OpenCMISS NOTES

http://www.opencmiss.org/

August 22, 2018

Contents

List of Figures

List of Tables

List of Symbols

$\boldsymbol{F}$	Deformation gradient tensor. Defined at page 126.
$\boldsymbol{U}$	Right stretch tensor. Defined at page 127.
$\boldsymbol{V}$	Left stretch tensor. Defined at page 127.
$\boldsymbol{R}$	Rotation tensor. Defined at page 127.
$\boldsymbol{C}$	Right Cauchy-Green (or Green) deformation tensor. Defined at page 128.
$\boldsymbol{B}$	Piola deformation tensor. Defined at page 128.
$\boldsymbol{E}$	Green-Lagrange strain tensor. Defined at page 128.
$\boldsymbol{b}$	Left Cauchy-Green (or Finger) deformation tensor. Defined at page 129.
$\boldsymbol{c}$	Cauchy deformation tensor. Defined at page 129.
$\boldsymbol{e}$	Euler-Almansi strain tensor. Defined at page 129.
$\boldsymbol{\sigma}$	Cauchy stress tensor. Defined at page 131.
$\boldsymbol{\tau}$	Kirchoff stress tensor. Defined at page 131.
$\boldsymbol{P}$	First Piola-Kirchoff stress tensor. Defined at page 131.
$\boldsymbol{S}$	Second Piola-Kirchoff stress tensor. Defined at page 132.
$\mathsf{\mathbf{A}}$	First elasticity tensor. Defined at page 136.

C	Second elasticity tensor. Defined at page 136.
a	First spatial elasticity tensor. Defined at page 138.
c	Second spatial elasticity tensor. Defined at page 138.

Chapter 1

Introduction

Chapter 2

Basis Function and Interpolation

2.1 Representing a One-Dimensional Field

Consider the problem of finding a mathematical expression $u(x)$ to represent a one-dimensional field *e.g.*, measurements of temperature u against distance x along a bar, as shown in Figure 2.1.

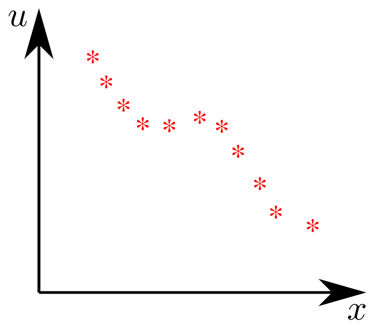

FIGURE 2.1: Temperature distribution $u(x)$ along a bar. The points are the measured temperatures.

One approach would be to use a polynomial expression $u(x) = a + bx + cx^2 + dx^3 + \dots$ and to estimate the values of the parameters a, b, c and d from a least-squares fit to the data. As the degree of the polynomial is increased the data points are fitted with increasing accuracy and polynomials provide a very convenient form of expression because they can be differentiated and integrated readily. For low degree polynomials this is a satisfactory approach, but if the polynomial order is increased further to improve the accuracy of fit a problem arises:

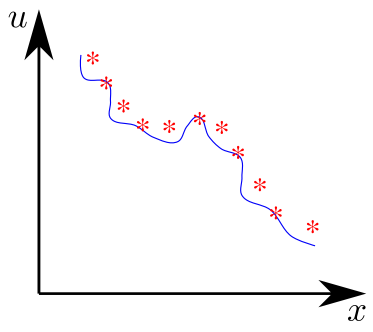

FIGURE 2.2: A least-squares polynomial fit to the data, showing the unacceptable oscillation between data points

the polynomial can be made to fit the data accurately, but it oscillates unacceptably between the data points, as shown in Figure 2.2.

2.2 Summation Notation

The following (Einstein) summation notation will be used throughout these notes. In order to eliminate summation symbols repeated "dummy" indices will be used *i.e.,*

$$\sum_{i=1}^{n} a^i b_i = a^i b_i \tag{2.1}$$

To indicate an index that is not summed, parentheses will be used *i.e.,* $a^{(i)}b_{(i)}$ is talking about the singular expression for i *e.g.,* a^1b_1, a^2b_2 *etc.*

In order to indicate a summation the sum must occur over indices that are different sub/superscript *i.e.,* the sum must be over an "upper" and a "lower" index or a "lower" and an "upper" index. Note that it may be useful to remember that if an index appears in the denominator of a fractional expression then the index upper- or lower- ness is "reversed".

For some quantities with both upper and lower indices a dot will be used to indicate the "second" index *e.g.,* in the expression $A^i_{.j}$ then i can be considered the first index and j the second index.

2.3 Basis Functions and Interpolation

Both the finite element method (FEM) and the boundary element method (BEM) use interpolation in finding a field solution *i.e.,* the methods find the solution at a number of points in the domain of interest and then approximate the solution between these points using interpolation. The points at which the solution is found are known as *nodes*. *Basis functions* are used to interpolate the field between nodes within a subregion of the domain known as an *element*. Interpolation is achieved by mapping the field coordinate onto a *local parametric*, or ξ, coordinate (which varies from 0 to 1) within each element. The global nodes which make up each element are also mapped onto local element nodes and the basis functions are chosen (in terms of polynomials of the local parametric coordinate) such that the interpolated field is equal to the known nodal values at each node and is thus continuous between elements. A schematic of this scheme is shown in Figure 2.3.

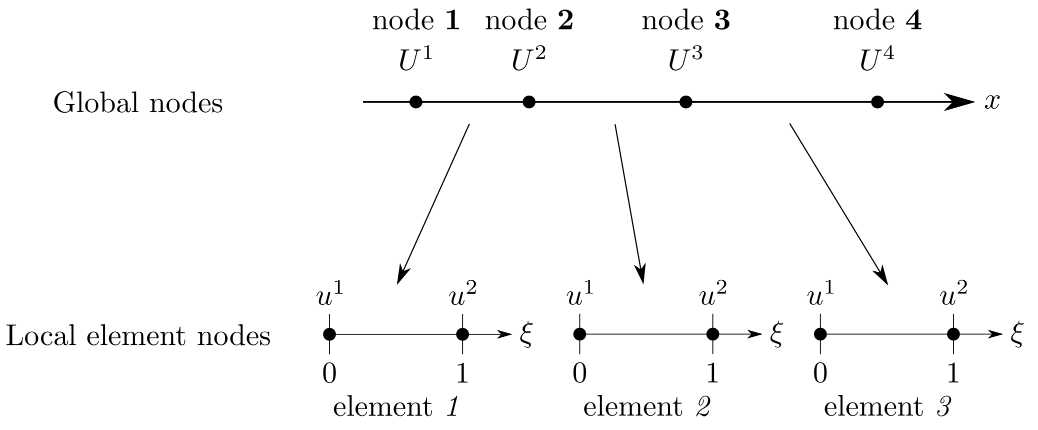

FIGURE 2.3: A schematic of the relationship between local and global nodes, elements and the parametric elemental ξ coordinate.

2.3.1 Lagrangian Basis Functions

One important family of basis functions are the Lagrange basis functions. This family has one basis function for each of the local element nodes and are defined such that, at a particular node, only one basis function is non-zero and has the value of one. In this sense a basis function can be thought of as being associated with a local node and serves to weight the interpolated solution in terms of the field value at that node. Lagrange basis functions hence provide only C^0 continuity of the field variable across element boundaries.

Linear Lagrange basis functions

The simplest basis functions of the Lagrange family are the one-dimensional linear Lagrange basis functions. These basis functions involve two local nodes and are defined as

$$\begin{aligned} \varphi_1(\xi) &= 1 - \xi \\ \varphi_2(\xi) &= \xi \end{aligned} \tag{2.2}$$

The two one-dimensional linear Lagrange basis functions are gshown in Figure 2.4.

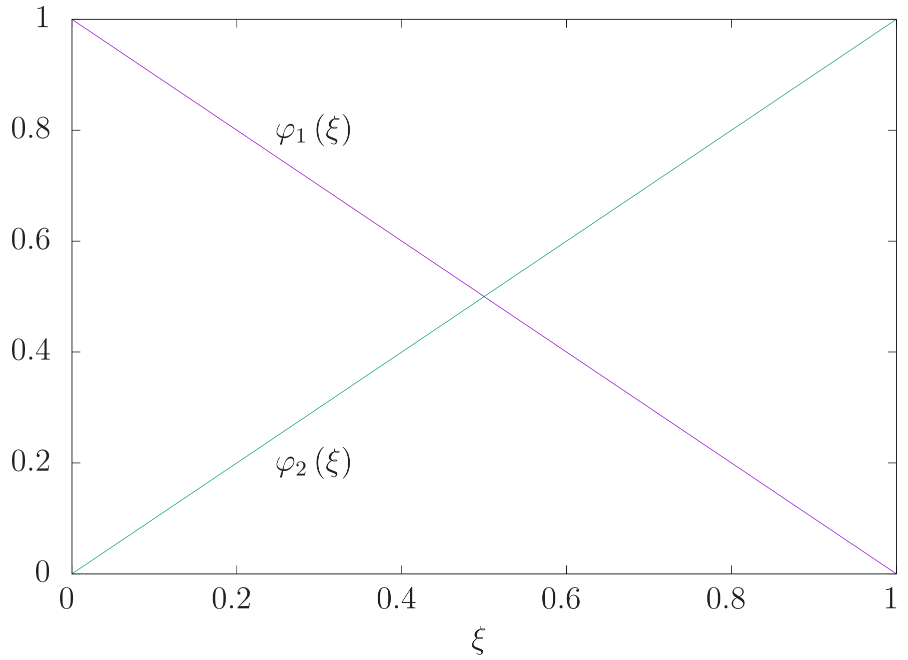

FIGURE 2.4: Linear Lagrange basis functions.

The interpolation of a field variable, u, using these basis functions is given by

$$\begin{aligned} u(\xi) &= \varphi_1(\xi)\, u^1 + \varphi_2(\xi)\, u^2 \\ &= (1 - \xi)\, u^1 + \xi u^2 \end{aligned} \tag{2.3}$$

where u^1 and u^2 are the values of the field variable at the first and second local nodes respectively. These basis functions hence provide a linear variation between the local nodal values with the local element coordinate, ξ.

Quadratic Lagrange basis functions

Lagrange basis functions can also be used to provide higher order variations, for example the one-dimensional quadratic Lagrange basis functions involve three local nodes and can provide a quadratic variation of field parameter with ξ. They are defined as

$$\begin{aligned}\varphi_1(\xi) &= 2\left(\xi-\frac{1}{2}\right)(\xi-1)\\ \varphi_2(\xi) &= 4\xi(1-\xi)\\ \varphi_3(\xi) &= 2\xi\left(\xi-\frac{1}{2}\right)\end{aligned} \tag{2.4}$$

The three one-dimensional quadratic Lagrange basis functions are shown in Figure 2.5.

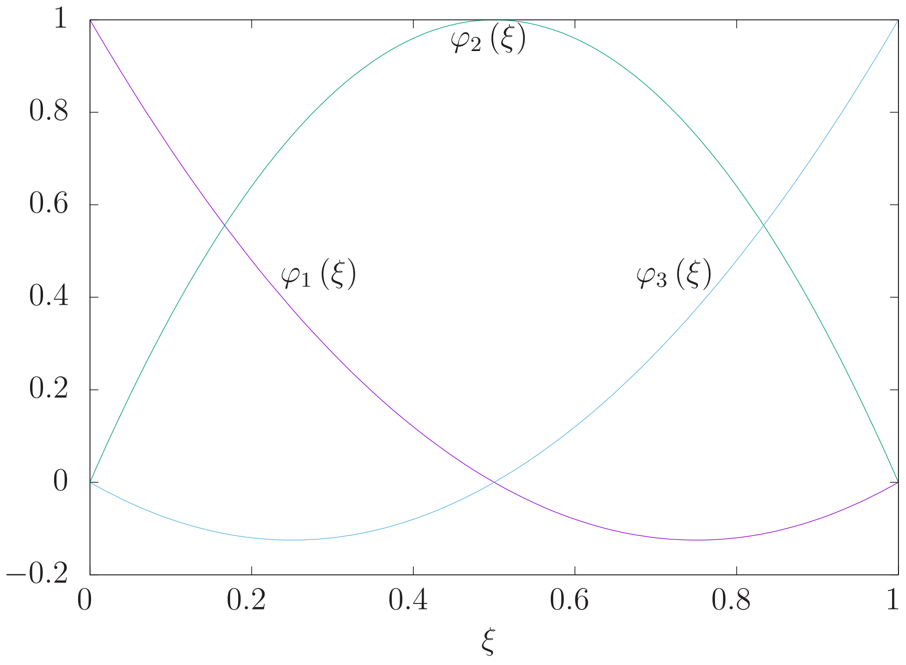

FIGURE 2.5: Quadratic Lagrange basis functions.

The interpolation formula is

$$\begin{aligned} u(\xi) &= \varphi_1(\xi)\, u^1 + \varphi_2(\xi)\, u^2 + \varphi_3(\xi)\, u^3 \\ &= 2\left(\xi - \frac{1}{2}\right)(\xi - 1)\, u^1 + 4\xi(1-\xi)\, u^2 + 2\xi\left(\xi - \frac{1}{2}\right) u^3 \end{aligned} \tag{2.5}$$

Cubic Lagrange basis functions

One-dimensional cubic Lagrange basis functions involve four local nodes and can provide a cubic variation of field parameter with ξ. They are defined as

$$\begin{aligned} \varphi_1(\xi) &= \frac{1}{2}(3\xi - 1)(3\xi - 2)(1 - \xi) \\ \varphi_2(\xi) &= \frac{9}{2}\xi(3\xi - 2)(\xi - 1) \\ \varphi_3(\xi) &= \frac{9}{2}\xi(3\xi - 1)(1 - \xi) \\ \varphi_4(\xi) &= \frac{1}{2}\xi(3\xi - 1)(3\xi - 2) \end{aligned} \tag{2.6}$$

The four one-dimensional cubic Lagrange basis functions are shown in Figure 2.6.

