company, Amsterdam, pages 65-75.

Knuth, D. E. (1969) *The Art Of Computer Programming Volume 2: Seminumerical Algorithms,* Addison-Wesley, Reading, Massachusetts.

Kontopidis, G. D., Limbert, D. E. (1983) "A predictor-corrector contouring algorithm for isoparametric 3D elements" *International Journal For Numerical Methods In Engineering,* 1983, volume 19, pages 995-1004.

Kronsjö, L. I. (1979) *Algorithms: Their Complexity And Efficiency,* John Wiley & Sons, Chichester.

Kung, H. T. (1976) "New algorithms and lower bounds for the parallel evaluation of certain rational expressions and recurrences" *Journal Of The Association For Computing Machinery,* April 1976, volume 23, number 2, pages 252-261.

Lehman, R. S. (1959) "Developments at an analytic corner of solutions of elliptic partial differential equations" *Journal Of Mathematics And Mechanics,* 1959, volume 8, pages 727-760.

Lewis, J. W. (1981) "Interchanging spline curves using IGES" *Computer-Aided Design,* November 1981, volume 13, number 6, pages 359-364.

Lozinskii, E. L. (1983) "An algorithm for parallel evaluation of functions" *Journal Of Algorithms,* September 1983, volume 4, number 3, pages 271-281.

Mikosko, J., Kotov, V. E. (1984) *Algorithms, Software and Hardware Of Parallel Computers,* Springer Verlag, Berlin.

Motzkin, T. S. (1955) "Evaluation of polynomials" *Bulletin Of The American Mathematical Society,* March 1955, volume 61, page 163.

Newbery, A. C. R. (1974) "Error analysis for polynomial evaluation" *Mathematics Of Computation,* July 1974, volume 28, number 127, pages 789-793.

Newbery, A. C. R. (1975) "Polynomial evaluation schemes" *Mathematics Of Computation,* October 1975, volume 29, number 132, pages 1046-1050.

Newman, W. M., Sproull, R. F. (1979) *Principles Of Interactive Computer Graphics, Second Edition,* McGraw-Hill International Book Company.

Oden, J. T. (1972) *Finite Elements Of Nonlinear Continua,* McGraw-Hill Book Company, New York.

Oden, J. T., Reddy, J. N. (1976) *An Introduction To The Mathematical Theory Of Finite Elements,* John Wiley & Sons, New York, 1976.

Oliver, J. (1979) "Rounding error propagation in polynomial evaluation schemes" *Journal Of Computational And Applied Mathematics,* 1979, volume 5, number 2, pages 85-97.

Pan, V. Y. (1959) "Schemes for computing polynomials with real coefficients" *Doklady Akademii Nauk SSSR,* 1959, volume 127, pages 266-269, (Russian), (English translation in *Mathematical Reviews,* 1962, volume 23, review number B560)

Powell, M. J. D. (1981) *Approximation Theory And Methods,* Cambridge University Press, Cambridge.

Rabin, M., Winograd, S. (1971) *Fast Evaluation Of Polynomials By Rational Preparation,* IBM Technical Report RC 3645, December 1971.

Rack, H., Reimer, M. (1982) "The numerical stability of evaluation schemes for polynomials based on the Lagrange interpolation form" *BIT,* 1982, volume 22, pages 101-107.

Rice, J. R. (1965) "On the conditioning of polynomial and rational forms" *Numerische Mathematik,* 1965, volume 7, pages 426-435.

Rivlin, T. J. (1974) *The Chebyshev Polynomials,* Wiley, New York.

Schumaker, L. L. (1981) *Spline Functions: Basic Theory,* John Wiley & Sons, New York.

Sharpe, R. J., Thorne, R. W. (1982) "Numerical method for extracting an arc length parameterization from parametric curves" *Computer-Aided Design,* March 1982, volume 14, number 2, pages 79-81.

Shaw, M., Traub, J. F. (1974) "On the number of multiplications for the evaluation of a polynomial and some of its derivatives" *Journal Of The Association For Computing Machinery,* 1974, volume 21, number 1, pages 161-167.

Smith, F. J. (1965) "An algorithm for summing orthogonal polynomial series and their derivatives with applications to curve fitting and interpolation" *Mathematics Of Computation,* volume 19, pages 33-36.

Stanton, E. L., Crain, L. M., Neu, T. F. (1977) "A parametric cubic modelling system for general solids of composite material" *International Journal For Numerical Methods In Engineering,* 1977, volume 11, pages 653-670.

Strang, G., Fix, G. J. (1973) *An Analysis Of The Finite Element Method,* Prentice-Hall Inc., Englewood Cliffs.

Sørensen, H. H., Hoppe, V. (1978) "Interface problems between finite element regions with dissimilar function spaces" *The Mathematics Of Finite Elements And Applications 1978,* April 1978.

Timmer, H. G. (1980) "Alternative representation for parametric cubic curves and surfaces" *Computer-Aided Design,* January 1980, volume 12, number 1, pages 25-28.

van Genuchten, M. T., Pinder, G. F., Frind, E. O. (1977) "Simulation of two-dimensional contaminant transport with isoparametric hermitian finite elements" *Water Resources Research,* April 1977, volume 13, number 2, pages 451-458.

Wang, C. Y. (1981) "Shape classification of the parametric cubic curve and parametric B-spline cubic curve" *Computer-Aided Design,* July 1981, volume 13, number 4, pages 199-206.

Wilkinson, J. H. and Reinsch, C. (1971) *Linear Algebra,* Springer-Verlag, Berlin.

Woznaikowski, H. (1974) "Rounding error analysis for the evaluation of a polynomial and some of its derivatives" *SIAM Journal On Numerical Analysis,* September 1974, volume 11, number 4, pages 780-787.

Zienkiewicz, O. C. (1977) *The Finite Element Method, Third Edition,* McGraw-Hill Book Company (UK) Limited, London.

Zienkiewicz, O. C., Morgan, K. (1982) *Finite Elements And Approximation,* John Wiley & Sons, New York.