The interpolation formula is

$$\begin{aligned} u(\xi) &= \varphi_1(\xi)\, u^1 + \varphi_2(\xi)\, u^2 + \varphi_3(\xi)\, u^3 + \varphi_4(\xi)\, u^4 \\ &= \frac{1}{2}(3\xi - 1)(3\xi - 2)(1 - \xi)\, u^1 + \frac{9}{2}\xi(3\xi - 2)(\xi - 1)\, u^2 \\ &\quad + \frac{9}{2}\xi(3\xi - 1)(1 - \xi)\, u^3 + \frac{1}{2}\xi(3\xi - 1)(3\xi - 2)\, u^4 \end{aligned} \tag{2.7}$$

General Lagrange basis functions

In general the interpolation formula for the Lagrange family of basis functions is, using *Einstein summation notation*, given by

$$u(\xi) = \varphi_\alpha(\xi)\, u^\alpha \quad \alpha = 1, \ldots, n_e \tag{2.8}$$

where n_e is the number of local nodes in the element. Einstein summation notation uses a repeated index in a product expression to imply summation. For example Equation (2.8) is

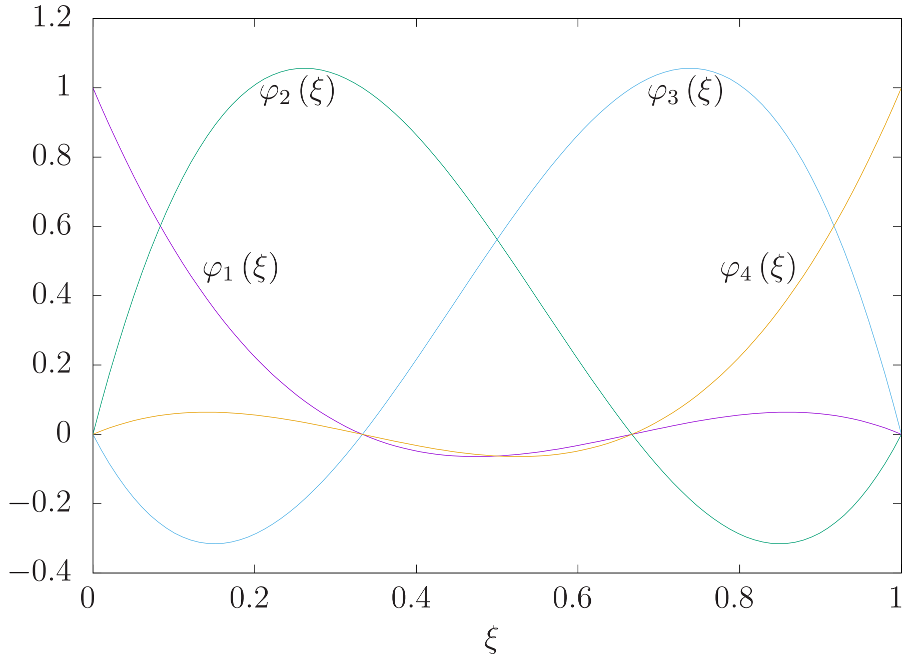

FIGURE 2.6: Cubic Lagrange basis functions.

equivalent to

$$u(\xi) = \sum_{\alpha=1}^{n_e} \varphi_\alpha(\xi)\, u^\alpha \tag{2.9}$$

Bilinear Lagrange basis functions

Multi-dimensional Lagrange basis functions can be constructed from the tensor, or outer, products of the one-dimensional Lagrange basis functions. For example the two-dimensional bilinear Lagrange basis functions have four local nodes with the basis functions given by

$$\begin{aligned}
\varphi_1(\xi_1,\xi_2) &= \varphi_1(\xi_1)\,\varphi_1(\xi_2) = (1-\xi_1)(1-\xi_2)\\
\varphi_2(\xi_1,\xi_2) &= \varphi_2(\xi_1)\,\varphi_1(\xi_2) = \xi_1(1-\xi_2)\\
\varphi_3(\xi_1,\xi_2) &= \varphi_1(\xi_1)\,\varphi_2(\xi_2) = (1-\xi_1)\,\xi_2\\
\varphi_4(\xi_1,\xi_2) &= \varphi_2(\xi_1)\,\varphi_2(\xi_2) = \xi_1\xi_2
\end{aligned} \tag{2.10}$$

The four two-dimensional bilinear Lagrange basis functions are shown in Figure 2.7.

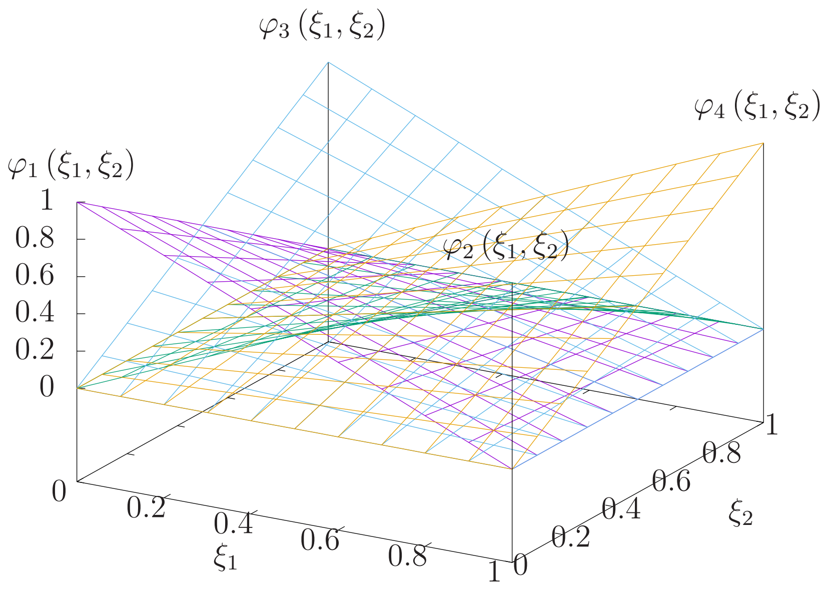

FIGURE 2.7: Bilinear Lagrange basis functions.

The multi-dimensional interpolation formula is still a sum of the products of the nodal basis function and the field value at the node. For example the interpolated geometric position vector within an element is given by

$$\begin{aligned}\boldsymbol{x}(\xi_1, \xi_2) &= \varphi_\alpha(\xi_1, \xi_2)\,\boldsymbol{x}^\alpha \\ &= \varphi_1(\xi_1, \xi_2)\,\boldsymbol{x}^1 + \varphi_2(\xi_1, \xi_2)\,\boldsymbol{x}^2 + \varphi_3(\xi_1, \xi_2)\,\boldsymbol{x}^3 + \varphi_4(\xi_1, \xi_2)\,\boldsymbol{x}^4\end{aligned} \tag{2.11}$$

where, for the vector field, each component is interpolated separately using the given basis functions.

Biquadratic Lagrange basis functions

The two-dimensional biquadratic Lagrange basis functions have nine local nodes with the basis functions given by

$$\begin{aligned}
\varphi_1(\xi_1,\xi_2) &= \varphi_1(\xi_1)\varphi_1(\xi_2) = 4\left(\xi_1-\frac{1}{2}\right)(\xi_1-1)\left(\xi_2-\frac{1}{2}\right)(\xi_2-1)\\
\varphi_2(\xi_1,\xi_2) &= \varphi_2(\xi_1)\varphi_1(\xi_2) = 8\xi_1(1-\xi_1)\left(\xi_2-\frac{1}{2}\right)(\xi_2-1)\\
\varphi_3(\xi_1,\xi_2) &= \varphi_3(\xi_1)\varphi_1(\xi_2) = 4\xi_1\left(\xi_1-\frac{1}{2}\right)\left(\xi_2-\frac{1}{2}\right)(\xi_2-1)\\
\varphi_4(\xi_1,\xi_2) &= \varphi_1(\xi_1)\varphi_2(\xi_2) = 8\left(\xi_1-\frac{1}{2}\right)(\xi_1-1)\xi_2(1-\xi_2)\\
\varphi_5(\xi_1,\xi_2) &= \varphi_2(\xi_1)\varphi_2(\xi_2) = 16\xi_1(1-\xi_1)\xi_2(1-\xi_2)\\
\varphi_6(\xi_1,\xi_2) &= \varphi_3(\xi_1)\varphi_2(\xi_2) = 8\xi_1\left(\xi_1-\frac{1}{2}\right)\xi_2(1-\xi_2)\\
\varphi_7(\xi_1,\xi_2) &= \varphi_1(\xi_1)\varphi_3(\xi_2) = 4\left(\xi_1-\frac{1}{2}\right)(\xi_1-1)\xi_2\left(\xi_2-\frac{1}{2}\right)\\
\varphi_8(\xi_1,\xi_2) &= \varphi_2(\xi_1)\varphi_3(\xi_2) = 8\xi_1(1-\xi_1)\xi_2\left(\xi_2-\frac{1}{2}\right)\\
\varphi_9(\xi_1,\xi_2) &= \varphi_3(\xi_1)\varphi_3(\xi_2) = 4\xi_1\left(\xi_1-\frac{1}{2}\right)\xi_2\left(\xi_2-\frac{1}{2}\right)
\end{aligned} \tag{2.12}$$

The four two-dimensional biquadratic Lagrange basis functions are shown in Figure 2.8.

2.3.2 Hermitian Basis Functions

Hermitian basis functions preserve continuity of the derivative of the interpolating variable *i.e.,* C^1 continuity, with respect to ξ across element boundaries by defining additional nodal derivative parameters. Like Lagrange bases, Hermitian basis functions are also chosen so that, at a particular node, only one basis function is non-zero and equal to one. They also are chosen so that, at a particular node, the *derivative* of only one of four basis functions is non-zero and is equal to one. Hermitian basis functions hence serve to weight the interpolated solution in terms of the field value and derivative of the field value at nodes.

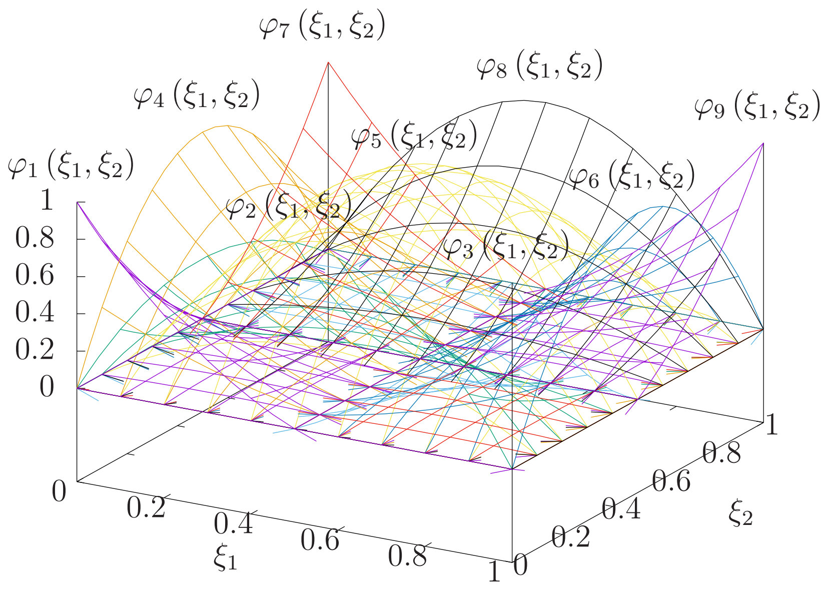

FIGURE 2.8: Biquadratic Lagrange basis functions.

Cubic Hermite basis functions

Cubic Hermite basis functions are the simplest of the Hermitian family and involve two local nodes per element. The interpolation within each element is in terms of $\boldsymbol{x}^{\alpha}$ and $\left.\frac{d\boldsymbol{x}}{d\xi}\right|_{\alpha}$ and is given by

$$\boldsymbol{x}(\xi) = \Psi_1^0(\xi)\,\boldsymbol{x}^1 + \Psi_1^1(\xi)\left.\frac{d\boldsymbol{x}}{d\xi}\right|_1 + \Psi_2^0(\xi)\,\boldsymbol{x}^2 + \Psi_2^1(\xi)\left.\frac{d\boldsymbol{x}}{d\xi}\right|_2 \tag{2.13}$$

where the four one-dimensional cubic Hermite basis functions are given in Equation (2.14) and shown in Figure 2.9.

$$\begin{aligned}\Psi_1^0(\xi) &= 1 - 3\xi^2 + 2\xi^3\\ \Psi_1^1(\xi) &= \xi(\xi-1)^2\\ \Psi_2^0(\xi) &= \xi^2(3-2\xi)\\ \Psi_2^1(\xi) &= \xi^2(\xi-1)\end{aligned} \tag{2.14}$$

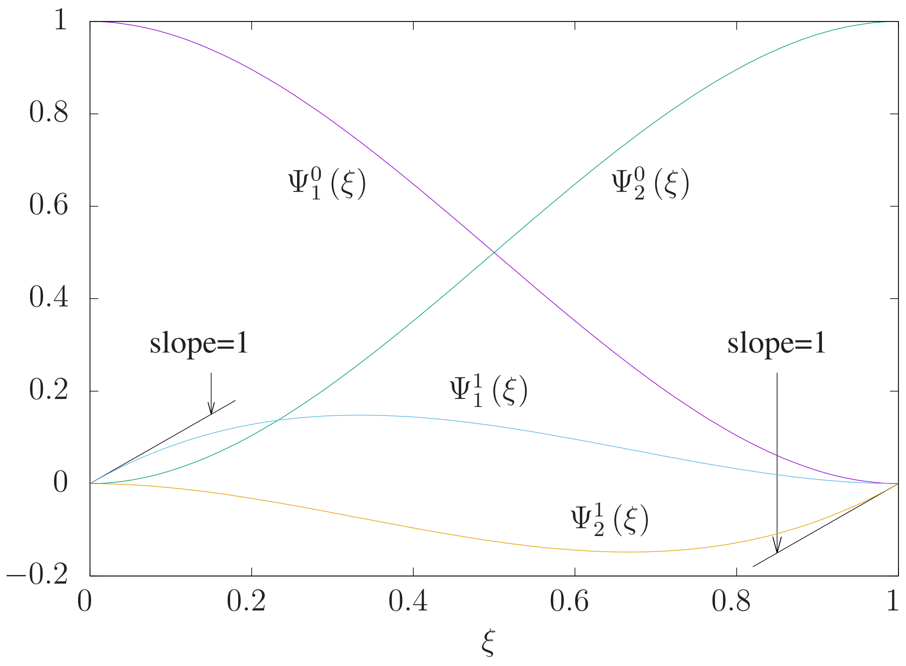

FIGURE 2.9: Cubic Hermite basis functions.

Scaling

One further step is required to make cubic Hermite basis functions useful in practice. Consider the two cubic Hermite elements shown in Figure 2.10.

The derivative $\left.\dfrac{d\boldsymbol{x}}{d\xi}\right|_{\alpha}$ defined at local node α is dependent upon the local element ξ-coordinate and is therefore, in general, different in the two adjacent elements. Interpretation of the derivative is hence difficult as two derivatives with the same magnitude in different parts of the mesh might represent two completely different physical derivatives. This is problematic for modelling and computation if the interpretation of the magnitude of the derivative (or *scaling*) is unknown *e.g.,* we cannot assign physical units. If the scaling varies throughout the mesh then a derivative at a node that has a magnitude of, say, 5 will be different from another derivative at another node that also has the magnitude of 5. Thus, a numerical solver that is given a vector of derivative values would assume that the scalings are the same and interpret the magnitudes identically. This would mean that algorithms may fail *e.g.,* if, say, we needed to compute the norm of a vector of derivatives then by assuming the same scaling the wrong result would be

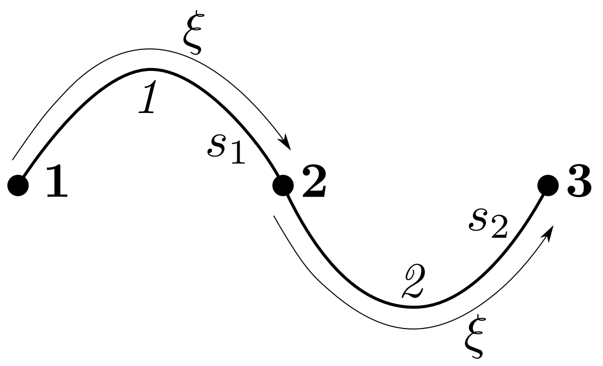

FIGURE 2.10: Two cubic Hermite elements (denoted by *1* and *2*) formed from three nodes (shown as a ● and denoted by $\mathbf{1}, \mathbf{2}$ and $\mathbf{3}$) and having arc-lengths s_1 and s_2 respectively.

computed.

In order to the have a consistent interpretation of the derivative throughout the mesh it is better to base the interpolation on a physical coordinate. Whilst we are free to choose the physical coordinate to be anything the optimum choice is arc length as this is what physical processes are based on. However, arc-length is extremely difficult to use as an interpolation parameter as the inherent nonlinearity involved in its calculation makes conversion to and from coordinates non trivial. The solution is to find a parameter that scales in the same way as arc-length or as close to it as we can.

Consider then basing the derivatives on an arc-length coordinate at nodes, $\dfrac{d\boldsymbol{x}^{\alpha}}{ds}$, with

$$\begin{aligned} \left.\frac{d\boldsymbol{x}}{d\xi}\right|_{\alpha} &= \frac{d\boldsymbol{x}^{\Delta(\alpha,e)}}{ds}\left(\frac{ds}{d\xi}\right)_e \\ &= \frac{d\boldsymbol{x}^{\Delta(\alpha,e)}}{ds} S\left(e\right) \end{aligned} \tag{2.15}$$

used to determine $\left.\dfrac{d\boldsymbol{x}}{d\xi}\right|_{\alpha}$. Here $\dfrac{d\boldsymbol{x}}{ds}$ is a physical arc-length derivative, $\Delta\left(\alpha, e\right)$ is the global node number of local node α in element e, $\left(\dfrac{ds}{d\xi}\right)_e$ is an element *scale factor*, denoted by $S\left(e\right)$, which scales the arc-length derivative to the ξ-coordinate derivative. Thus $\dfrac{d\boldsymbol{x}}{ds}$ is constrained to be continuous across element boundaries rather than $\dfrac{d\boldsymbol{x}}{d\xi}$. The cubic Hermite interpolation formula now becomes

$$\boldsymbol{x}(\xi) = \Psi_1^0(\xi)\,\boldsymbol{x}^1 + \Psi_1^1(\xi)\,\frac{d\boldsymbol{x}^1}{ds}S(e) + \Psi_2^0(\xi)\,\boldsymbol{x}^2 + \Psi_2^1(\xi)\,\frac{d\boldsymbol{x}^2}{ds}S(e) \tag{2.16}$$

By interpolating with respect to s rather than with respect to ξ there is some liberty as to the choice of the element scale factor, $S(e)$. The choice of the scale factor will, however, affect how ξ changes with s. It is computationally desirable to have a relatively uniform change of ξ with s (for example not biasing the Gaussian quadrature – see later – scheme to one end of the element). For this reason the element scale factor is chosen as some function of the arc-length of the element, s_e. The simplest linear function that can be chosen is the arc-length itself. This type of scaling is called *arc-length scaling*.

To calculate the arc-length for a particular element an iterative process is needed. The arc-length for a one-dimensional element in two-dimensions is defined as

$$\text{arc-length, } s_e = \int_0^1 \left\| \frac{d\boldsymbol{x}(\xi)}{d\xi} \right\|_2 \mathbf{d}\xi = \int_0^1 \sqrt{\left(\frac{dx(\xi)}{d\xi}\right)^2 + \left(\frac{dy(\xi)}{d\xi}\right)^2}\,\mathbf{d}\xi \tag{2.17}$$

However, since the interpolation of $\boldsymbol{x}(\xi)$, as defined in Equation (2.16), uses the arc-length in the calculation of the scaling factor, an iterative root finding technique is needed to obtain the arc-length.

Thus, for an element e, the one-dimensional cubic Hermite interpolation formula in Equation (2.16) becomes

$$\boldsymbol{x}(\xi) = \Psi_\alpha^u(\xi)\,\boldsymbol{x}_{,u}^\alpha S(e,u) \tag{2.18}$$

where α varies from 1 to 2, u varies from 0 to 1, $\boldsymbol{x}_{,0}^\alpha = \boldsymbol{x}^\alpha$, $\boldsymbol{x}_{,1}^\alpha = \dfrac{d\boldsymbol{x}^\alpha}{ds}$, $S(e,0) = 1$ and $S(e,1) = S(e) = s_e$. Equation (2.18) is equivalent to

$$\boldsymbol{x}(\xi) = \Psi_1^0(\xi)\,\boldsymbol{x}_{,0}^1 S(e,0) + \Psi_1^1(\xi)\,\boldsymbol{x}_{,1}^1 S(e,1) + \Psi_2^0(\xi)\,\boldsymbol{x}_{,0}^2 S(e,0) + \Psi_2^1(\xi)\,\boldsymbol{x}_{,1}^2 S(e,1) \tag{2.19}$$

i.e., there is an implied sum with α and u for $\Psi_\alpha^u(\xi)$ and $\boldsymbol{x}_{,u}^\alpha$ but not for $S(e,u)$.

There is one final condition that must be placed on the ξ to arc-length transformation to ensure arc-length derivatives. This condition is based on the geometric defintion of arc-length which is given by Pythagorus *i.e.,* for two-dimensions in rectangular cartesian coordinate we have

$$ds^2 = dx^2 + dy^2 \tag{2.20}$$

or, in general coordinates,

$$ds^2 = g_{ij} dx^i dx^j \tag{2.21}$$

where g_{ij} are the components of the metric tensor.

Rearranging Equation (2.20) we find that the arc-length derivative vector at a node for geometric like fields, for rectangular cartesian coordinates, must have unit magnitude. Thus for global node A we have

$$\left\| \frac{d\boldsymbol{x}^A}{ds} \right\|_2 = 1 \tag{2.22}$$

In general coordinates this condition becomes

$$\left\| \frac{\partial \boldsymbol{x}^A}{\partial s_k} \right\|_2 = \sqrt{\det \boldsymbol{g}} \tag{2.23}$$

where s_k is the k^{th} global arc-length direction and $\boldsymbol{g}$ is the metric tensor.

The use of this constraint on arc-length derivative magnitude ensures that there is continuity with respect to a physical parameter, s, rather than with respect to a mathematical parameter ξ. The set of mesh parameters, $\boldsymbol{u}$, for cubic Hermite interpolation hence contains the set of nodal values (or positions), the set of nodal arc-length derivatives and the set of scale factors.

Extension to higher orders

Bicubic Hermite basis functions are the two-dimensional extension of the one-dimensional cubic Hermite basis functions. They are formed from the tensor (or outer) product of two of the one-dimensional cubic Hermite basis functions defined in Equation (2.14). The interpolation formula for the point $\boldsymbol{x}\left(\xi_1, \xi_2\right)$ within an element is obtained from the bicubic Hermite interpolation formula (**?**),

$$\begin{aligned}\boldsymbol{x}(\xi_1,\xi_2) = {} & \Psi_1^0(\xi_1)\,\Psi_1^0(\xi_2)\,\boldsymbol{x}^1 + \Psi_2^0(\xi_1)\,\Psi_1^0(\xi_2)\,\boldsymbol{x}^2 + \\ & \Psi_1^0(\xi_1)\,\Psi_2^0(\xi_2)\,\boldsymbol{x}^3 + \Psi_2^0(\xi_1)\,\Psi_2^0(\xi_2)\,\boldsymbol{x}^4 + \\ & \Psi_1^1(\xi_1)\,\Psi_1^0(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_1}\right|_1 + \Psi_2^1(\xi_1)\,\Psi_1^0(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_1}\right|_2 + \\ & \Psi_1^1(\xi_1)\,\Psi_2^0(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_1}\right|_3 + \Psi_2^1(\xi_1)\,\Psi_2^0(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_1}\right|_4 + \\ & \Psi_1^0(\xi_1)\,\Psi_1^1(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_2}\right|_1 + \Psi_2^0(\xi_1)\,\Psi_1^1(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_2}\right|_2 + \\ & \Psi_1^0(\xi_1)\,\Psi_2^1(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_2}\right|_3 + \Psi_2^0(\xi_1)\,\Psi_2^1(\xi_2)\left.\frac{\partial\boldsymbol{x}}{\partial\xi_2}\right|_4 + \\ & \Psi_1^1(\xi_1)\,\Psi_1^1(\xi_2)\left.\frac{\partial^2\boldsymbol{x}}{\partial\xi_1\partial\xi_2}\right|_1 + \Psi_2^1(\xi_1)\,\Psi_1^1(\xi_2)\left.\frac{\partial^2\boldsymbol{x}}{\partial\xi_1\partial\xi_2}\right|_2 + \\ & \Psi_1^1(\xi_1)\,\Psi_2^1(\xi_2)\left.\frac{\partial^2\boldsymbol{x}}{\partial\xi_1\partial\xi_2}\right|_3 + \Psi_2^1(\xi_1)\,\Psi_2^1(\xi_2)\left.\frac{\partial^2\boldsymbol{x}}{\partial\xi_1\partial\xi_2}\right|_4\end{aligned} \tag{2.24}$$

As with one-dimensional cubic Hermite elements, the derivatives with respect to ξ in the two-dimensional interpolation formula above are expressed as the product of a nodal arc-length derivative and a scale factor. This is, however, complicated by the fact that there are now multiple ξ directions at each node. From the product rule the transformation from an ξ based derivative to an arc-length based derivative is given by,

$$\frac{\partial\boldsymbol{x}}{\partial\xi_l} = \frac{\partial\boldsymbol{x}}{\partial s_1}\frac{\partial s_1}{\partial\xi_l} + \frac{\partial\boldsymbol{x}}{\partial s_2}\frac{\partial s_2}{\partial\xi_l} \tag{2.25}$$

Now, by definition, the l^{th} arc-length direction is only a function of the l^{th} ξ direction, hence the derivative at local node α is

$$\left.\frac{\partial\boldsymbol{x}}{\partial\xi_l}\right|_\alpha = \frac{\partial\boldsymbol{x}^{\Delta(\alpha,e)}}{\partial s_l}S(e,l) \tag{2.26}$$

and the cross-derivative is

$$\left.\frac{\partial^2\boldsymbol{x}}{\partial\xi_1\partial\xi_2}\right|_\alpha = \frac{\partial^2\boldsymbol{x}^{\Delta(\alpha,e)}}{\partial s_1\partial s_2}S(e,1)\,S(e,2) \tag{2.27}$$

Unlike the one-dimensional cubic Hermite case a condition must be placed on this transformation in order to maintain C^1 continuity across element boundaries.

Consider the line between global nodes $\mathbf{1}$ and $\mathbf{2}$ in the two bicubic Hermite elements shown in Figure 2.11.

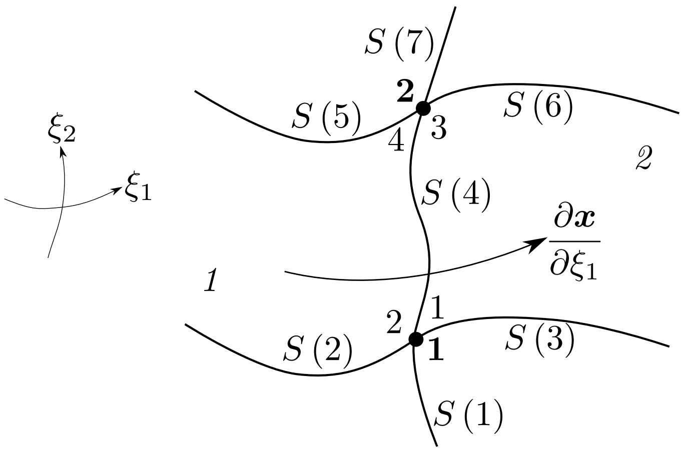

FIGURE 2.11: Two bicubic Hermite elements (denoted by *1* and *2*). The global node numbers are given in boldface, the local node numbers in normal text and the element scale factors used along each line are denoted by $S(l)$.

For C^1 continuity, as opposed to G^1 continuity, between these elements the derivative with respect to ξ_1, that is $\dfrac{\partial \boldsymbol{x}(\xi_2)}{\partial \xi_1}$, must be continuous[1]. The formula for this derivative in element *1* along the boundary between elements *1* and *2* is

$$\frac{\partial \boldsymbol{x}(1,\xi_2)}{\partial \xi_1} = \Psi_0^1(\xi_2)\left.\frac{\partial \boldsymbol{x}}{\partial \xi_1}\right|_2 + \Psi_0^2(\xi_2)\left.\frac{\partial \boldsymbol{x}}{\partial \xi_1}\right|_4 + \Psi_1^1(\xi_2)\left.\frac{\partial^2 \boldsymbol{x}}{\partial \xi_1 \partial \xi_2}\right|_2 + \Psi_1^2(\xi_2)\left.\frac{\partial^2 \boldsymbol{x}}{\partial \xi_1 \partial \xi_2}\right|_4 \tag{2.28}$$

and for element *2* is

$$\frac{\partial \boldsymbol{x}(0,\xi_2)}{\partial \xi_1} = \Psi_0^1(\xi_2)\left.\frac{\partial \boldsymbol{x}}{\partial \xi_1}\right|_1 + \Psi_0^2(\xi_2)\left.\frac{\partial \boldsymbol{x}}{\partial \xi_1}\right|_3 + \Psi_1^1(\xi_2)\left.\frac{\partial^2 \boldsymbol{x}}{\partial \xi_1 \partial \xi_2}\right|_1 + \Psi_1^2(\xi_2)\left.\frac{\partial^2 \boldsymbol{x}}{\partial \xi_1 \partial \xi_2}\right|_3 \tag{2.29}$$

[1]For C^1 continuity the normals either side of an element boundary must be in the same direction *and* have the same magnitude. For G^1 continuity the normals must only have the same direction.

Now substituting Equations (2.26) and (2.27) into the above equations yields for element *1*

$$\begin{aligned}\frac{\partial \boldsymbol{x}\left(1,\xi_2\right)}{\partial \xi_1} &= \Psi_0^1\left(\xi_2\right)\frac{\partial \boldsymbol{x}^2}{\partial s_1}S\left(2\right) + \Psi_0^2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^4}{\partial s_1}S\left(5\right) + \\ &\quad \Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^2}{\partial s_1 \partial s_2}S\left(2\right)S\left(4\right) + \Psi_1^2\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^4}{\partial s_1 \partial s_2}S\left(5\right)S\left(4\right)\end{aligned} \tag{2.30}$$

and for element *2*

$$\begin{aligned}\frac{\partial \boldsymbol{x}\left(0,\xi_2\right)}{\partial \xi_1} &= \Psi_0^1\left(\xi_1\right)\frac{\partial \boldsymbol{x}^1}{\partial s_1}S\left(3\right) + \Psi_0^2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^3}{\partial s_1}S\left(6\right) + \\ &\quad \Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^1}{\partial s_1 \partial s_2}S\left(3\right)S\left(4\right) + \Psi_1^2\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}S\left(6\right)S\left(4\right)\end{aligned} \tag{2.31}$$

Now local node 2 in element *1* and local node 1 in element *2* is the same as global node **1** and local node 4 in element *1* and local node 3 in element *2* is the same as global node **2**. Hence for a given ξ_2 the condition for C^1 continuity across the element boundary is

$$\begin{aligned}&\Psi_0^1\left(\xi_2\right)\frac{\partial \boldsymbol{x}^{\mathbf{1}}}{\partial s_1}S\left(2\right) + \Psi_0^2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^{\mathbf{2}}}{\partial s_1}S\left(5\right) + \Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^{\mathbf{1}}}{\partial s_1 \partial s_2}S\left(2\right)S\left(4\right) \\ &\quad + \Psi_1^2\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^{\mathbf{2}}}{\partial s_1 \partial s_2}S\left(5\right)S\left(4\right) = \Psi_0^1\left(\xi_2\right)\frac{\partial \boldsymbol{x}^{\mathbf{1}}}{\partial s_1}S\left(3\right) + \Psi_0^2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^{\mathbf{2}}}{\partial s_1}S\left(6\right) \\ &\qquad + \Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^{\mathbf{1}}}{\partial s_1 \partial s_2}S\left(3\right)S\left(4\right) + \Psi_1^2\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^{\mathbf{2}}}{\partial s_1 \partial s_2}S\left(6\right)S\left(4\right)\end{aligned} \tag{2.32}$$

or

$$\begin{aligned}&\left(S\left(2\right) - S\left(3\right)\right)\left(\Psi_0^1\left(\xi_2\right)\frac{\partial \boldsymbol{x}^{\mathbf{1}}}{\partial s_1} + \Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^{\mathbf{1}}}{\partial s_1 \partial s_2}S\left(4\right)\right) = \\ &\qquad \left(S\left(6\right) - S\left(5\right)\right)\left(\Psi_0^2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^{\mathbf{2}}}{\partial s_1} + \Psi_1^2\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^{\mathbf{2}}}{\partial s_1 \partial s_2}S\left(4\right)\right)\end{aligned} \tag{2.33}$$

Now by choosing the scale factors to be equal on either side of node **1** and **2** (*i.e.,* $S\left(2\right) = S\left(3\right) = \mathcal{S}\left(\mathbf{1}\right)$ and $S\left(5\right) = S\left(6\right) = \mathcal{S}\left(\mathbf{2}\right)$), that is nodal based scale factors, Equation (2.33) is satisfied for any choice of the scale factors. Hence nodal scale factors are a sufficient condition to ensure C^1 continuity. If it is desired that the scale factors be different either side of the node then Equation (2.33) must be satisfied to ensure continuity.

The choice of the scale factors again determines the ξ to s spacing. We have a number of choices for the scale factor depending on whether or not the ξ to s spacing should favour bigger or smaller elements. One choice which equally favours both elements either side of the node is for the scale factors to be chosen to be nodally based and equal to the average arc-length on either side of the node for each ξ direction *i.e.,* for the l^{th} direction

$$\mathcal{S}(A,l) = \frac{s_l\left(A_{\ominus}(l)\right) + s_l\left(A_{\oplus}(l)\right)}{2} \tag{2.34}$$

where $\mathcal{S}(A,l)$ is the nodal scale factor in the l^{th} ξ direction at global node A, $A_{\ominus}(l)$ is the element immediately preceding (in the l^{th} direction) node A, and $A_{\oplus}(l)$ is the element immediately after (in the l^{th} direction) node A and $s_l(e)$ is the arc-length in the l^{th} ξ direction from node A in element e. This type of scaling is known as *arithmetic mean arc-length scaling*.

Other means can be used *i.e., geometric mean arc-length scaling*

$$\mathcal{S}(A,l) = \sqrt{s_l\left(A_{\ominus}(l)\right) s_l\left(A_{\oplus}(l)\right)} \tag{2.35}$$

or *harmonic mean arc-length scaling*

$$\mathcal{S}(A,l) = \frac{s_l\left(A_{\ominus}(l)\right) s_l\left(A_{\oplus}(l)\right)}{s_l\left(A_{\ominus}(l)\right) + s_l\left(A_{\oplus}(l)\right)} \tag{2.36}$$

Other possible scalings include *minimum arc-length scaling*

$$\mathcal{S}(A,l) = \min\left(s_l\left(A_{\ominus}(l)\right), s_l\left(A_{\oplus}(l)\right)\right) \tag{2.37}$$

and *maximum arc-length scaling*

$$\mathcal{S}(A,l) = \max\left(s_l\left(A_{\ominus}(l)\right), s_l\left(A_{\oplus}(l)\right)\right) \tag{2.38}$$

and *Root Mean Squared (RMS) arc-length scaling*

$$\mathcal{S}(A,l) = \sqrt{\frac{s_l\left(A_{\ominus}(l)\right)^2 + s_l\left(A_{\oplus}(l)\right)^2}{2}} \tag{2.39}$$

To investigate the effect of different types of scaling consider the following test problem.

Consider a fuction $u(x) = A\cos x$ in a one-dimension domain from $x \in [0, L]$. The domain is modelled with a mesh consisting of three cubic Hermite elements. The first of the four nodes is located at $x = 0$, the second at $x = a$, the third at $x = b$ and the final node at $x = L$. The analytic first derivative would be $u^{'}(x) = -A\sin x$ and the analytic second derivative would be $u^{''}(x) = -A\cos x$. The nodal values and derivatives are set to their analytic values.

As a specific example of this test problem let us first consider unevenly spaced elements with $L = 2\pi$, $A = 2$, $a = 1.5$ and $b = 3.5$ is given in Figure 2.12 for the function, Figure 2.13 for the first derivative and Figure 2.14 for the second derivative. Note that the first derivative is discontinuous at element boundaries for arc-length scaling.

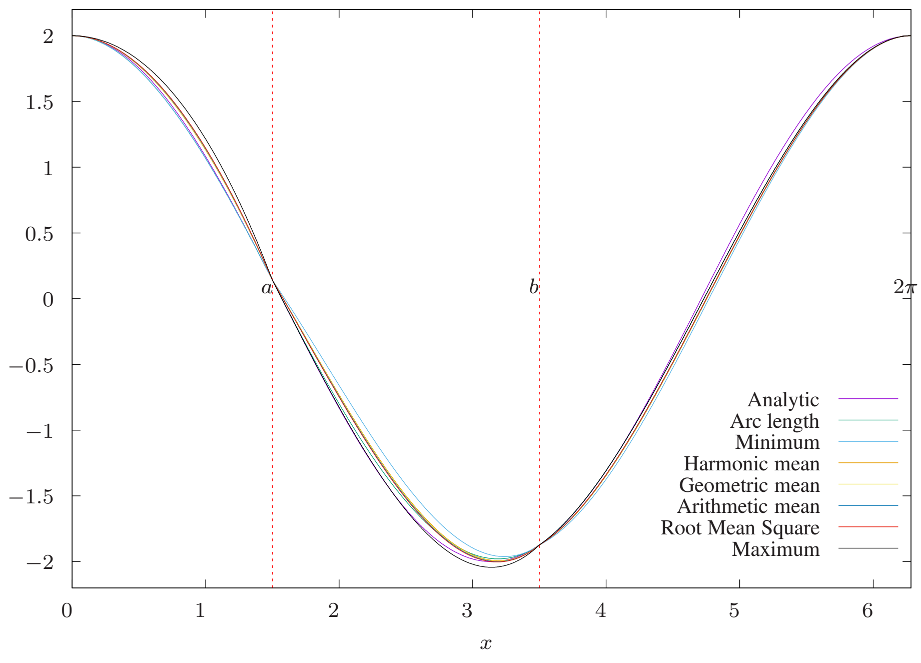

FIGURE 2.12: Function scaling for uneven element spacing.

Now consider this problem with evenly spaced elements *i.e.,* with $a = \frac{2\pi}{3}$ and $b = \frac{4\pi}{3}$. The interpolations are given in Figure 2.15 for the function, Figure 2.16 for the first derivative and Figure 2.17 for the second derivative. Note that the value of the function and the first derivative is continuous at element boundaries for all scaling types.

Let us now consider the influence of scaling type on geometric variables which require normalised arc-length derivatives. For this problem let us consider modelling a circle with three

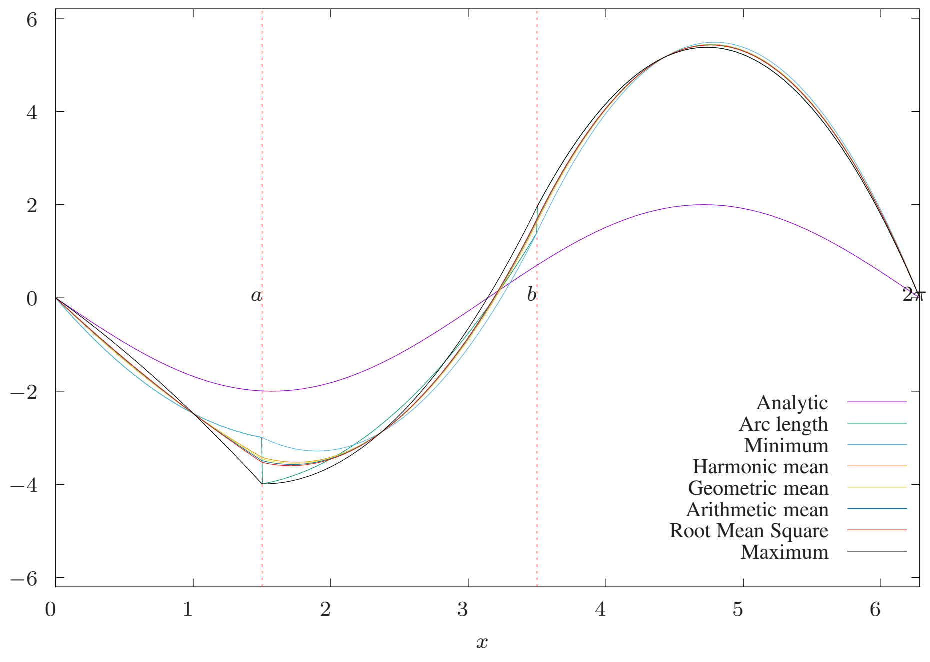

FIGURE 2.13: First derivative scaling for uneven element spacing.

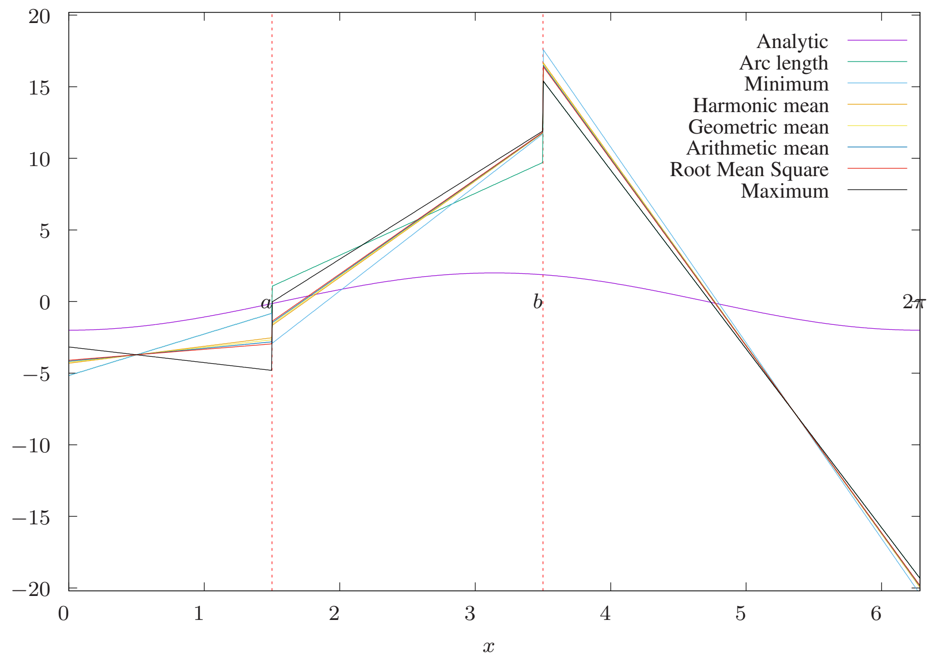

FIGURE 2.14: Second derivative scaling for uneven element spacing.

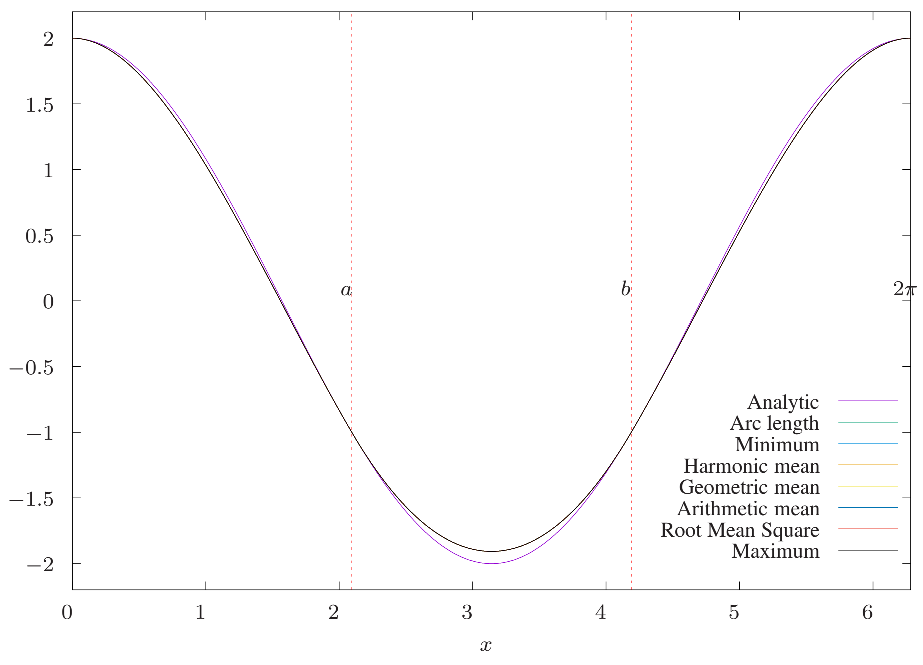

FIGURE 2.15: Function scaling for even element spacing.

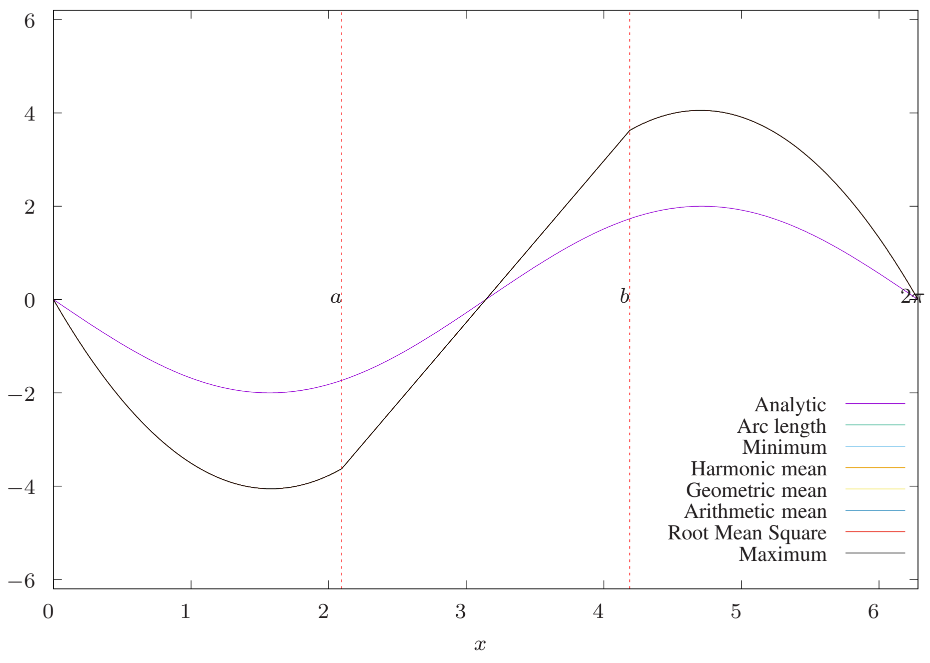

FIGURE 2.16: First derivative scaling for even element spacing.

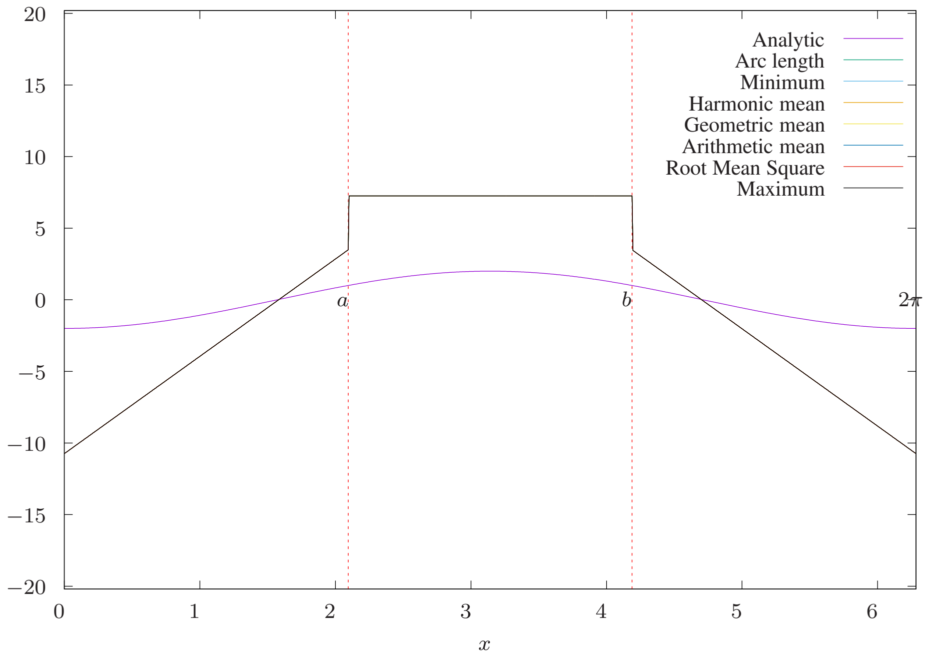

FIGURE 2.17: Second derivative scaling for even element spacing.

cubic Hermite elements. The analytic description is given by $x(\theta) = A\cos\theta$ and $y(\theta) = A\sin\theta$. And the derivatives are given by $\dfrac{\partial x(\theta)}{\partial s} = -\sin\theta$, $\dfrac{\partial y(\theta)}{\partial s} = \cos\theta$, $\dfrac{\partial^2 x(\theta)}{\partial s^2} = -\cos\theta$ and $\dfrac{\partial^2 y(\theta)}{\partial s^2} = -\sin\theta$. The analytic value of the arc-length for an element is given by $s = A\Delta\theta$ where $\Delta\theta$ is the angle subtended by the element.

First let us consider unevenly spaced elements with $A = 2$, $a = 1.5$ and $b = 3.5$. The nodal positions and derivatives are set to their analytic values and the element arc-lengths are set to their analytic value. The interpolation is given in Figure 2.18 for the function, Figure 2.19 for the first derivative and Figure 2.20 for the second derivative. Note that the first derivative is discontinuous at element boundaries for arc-length scaling.

Now consider this problem with evenly spaced elements *i.e.*, with $a = \frac{2\pi}{3}$ and $b = \frac{4\pi}{3}$. The interpolations are given in Figure 2.21 for the function, Figure 2.22 for the first derivative and Figure 2.23 for the second derivative. Note that the value of the function and the first derivative is continuous at element boundaries for all scaling types.

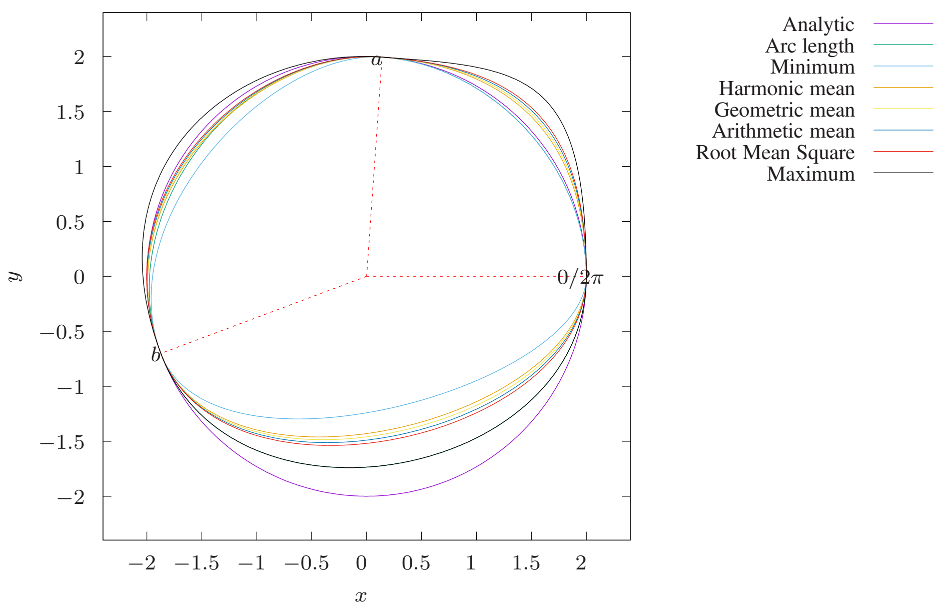

FIGURE 2.18: Geometric circle scaling for uneven element spacing.

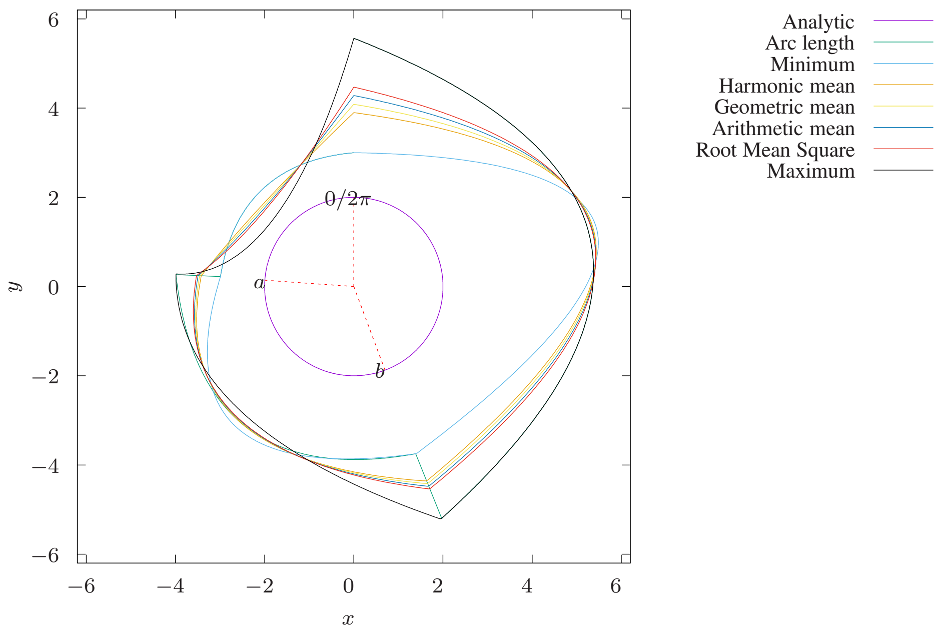

FIGURE 2.19: Geometric circle first derivative scaling for uneven element spacing.

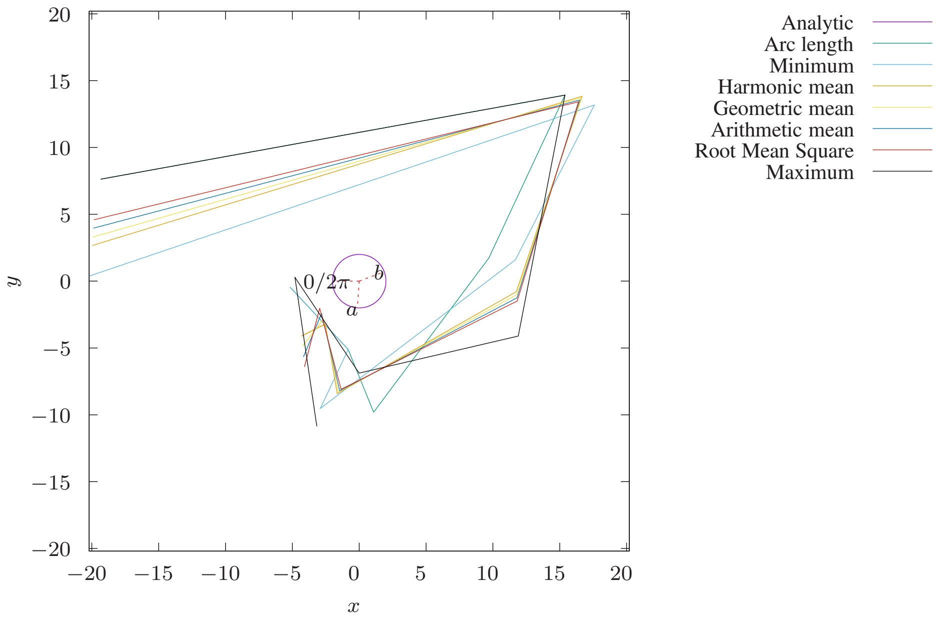

FIGURE 2.20: Geometric circle second derivative scaling for uneven element spacing.

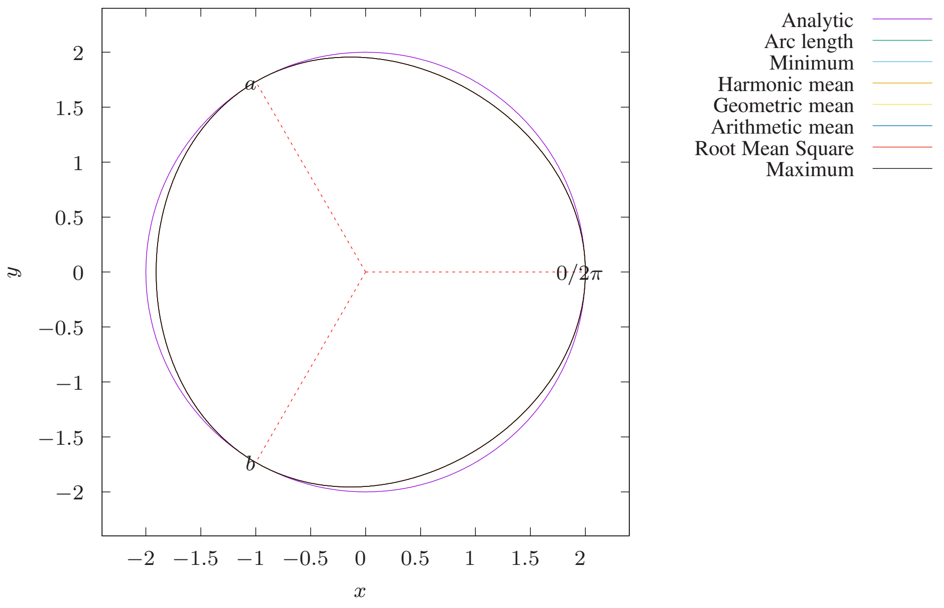

FIGURE 2.21: Geometric circle scaling for even element spacing.

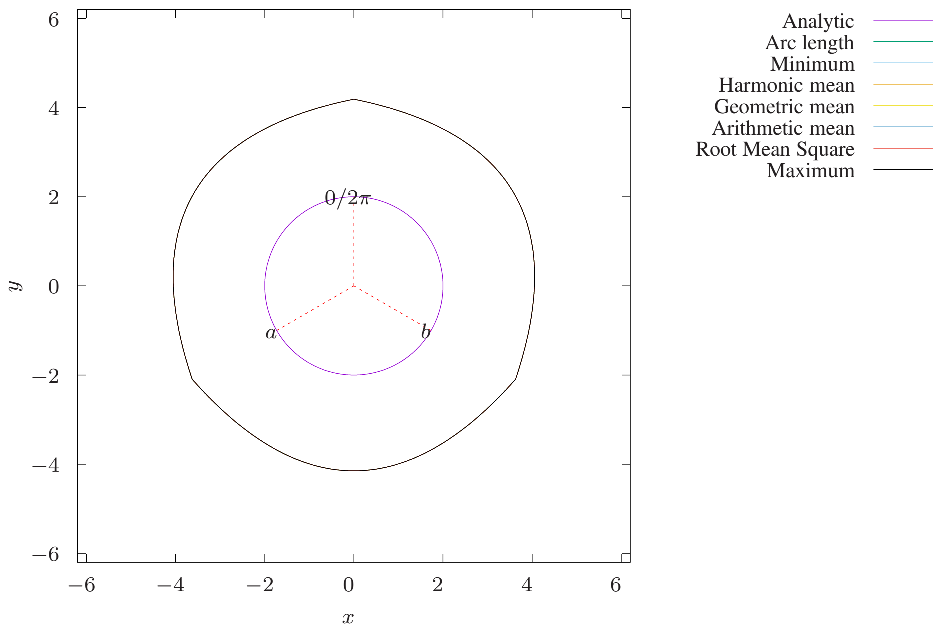

FIGURE 2.22: Geometric circle first derivative scaling for even element spacing.

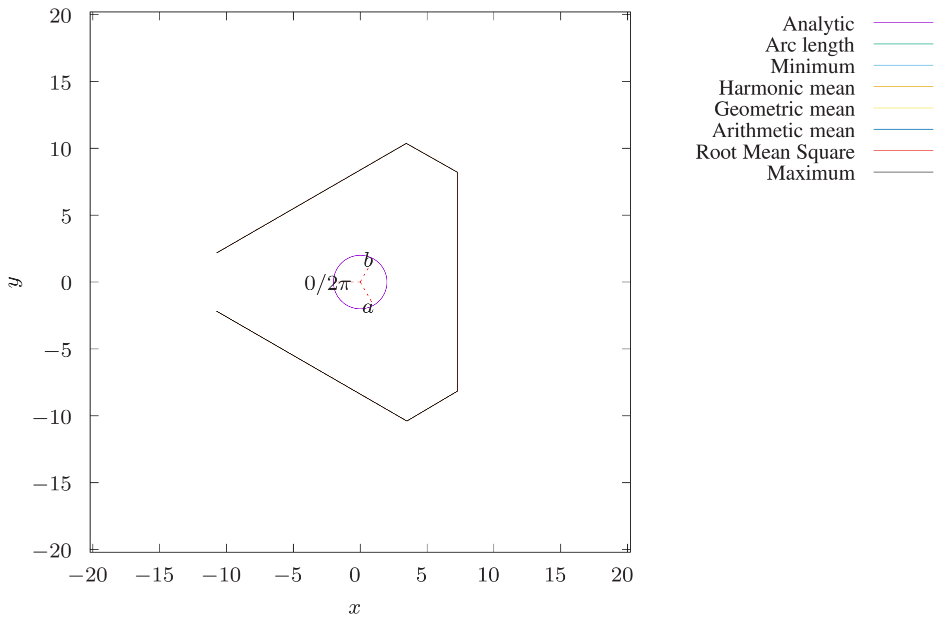

FIGURE 2.23: Geometric circle second derivative scaling for even element spacing.

Hermite-sector elements

One problem that arises when using quadrilateral elements (such as bicubic Hermite elements) to describe a surface is that it is impossible to 'close the surface' in three-dimensions whilst maintaining consistent ξ_1 and ξ_2 directions throughout the mesh. This is important as C^1 continuity requires either consistent ξ directions or a transformation at each node to take into account the inconsistent directions (**?**).

One solution to this problem is to *collapse* a bicubic Hermite element. This entails placing one of the four local nodes of the element at the same geometric location as another local node of the element and results in a triangular element from which it is possible to close the surface. There are two main problems with this solution. The first is that one of the two ξ directions at the collapsed node is undefined. The second is that the distance between the two nodes at the same location is zero. Numerical problems can result from this zero distance. An alternative strategy has been developed in which special elements, called "Hermite-sector" elements, are used to close a bicubic Hermite surface in three-dimensions. There are two types of elements depending on whether the ξ (or s) directions come together at local node one or local node three. These two elements are shown in Figure 2.24.

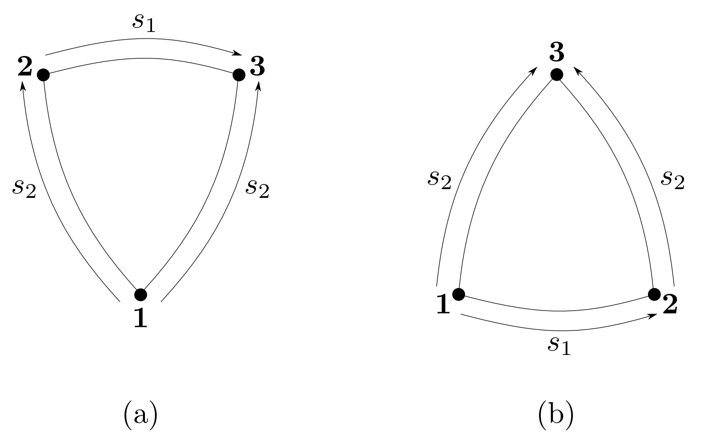

FIGURE 2.24: Hermite-sector elements. (a) Apex node one element. (b) Apex node three element.

From Figure 2.24 it can be seen that the s_2 direction is not unique at the apex nodes. This gives us two choices for the interpolation within the element: ignore the s_2 derivative when interpolating or set the s_2 derivative identically to zero.

Ignore s_2 apex derivative: For this case it can be seen from Figure 2.24 that the interpolation in the ξ_1 direction is just the standard cubic Hermite interpolation. The interpolation in the ξ_2 direction is now a little different in that the nodal arc-length derivative has been dropped as it is no longer defined at the apex node. For an apex node one element shown in Figure 2.24(a) the interpolation for the line between local node one and local node n is now quadratic and is given by

$$\boldsymbol{x}\left(\xi_2\right) = \zeta_1\left(\xi_2\right)\boldsymbol{x}^1 + \zeta_2\left(\xi_2\right)\boldsymbol{x}^n + \zeta_3\left(\xi_2\right)\left.\frac{\partial \boldsymbol{x}}{\partial \xi_2}\right|_n \tag{2.40}$$

with the basis functions given by

$$\begin{aligned}\zeta_1\left(\xi\right) &= \left(\xi - 1\right)^2\\ \zeta_2\left(\xi\right) &= 2\xi - \xi^2\\ \zeta_3\left(\xi\right) &= \xi^2 - \xi\end{aligned} \tag{2.41}$$

For the apex node three element shown in Figure 2.24(b) the interpolation for the line connecting local node n with local node three is given by

$$\boldsymbol{x}\left(\xi_2\right) = \eta_1\left(\xi_2\right)\boldsymbol{x}^3 + \eta_2\left(\xi_2\right)\boldsymbol{x}^n + \eta_3\left(\xi_2\right)\left.\frac{\partial \boldsymbol{x}}{\partial \xi_2}\right|_n \tag{2.42}$$

with the basis functions given by

$$\begin{aligned}\eta_1\left(\xi\right) &= \xi^2\\ \eta_2\left(\xi\right) &= 1 - \xi^2\\ \eta_3\left(\xi\right) &= \xi - \xi^2\end{aligned} \tag{2.43}$$

The full interpolation formula for the sector element can then be found by taking the tensor product of the interpolation in the ξ_1 direction, given in Equation (2.13), with the interpolation in the ξ_2 direction (given by either Equations (2.40) or (2.42)). The interpolation formula can be converted from nodal ξ derivatives to nodal arc-length derivatives using the procedure outlined for the bicubic Hermite case. For example, the interpolation formulae for an apex node one element is

$$\begin{aligned}\boldsymbol{x}\left(\xi_1,\xi_2\right) = {} & \zeta_1\left(\xi_2\right)\boldsymbol{x}^1 + \Psi_1^0\left(\xi_1\right)\zeta_2\left(\xi_2\right)\boldsymbol{x}^2 + \Psi_2^0\left(\xi_1\right)\zeta_2\left(\xi_2\right)\boldsymbol{x}^3 + \\ & \Psi_1^1\left(\xi_1\right)\zeta_2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^2}{\partial s_1}\mathcal{S}\left(2,1\right) + \Psi_2^1\left(\xi_1\right)\zeta_2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^3}{\partial s_1}\mathcal{S}\left(3,1\right) + \\ & \Psi_1^0\left(\xi_1\right)\zeta_3\left(\xi_2\right)\frac{\partial \boldsymbol{x}^2}{\partial s_2}\mathcal{S}\left(2,2\right) + \Psi_2^0\left(\xi_1\right)\zeta_3\left(\xi_2\right)\frac{\partial \boldsymbol{x}^3}{\partial s_2}\mathcal{S}\left(3,2\right) + \\ & \Psi_1^1\left(\xi_1\right)\zeta_3\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^2}{\partial s_1 \partial s_2}\mathcal{S}\left(2,1\right)\mathcal{S}\left(2,2\right) + \Psi_2^1\left(\xi_1\right)\zeta_3\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\mathcal{S}\left(3,1\right)\mathcal{S}\left(3,2\right)\end{aligned} \tag{2.44}$$

Care must be taken when using Hermite-sector elements for rapidly changing surfaces. Consider an apex node one element with undefined s_2 apex derivatives. The rate of change of $\boldsymbol{x}$ with respect to ξ_1 along the line from node one to node three (*i.e.,* $\xi_1 = 1$) is

$$\begin{aligned}\frac{\partial \boldsymbol{x}\left(1,\xi_2\right)}{\partial \xi_1} &= \zeta_2\left(\xi_2\right)\frac{\partial \boldsymbol{x}^3}{\partial s_1}\mathcal{S}\left(3,1\right) + \zeta_3\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\mathcal{S}\left(3,1\right)\mathcal{S}\left(3,2\right) \\ &= \mathcal{S}\left(3,1\right)\left(\left(2\xi_2 - \xi_2^2\right)\frac{\partial \boldsymbol{x}^3}{\partial s_1} + \left(\xi_2^2 - \xi_2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\mathcal{S}\left(3,2\right)\right)\end{aligned} \tag{2.45}$$

Taking the dot product of $\dfrac{\partial \boldsymbol{x}\left(1,\xi_2\right)}{\partial \xi_1}$ with $\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}$ gives

$$\frac{\partial \boldsymbol{x}\left(1,\xi_2\right)}{\partial \xi_1}\cdot\frac{\partial \boldsymbol{x}^3}{\partial s_1} = \mathcal{S}\left(3,1\right)\left(\left(2\xi_2 - \xi_2^2\right)\frac{\partial \boldsymbol{x}^3}{\partial s_1}\cdot\frac{\partial \boldsymbol{x}^3}{\partial s_1} + \left(\xi_2^2 - \xi_2\right)\mathcal{S}\left(3,2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\frac{\partial \boldsymbol{x}^3}{\partial s_1}\right) \tag{2.46}$$

The normality constraint for arc-length derivatives means that $\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}\cdot\dfrac{\partial \boldsymbol{x}^3}{\partial s_1} = 1$ and thus the right hand side of Equation (2.46) divided by $\mathcal{S}\left(3,1\right)$ (*i.e.,* normalised by $\mathcal{S}\left(3,1\right)$) is the quadratic

$$\left(2\xi_2 - \xi_2^2\right) + \left(\xi_2^2 - \xi_2\right)\mathcal{S}\left(3,2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\frac{\partial \boldsymbol{x}^3}{\partial s_1}$$

or

$$\left(\mathcal{S}\left(3,2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\frac{\partial \boldsymbol{x}^3}{\partial s_1} - 1\right)\xi_2^2 + \left(2 - \mathcal{S}\left(3,2\right)\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\frac{\partial \boldsymbol{x}^3}{\partial s_1}\right)\xi_2$$

This quadratic is 1 at $\xi_2 = 1$ and always has a root at $\xi_2 = 0$. Consider the case of this quadratic having its second root in the interval $(0,1)$. This would mean that at some point in the interval $(0,1)$ the dot product of $\dfrac{\partial \boldsymbol{x}(1,\xi_2)}{\partial \xi_1}$ and $\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}$ would go from zero to negative and then positive as ξ_2 changed from 0 to 1 *i.e.,* the angle between $\dfrac{\partial \boldsymbol{x}(1,\xi_2)}{\partial \xi_1}$ and $\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}$ would, at some stage, be greater than ninety degrees. As the direction of the normal to the surface along the line between local node one and three is given by the cross product of $\dfrac{\partial \boldsymbol{x}(1,\xi_2)}{\partial \xi_1}$ and $\dfrac{\partial \boldsymbol{x}(1,\xi_2)}{\partial \xi_2}$ then, if the quadratic became sufficiently negative, the normal to the surface could reverse direction from an outward to an inward normal as ξ_2 changed from 0 to 1. This is clearly undesirable. In fact even if the quadratic is only slightly negative the resulting surface would be grossly deformed.

To avoid these effects the second root of the quadratic must be outside the interval $(0,1)$. From the quadratic formula the conditions for this are

$$\frac{\mathcal{S}(3,2)\,\dfrac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}-2}{\mathcal{S}(3,2)\,\dfrac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}-1}<0 \tag{2.47}$$

and

$$\frac{\mathcal{S}(3,2)\,\dfrac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}-2}{\mathcal{S}(3,2)\,\dfrac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\cdot\dfrac{\partial \boldsymbol{x}^3}{\partial s_1}-1}>1 \tag{2.48}$$

that is (for the line from local node one to local node n)

$$\frac{\partial^2 \boldsymbol{x}^n}{\partial s_1 \partial s_2}\cdot\frac{\partial \boldsymbol{x}^n}{\partial s_1}<\frac{2}{\mathcal{S}(n,2)} \tag{2.49}$$

The simplest way to interpret this constraint is that if the element is large (*i.e.,* $\mathcal{S}(n,2)$ is large) then $\dfrac{\partial^2 \boldsymbol{x}^n}{\partial s_1 \partial s_2}\cdot\dfrac{\partial \boldsymbol{x}^n}{\partial s_1}$ must be small. The simplest way for this to happen is to ensure the magnitude of the components of $\dfrac{\partial^2 \boldsymbol{x}^n}{\partial s_1 \partial s_2}$ are small (or of opposite sign to the comparable components of $\dfrac{\partial \boldsymbol{x}^n}{\partial s_1}$).

The equivalent interpolation formula to Equation (2.44) for an apex node three Hermite-sector element is

$$\begin{aligned}\boldsymbol{x}(\xi_1,\xi_2) = {} & \Psi_1^0(\xi_1)\,\eta_2(\xi_2)\,\boldsymbol{x}^1 + \Psi_2^0(\xi_1)\,\eta_2(\xi_2)\,\boldsymbol{x}^2 + \eta_1(\xi_2)\,\boldsymbol{x}^3 + \\ & \Psi_1^1(\xi_1)\,\eta_2(\xi_2)\,\frac{\partial \boldsymbol{x}^1}{\partial s_1}\mathcal{S}(1,1) + \Psi_2^1(\xi_1)\,\eta_2(\xi_2)\,\frac{\partial \boldsymbol{x}^2}{\partial s_1}\mathcal{S}(2,1) + \\ & \Psi_1^0(\xi_1)\,\eta_3(\xi_2)\,\frac{\partial \boldsymbol{x}^1}{\partial s_2}\mathcal{S}(1,2) + \Psi_2^0(\xi_1)\,\eta_3(\xi_2)\,\frac{\partial \boldsymbol{x}^2}{\partial s_2}\mathcal{S}(2,2) + \\ & \Psi_1^1(\xi_1)\,\eta_3(\xi_2)\,\frac{\partial^2 \boldsymbol{x}^1}{\partial s_1 \partial s_2}\mathcal{S}(1,1)\,\mathcal{S}(1,2) + \Psi_2^1(\xi_1)\,\eta_3(\xi_2)\,\frac{\partial^2 \boldsymbol{x}^2}{\partial s_1 \partial s_2}\mathcal{S}(2,1)\,\mathcal{S}(2,2)\end{aligned} \tag{2.50}$$

and the equivalent constraint for apex node three Hermite-sector elements (for the line from local node n to local node three) is

$$\frac{\partial^2 \boldsymbol{x}^n}{\partial s_1 \partial s_2} \cdot \frac{\partial \boldsymbol{x}^n}{\partial s_1} > \frac{-2}{\mathcal{S}(n,2)} \tag{2.51}$$

Zero s_2 apex derivative: For this case the sector basis functions are just the cubic Hermite basis functions. The corresponding interpolation formulae for an apex node one element is hence

$$\begin{aligned}\boldsymbol{x}(\xi_1,\xi_2) = {} & \Psi_1^0(\xi_2)\,\boldsymbol{x}^1 + \Psi_1^0(\xi_1)\,\Psi_2^0(\xi_2)\,\boldsymbol{x}^2 + \Psi_2^0(\xi_1)\,\Psi_2^0(\xi_2)\,\boldsymbol{x}^3 + \\ & \Psi_1^1(\xi_1)\,\Psi_2^0(\xi_2)\,\frac{\partial \boldsymbol{x}^2}{\partial s_1}\mathcal{S}(2,1) + \Psi_2^1(\xi_1)\,\Psi_2^0(\xi_2)\,\frac{\partial \boldsymbol{x}^3}{\partial s_1}\mathcal{S}(3,1) + \\ & \Psi_1^0(\xi_1)\,\Psi_2^1(\xi_2)\,\frac{\partial \boldsymbol{x}^2}{\partial s_2}\mathcal{S}(2,2) + \Psi_2^0(\xi_1)\,\Psi_2^1(\xi_2)\,\frac{\partial \boldsymbol{x}^2}{\partial s_2}\mathcal{S}(3,2) + \\ & \Psi_1^1(\xi_1)\,\Psi_2^1(\xi_2)\,\frac{\partial^2 \boldsymbol{x}^2}{\partial s_1 \partial s_2}\mathcal{S}(2,1)\,\mathcal{S}(2,2) + \Psi_2^1(\xi_1)\,\Psi_2^1(\xi_2)\,\frac{\partial^2 \boldsymbol{x}^3}{\partial s_1 \partial s_2}\mathcal{S}(3,1)\,\mathcal{S}(3,2)\end{aligned} \tag{2.52}$$

and the condition to avoid reversal of the normal is

$$\frac{\partial^2 \boldsymbol{x}^n}{\partial s_1 \partial s_2} \cdot \frac{\partial \boldsymbol{x}^n}{\partial s_1} < \frac{3}{\mathcal{S}(n,2)} \tag{2.53}$$

and for the apex node three element the interpolation formula is

$$\begin{aligned}\boldsymbol{x}\left(\xi_1,\xi_2\right) = {} & \Psi_1^0\left(\xi_1\right)\Psi_2^0\left(\xi_2\right)\boldsymbol{x}^1 + \Psi_2^0\left(\xi_1\right)\Psi_2^0\left(\xi_2\right)\boldsymbol{x}^2 + \Psi_2^0\left(\xi_2\right)\boldsymbol{x}^3 + \\ & \Psi_1^1\left(\xi_1\right)\Psi_1^0\left(\xi_2\right)\frac{\partial \boldsymbol{x}^1}{\partial s_1}\mathcal{S}\left(1,1\right) + \Psi_2^1\left(\xi_1\right)\Psi_1^0\left(\xi_2\right)\frac{\partial \boldsymbol{x}^2}{\partial s_1}\mathcal{S}\left(2,1\right) + \\ & \Psi_1^0\left(\xi_1\right)\Psi_1^1\left(\xi_2\right)\frac{\partial \boldsymbol{x}^1}{\partial s_2}\mathcal{S}\left(1,2\right) + \Psi_2^0\left(\xi_1\right)\Psi_1^1\left(\xi_2\right)\frac{\partial \boldsymbol{x}^2}{\partial s_2}\mathcal{S}\left(2,2\right) + \\ & \Psi_1^1\left(\xi_1\right)\Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^1}{\partial s_1 \partial s_2}\mathcal{S}\left(1,1\right)\mathcal{S}\left(1,2\right) + \Psi_2^1\left(\xi_1\right)\Psi_1^1\left(\xi_2\right)\frac{\partial^2 \boldsymbol{x}^2}{\partial s_1 \partial s_2}\mathcal{S}\left(2,1\right)\mathcal{S}\left(2,2\right)\end{aligned} \tag{2.54}$$

with a condition of

$$\frac{\partial^2 \boldsymbol{x}^n}{\partial s_1 \partial s_2}\cdot\frac{\partial \boldsymbol{x}^n}{\partial s_1} > \frac{-3}{\mathcal{S}\left(n,2\right)} \tag{2.55}$$

Although the Hermite-sector basis function in which the s_2 apex node derivatives are identically zero have an increased limit on the cross-derivative constraints (a right hand side numerator of ± 3 instead of ± 2) they have the problem that as all derivatives vanish at the apex any interpolated function has a zero Hessian at the apex. As this can cause numerical problems the Hermite-sector basis functions which have an undefined s_2 derivative are prefered.

2.3.3 Simplex Basis Functions

Simplex basis function and its derivatives are evaluated with respect to external $\boldsymbol{\xi}$ coordinates.

For Simplex line elements there are two area coordinates which are a function of ξ_1 *i.e.,*

$$L_1 = 1 - \xi_1 \tag{2.56}$$

$$L_2 = \xi_1 - 1 \tag{2.57}$$

The derivatives wrt to external coordinates are then given by

$$\frac{\partial \boldsymbol{N}}{\partial \xi_1} = \frac{\partial \boldsymbol{N}}{\partial L_2} - \frac{\partial \boldsymbol{N}}{\partial L_1} \tag{2.58}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial {\xi_1}^2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_1}^2} - 2\frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_2} + \frac{\partial^2 \boldsymbol{N}}{\partial {L_2}^2} \tag{2.59}$$

For Simplex triangle elements there are three area coordinates which are a function of ξ_1 and ξ_2 *i.e.,*

$$L_1 = 1 - \xi_1 \tag{2.60}$$

$$L_2 = 1 - \xi_2 \tag{2.61}$$

$$L_3 = \xi_1 + \xi_2 - 1 \tag{2.62}$$

The derivatives wrt to external coordinates are then given by

$$\frac{\partial \boldsymbol{N}}{\partial \xi_1} = \frac{\partial \boldsymbol{N}}{\partial L_3} - \frac{\partial \boldsymbol{N}}{\partial L_1} \tag{2.63}$$

$$\frac{\partial \boldsymbol{N}}{\partial \xi_2} = \frac{\partial \boldsymbol{N}}{\partial L_3} - \frac{\partial \boldsymbol{N}}{\partial L_2} \tag{2.64}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial {\xi_1}^2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_1}^2} - 2\frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_3} + \frac{\partial^2 \boldsymbol{N}}{\partial {L_3}^2} \tag{2.65}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial {\xi_2}^2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_2}^2} - 2\frac{\partial^2 \boldsymbol{N}}{\partial L_2 \partial L_3} + \frac{\partial^2 \boldsymbol{N}}{\partial {L_3}^2} \tag{2.66}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial \xi_1 \partial \xi_2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_3}^2} - \frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_3} - \frac{\partial^2 \boldsymbol{N}}{\partial L_2 \partial L_3} + \frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_2} \tag{2.67}$$

For Simplex tetrahedral elements there are four area coordinates which are a function of ξ_1, ξ_2 and ξ_3 *i.e.,*

$$L_1 = 1 - \xi_1 \tag{2.68}$$

$$L_2 = 1 - \xi_2 \tag{2.69}$$

$$L_3 = 1 - \xi_3 \tag{2.70}$$

$$L_4 = \xi_1 + \xi_2 + \xi_3 - 2 \tag{2.71}$$

The derivatives wrt to external coordinates are then given by

$$\frac{\partial \boldsymbol{N}}{\partial \xi_1} = \frac{\partial \boldsymbol{N}}{\partial L_4} - \frac{\partial \boldsymbol{N}}{\partial L_1} \tag{2.72}$$

$$\frac{\partial \boldsymbol{N}}{\partial \xi_2} = \frac{\partial \boldsymbol{N}}{\partial L_4} - \frac{\partial \boldsymbol{N}}{\partial L_2} \tag{2.73}$$

$$\frac{\partial \boldsymbol{N}}{\partial \xi_3} = \frac{\partial \boldsymbol{N}}{\partial L_4} - \frac{\partial \boldsymbol{N}}{\partial L_3} \tag{2.74}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial {\xi_1}^2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_1}^2} - 2\frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_4} + \frac{\partial^2 \boldsymbol{N}}{\partial {L_4}^2} \tag{2.75}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial {\xi_2}^2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_2}^2} - 2\frac{\partial^2 \boldsymbol{N}}{\partial L_2 \partial L_4} + \frac{\partial^2 \boldsymbol{N}}{\partial {L_4}^2} \tag{2.76}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial {\xi_3}^2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_3}^2} - 2\frac{\partial^2 \boldsymbol{N}}{\partial L_3 \partial L_4} + \frac{\partial^2 \boldsymbol{N}}{\partial {L_4}^2} \tag{2.77}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial \xi_1 \partial \xi_2} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_4}^2} - \frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_4} - \frac{\partial^2 \boldsymbol{N}}{\partial L_2 \partial L_4} + \frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_2} \tag{2.78}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial \xi_1 \partial \xi_3} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_4}^2} - \frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_4} - \frac{\partial^2 \boldsymbol{N}}{\partial L_3 \partial L_4} + \frac{\partial^2 \boldsymbol{N}}{\partial L_1 \partial L_3} \tag{2.79}$$

$$\frac{\partial^2 \boldsymbol{N}}{\partial \xi_2 \partial \xi_3} = \frac{\partial^2 \boldsymbol{N}}{\partial {L_4}^2} - \frac{\partial^2 \boldsymbol{N}}{\partial L_2 \partial L_4} - \frac{\partial^2 \boldsymbol{N}}{\partial L_3 \partial L_4} + \frac{\partial^2 \boldsymbol{N}}{\partial L_2 \partial L_3} \tag{2.80}$$

$$\frac{\partial^3 \boldsymbol{N}}{\partial \xi_1 \partial \xi_2 \partial \xi_3} = \frac{\partial^3 \boldsymbol{N}}{\partial L_4^3} - \frac{\partial^3 \boldsymbol{N}}{\partial L_1 \partial L_4^2} - \frac{\partial^3 \boldsymbol{N}}{\partial L_2 \partial L_4^2} - \frac{\partial^3 \boldsymbol{N}}{\partial L_3 \partial L_4^2} + \tag{2.81}$$

$$\frac{\partial^3 \boldsymbol{N}}{\partial L_1 \partial L_2 \partial L_4} + \frac{\partial^3 \boldsymbol{N}}{\partial L_1 \partial L_3 \partial L_4} + \frac{\partial^3 \boldsymbol{N}}{\partial L_2 \partial L_3 \partial L_4} - \frac{\partial^3 \boldsymbol{N}}{\partial L_1 \partial L_2 \partial L_3} \tag{2.82}$$

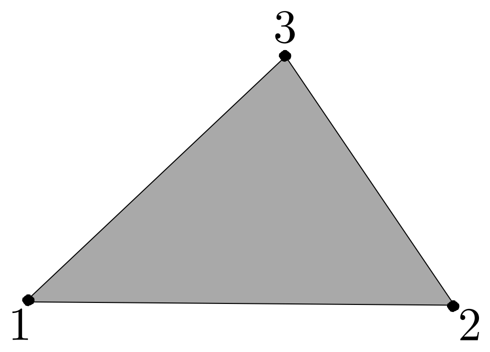

FIGURE 2.25: Linear triangle local node ordering.

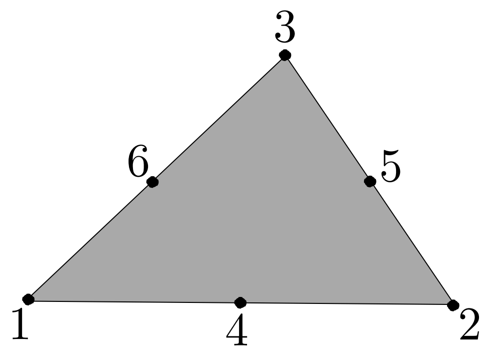

FIGURE 2.26: Quadratic triangle local node ordering.

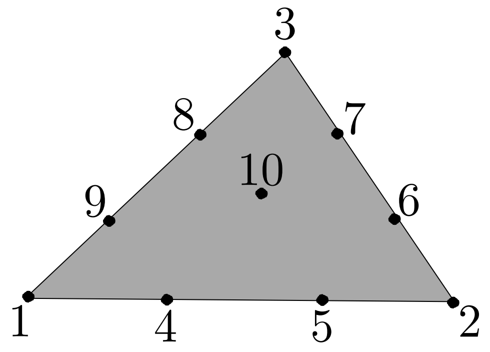

FIGURE 2.27: Cubic triangle local node ordering.

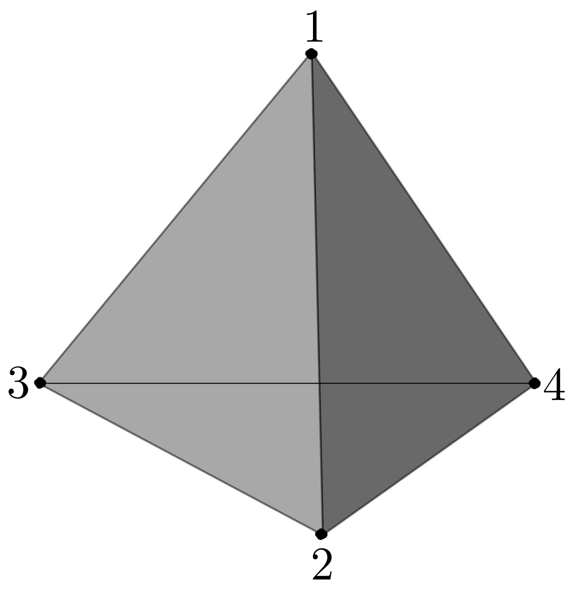

FIGURE 2.28: Linear tetrahedron local node ordering.

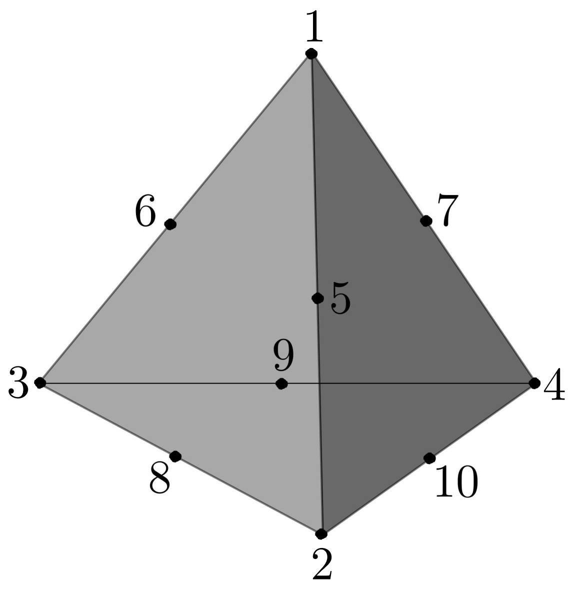

FIGURE 2.29: Qudadratic tetrahedron local node ordering.

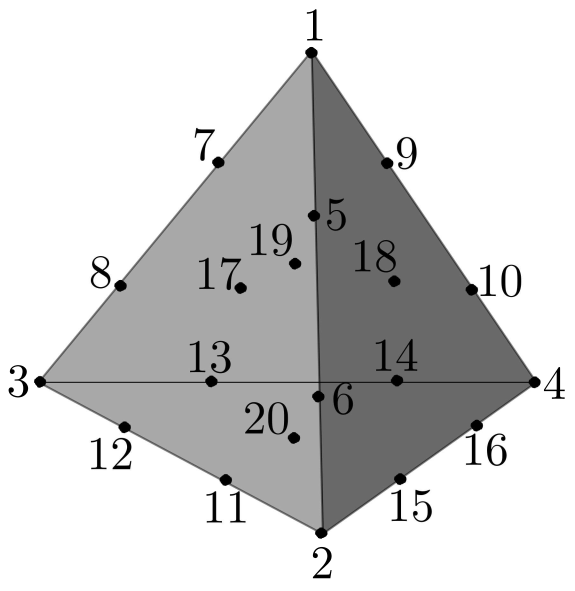

FIGURE 2.30: Cubic tetrahedron local node ordering.

2.3.4 General linear map

From the definition of Christoffel symbols

$$\frac{\partial^2 x^p}{\partial \xi^j \partial \xi^k} = \Gamma^m_{kj} \frac{\partial x^p}{\partial \xi^m} \tag{2.83}$$

i.e.,

$$\Gamma^i_{kj} = \frac{\partial^2 x^l}{\partial \xi^k \partial \xi^j} \frac{\partial \xi^i}{\partial x^l} \tag{2.84}$$

or in vector form

$$\Gamma^i_{jk} = \frac{\partial^2 \boldsymbol{x}}{\partial \xi^k \partial \xi^j} \cdot \nabla \xi^i \tag{2.85}$$

Chapter 3

Differential Geometry

3.1 Introduction

3.2 Topology

The topology of a space specifies the connectedness, continuity, compactness, and other properties that can be defined on the space of specified dimension. It brings in the concept of open neighbourhoods of points. Types of topology include

1. Discrete (OD) topology, $\mathbb{D}$.
2. Real topology, $\mathbb{R}^n$.
3. Spherical topology, $\mathbb{S}^n$.

3.3 Manifold

A manifold $\mathfrak{M}$ is a topological space (*i.e.,* a set of points with a specified topology) of a fixed dimension n in which every point is homeomorphic to $\mathbb{R}^n$ *i.e.,* they admit a coordinate system that is locally Euclidean. For FieldML purposes we need to relax this last restriction because we wish to allow branching structures, but we will still refer to this as a manifold the neighbourhoods of points at the branch do not admit a coordinate system that is locally Euclidean.

3.4 Vectors and Covectors

To motivate this section consider standing on a hill. Now, consider two vectors that can be applied. If we move around the hill at constant height we can talk about our velocity vector around the hill. We can also talk about moving down the slope of the hill. The gradient vector would give the steepest direction down the hill. If both these vectors had a unit magnitude in what ever units of measurement we are using we might consider these vectors to be similar (apart from the obvious difference in direction). Now consider a change of units *e.g.,* from m to cm. If our velocity was prevoiusly $1\ \mathrm{m\ s}^{-1}$ then it would be now $100\ \mathrm{cm\ s}^{-1}$. However, our gradient vector that was previously $1\ \mathrm{m}^{-1}$ would become $0.01\ \mathrm{cm}^{-1}$. If these two "vectors" are really so similar why do the transform differently under a change of units/coordinates? The reason is that these two "vectors" are not actually that similar at all. Indeed, one of them is not a vector at all but rather it is a *covector*.

3.5 Differential geometry of manifolds

3.5.1 Connections

3.5.2 Covariant Differentiation

A covariant derivative is a derivative along a tangent vector of a manifold. It is a generalisation of a directional derivative to manifolds. It is a derivative that under a coordinate transforms covariantly *i.e.,* linearly with the Jacobian. A covariant derivative is equivalent to the idea of a connection. For a connection ∇ and a vector field X then ∇_X is the covariant derivative.

The covariant derivative of a (r, s) tensor is an $(r, s+1)$ tensor defined by

$$\nabla_X Y = \nabla_X \left(X^i e_i\right) = Y\left(X^i\right) e_i + X^i \nabla_Y e_i \tag{3.1}$$

The covariant derivative has the following properties

Double covariant differentiation

For an (r, s) tensor field, T, the double covariant derivative is

$$\nabla^2 T = \nabla\nabla T \tag{3.2}$$

which is an $(r, s+2)$ tensor field. It is defined by

$$\nabla^2_{X,Y} T = \nabla_X \nabla_Y T - \nabla_{\nabla_X Y} T \tag{3.3}$$

3.6 Tensor Analysis

3.6.1 Base vectors

Now, if we have a vector, $\boldsymbol{v}$ we can write

$$\boldsymbol{v} = v^i \boldsymbol{g}_i \tag{3.4}$$

where v^i are the components of the contravariant vector, and $\boldsymbol{g}_i$ are the covariant base vectors.

Similarly, the vector $\boldsymbol{v}$ can also be written as

$$\boldsymbol{v} = v_i \boldsymbol{g}^i \tag{3.5}$$

where v_i are the components of the covariant vector, and $\boldsymbol{g}^i$ are the contravariant base vectors.

We now note that

$$\boldsymbol{v} = v^i \boldsymbol{g}_i = v^i \sqrt{g_{ii}} \hat{\boldsymbol{g}}_i \tag{3.6}$$

where $v^i \sqrt{g_{ii}}$ are the physical components of the vector and $\hat{\boldsymbol{g}}_i$ are the unit vectors given by

$$\hat{\boldsymbol{g}}_i = \frac{\boldsymbol{g}_i}{\sqrt{g_{ii}}} \tag{3.7}$$

3.6.2 Metric Tensors

Metric tensors are the inner product of base vectors. If $\boldsymbol{g}_i$ are the covariant base vectors then the covariant metric tensor is given by

$$g_{ij} = \boldsymbol{g}_i \cdot \boldsymbol{g}_j \tag{3.8}$$

Similarily if $\boldsymbol{g}^i$ are the contravariant base vectors then the contravariant metric tensor is given by

$$g^{ij} = \boldsymbol{g}^i \cdot \boldsymbol{g}^j \tag{3.9}$$

We can also form a mixed metric tensor from the dot product of a contravariant and a covariant base vector *i.e.,*

$$g^{i}_{.j} = \boldsymbol{g}^i \cdot \boldsymbol{g}_j \tag{3.10}$$

and

$$g_{i}^{.j} = \boldsymbol{g}_i \cdot \boldsymbol{g}^j \tag{3.11}$$

Note that for mixed tensors the "." indicates the order of the index *i.e.,* $g^{i}_{.j}$ indicates that the first index is contravariant and the second index is covariant whereas $g_{i}^{.j}$ indicates that the first index is covariant and the second index is contravariant.

If the base vectors are all mutually orthogonal and constant then $\boldsymbol{g}_i = \boldsymbol{g}^i$ and $g_{ij} = g^{ij}$.

The metric tensors generalise (Euclidean) distance *i.e.,*

$$ds^2 = g_{ij}dx^i dx^j \tag{3.12}$$

Raising and lowering indices

Note that multiplying by the covariant metric tensor lowers indices *i.e.,*

$$\begin{aligned} \boldsymbol{A}_i &= g_{ij}\boldsymbol{A}^j \\ A_{ij} &= g_{ik}g_{jl}A^{kl} = g_{jk}A_{i}^{.k} = g_{ik}A^{k}_{.j} \end{aligned} \tag{3.13}$$

and that multiplying by the contravariant metric tensor raises indices *i.e.,*

$$\begin{aligned} \boldsymbol{A}^i &= g^{ij}\boldsymbol{A}_j \\ A^{ij} &= g^{ik}g^{jl}A_{kl} = g^{ik}A_{k}^{.j} = g^{jk}A^{i}_{.k} \end{aligned} \tag{3.14}$$

and for the mixed tensors

$$\begin{aligned} A_i^{\cdot j} &= g^{jk} A_{ik} = g_{ik} A^{kj} \\ A^i_{\cdot j} &= g^{ik} A_{kj} = g_{jk} A^{ik} \end{aligned} \tag{3.15}$$

We can denote a tensor in which all indicies have been raised as a *sharp* tensor, $\boldsymbol{A}^\sharp$, and one in which all indicies have been lowered as a *flat* tensor, $\boldsymbol{A}^\flat$. This is known as *musical isomorphism i.e.,* isomorphism between the tangent and cotangent bundles of a manifold, $\mathfrak{M}$.

$$\sharp : \mathcal{T}^*\mathfrak{M} \rightarrow \mathcal{T}\mathfrak{M} \tag{3.16}$$

and

$$\flat : \mathcal{T}\mathfrak{M} \rightarrow \mathcal{T}^*\mathfrak{M} \tag{3.17}$$

Induced metric

An induced metric is the metric tensor induced on a submanifold that has been embedded into a larger manifold with a metric. If $\boldsymbol{\xi}$ are the coordinates in the submanifold and $\boldsymbol{x}(\boldsymbol{\xi})$ are the functions which embedded the submanifold into a larger manifold then the induced metric is given by

$$g_{ab} = \frac{\partial x^\mu}{\partial \xi^a} \frac{\partial x^\nu}{\partial \xi^b} g_{\mu\nu} \tag{3.18}$$

where a, b are the coordinate indices in the submanifold, μ, ν are the coordinate indices in the larger manifold, $g_{\mu\nu}$ are the components of the metric tensor in the larger manifold and g_{ab} are the components of the induced metric in the submanifold.

3.6.3 Transformations

The transformation rules for tensors in going from a $\boldsymbol{\nu}$ coordinate system to a $\boldsymbol{\xi}$ coordinate system are as follows:

For a covariant vector (a rank (0,1) tensor)

$$\tilde{a}_i = \frac{\partial \nu^a}{\partial \xi^i} a_a \tag{3.19}$$

For a contravariant vector (a rank (1,0) tensor)

$$\tilde{a}^i = \frac{\partial \xi^i}{\partial \nu^a} a^a \tag{3.20}$$

For a covariant tensor (a rank (0,2) tensor)

$$\tilde{A}_{ij} = \frac{\partial \nu^a}{\partial \xi^i} \frac{\partial \nu^b}{\partial \xi^j} A_{ab} \tag{3.21}$$

For a contravariant tensor (a rank (2,0) tensor)

$$\tilde{A}^{ij} = \frac{\partial \xi^i}{\partial \nu^a} \frac{\partial \xi^j}{\partial \nu^b} A^{ab} \tag{3.22}$$

and for Mixed tensors (rank (1,1) tensors)

$$\tilde{A}^i_{\cdot j} = \frac{\partial \xi^i}{\partial \nu^a} \frac{\partial \nu^b}{\partial \xi^j} A^a_{\cdot b} \tag{3.23}$$

and

$$\tilde{A}^{\cdot j}_i = \frac{\partial \nu^a}{\partial \xi^i} \frac{\partial \xi^j}{\partial \nu^b} A^{\cdot b}_a \tag{3.24}$$

3.6.4 Derivatives

Scalars

We note that a scalar quantity $u(\boldsymbol{\xi})$ has derivatives

$$\frac{\partial u}{\partial \xi^i} = u_{,i} \tag{3.25}$$

Or more formally, the covariant derivative $(\cdot_{;\cdot})$ of a rank 0 tensor u is

$$u_{;i} = \frac{\partial u}{\partial \xi^i} = u_{,i} \tag{3.26}$$

In more formal mathematical notation consider f as a map between the manifolds $\mathfrak{M}$ and $\mathfrak{N}$

i.e.,

$$f : \mathfrak{M} \to \mathfrak{N} \tag{3.27}$$

then, at the point u, the derivative can be thought of as

$$D_u f : \mathcal{T}_u\mathfrak{M} \to \mathcal{T}_{f(u)}\mathfrak{N} \tag{3.28}$$

i.e., the derivative operator takes a vector $v \in \mathcal{T}_u\mathfrak{M}$ and maps it to another vector $w \in \mathcal{T}_{f(u)}\mathfrak{N}$.

Second derivatives. The second derivative or Hessian operator is defined as

$$D^2_{u,v} f : \mathcal{T}_v\mathcal{T}_{\mathrm{u}}\mathfrak{M} \to \mathcal{T}_{f'(v)}\mathcal{T}_{\mathrm{f(u)}}\mathfrak{N} \tag{3.29}$$

i.e., the hessian operator takes the first input as a vector $v \in \mathcal{T}_u\mathfrak{M}$ as the base direction you are moving in and a second input as a vector $w \in \mathcal{T}\mathcal{T}\mathfrak{M}v$ as the direction you are varying in.

The definition of the covariant Hessian is

$$D^2_{X,Y} f = X\left(Y\left(f\right)\right) - \nabla_X Y\left(f\right) \tag{3.30}$$

or in component notation

$$D^2 f = \left(\frac{\partial^2 f}{\partial x^i \partial x^j} - \Gamma^k_{ij}\frac{\partial f}{\partial x^k}\right) dx^i \otimes dx^j \tag{3.31}$$

Vectors

The derivatives of a vector $\boldsymbol{v}$ are given by

$$\begin{aligned}\frac{\partial \boldsymbol{v}}{\partial \xi^i} &= \frac{\partial}{\partial \xi^i}\left(v^k \boldsymbol{g}_k\right)\\ &= \frac{\partial v^k}{\partial \xi^i}\boldsymbol{g}_k + v^k\frac{\partial \boldsymbol{g}_k}{\partial \xi^i}\\ &= {v^k}_{,i}\boldsymbol{g}_k + v^k\boldsymbol{g}_{k,i}\end{aligned} \tag{3.32}$$

Now introducing the notation

$$\Gamma^i_{jk} = \boldsymbol{g}^i \cdot \frac{\partial \boldsymbol{g}_j}{\partial x^k} \tag{3.33}$$

where Γ^i_{jk} are the Christoffel symbols of the second kind.

Note that the Christoffel symbols of the first kind are given by

$$\Gamma_{ijk} = \boldsymbol{g}_i \cdot \frac{\partial \boldsymbol{g}_j}{\partial x^k} \tag{3.34}$$

Note that

$$\begin{aligned} \Gamma^i_{jk} &= \boldsymbol{g}^i \cdot \boldsymbol{g}_{j,k} \\ &= \boldsymbol{g}^i \cdot \Gamma^l_{jk} \boldsymbol{g}_l \\ &= \Gamma^i_{jl} g^i_{.l} \end{aligned} \tag{3.35}$$

The Christoffel symbols of the first kind are also given by

$$\Gamma_{ijk} = \frac{1}{2}\left(\frac{\partial g_{ij}}{\partial \xi^k} + \frac{\partial g_{ik}}{\partial \xi^j} - \frac{\partial g_{jk}}{\partial \xi^i}\right) \tag{3.36}$$

and that Christoffel symbols of the second kind are given by

$$\begin{aligned} \Gamma^i_{jk} &= g^{il}\Gamma_{ljk} \\ &= \frac{1}{2} g^{il}\left(\frac{\partial g_{lj}}{\partial \xi^k} + \frac{\partial g_{lk}}{\partial \xi^j} - \frac{\partial g_{jk}}{\partial \xi^l}\right) \end{aligned} \tag{3.37}$$

Note that Christoffel symbols are not tensors and the have the following transformation laws from $\boldsymbol{\nu}$ to $\boldsymbol{\xi}$ coordinates

$$\Gamma_{ijk} = \Gamma_{abc}\frac{\partial \nu^b}{\partial \xi^j}\frac{\partial \nu^c}{\partial \xi^k}\frac{\partial \nu^a}{\partial \xi^i} + g_{ab}\frac{\partial \nu^c}{\partial \xi^i}\frac{\partial^2 \nu^c}{\partial \xi^j \partial \xi^k} \tag{3.38}$$

$$\Gamma^i_{jk} = \Gamma^a_{bc}\frac{\partial \xi^i}{\partial \nu^a}\frac{\partial \nu^b}{\partial \xi^k}\frac{\partial \nu^c}{\partial \xi^j} + \frac{\partial \xi^i}{\partial \nu^a}\frac{\partial^2 \nu^a}{\partial \xi^j \partial \xi^k} \tag{3.39}$$

(3.40)

We can now write (BELOW SEEMS WRONG - CHECK)

$$\begin{aligned}\boldsymbol{v}_{,i} &= v^k{}_{,i}\boldsymbol{g}_k + \Gamma^k_{ij}v^j\boldsymbol{g}_j \\ &= v^k{}_{,i}\boldsymbol{g}_k + \Gamma^j_{ik}v^k\boldsymbol{g}_k \\ &= \left(v^k{}_{,i} + \Gamma^j_{ik}v^k\right)\boldsymbol{g}_k \\ &= v^k{}_{;i}\boldsymbol{g}_k\end{aligned} \tag{3.41}$$

where $v^k{}_{;i}$ is the covariant derivative of v^k .

The covariant derivative of a contravariant (rank (0,1)) tensor v^k is

$$v^k{}_{;i} = v^k{}_{,i} + \Gamma^k_{ij}v^j \tag{3.42}$$

and the covariant derivative of a covariant tensor (rank (1,0)) v_k is

$$v_{k;i} = v_{k,i} - \Gamma^j_{ki}v_j \tag{3.43}$$

Tensors

The covariant derivative of a contravariant (rank (0,2)) tensor W^{mn} is

$$W^{mn}{}_{;i} = W^{mn}{}_{,i} + \Gamma^m_{ji}W^{jn} + \Gamma^n_{ji}W^{mj} \tag{3.44}$$

and the covariant derivative of a covariant (rank (2,0)) tensor W_{mn} is

$$W_{mn;i} = W_{mn,i} - \Gamma^j_{mi}W_{jn} - \Gamma^j_{ni}W_{mj} \tag{3.45}$$

and the covariant derivative of a mixed (rank (1,1)) tensor $W^m_{.n}$ is

$$W^m_{.n\,;i} = W^m_{.n\,,i} + \Gamma^m_{ji}W^j_{.n} - \Gamma^j_{ni}W^m_{.j} \tag{3.46}$$

3.6.5 Common Operators

For tensor equations to hold in any coordinate system the equations must involve tensor quantities *i.e.,* covariant derivatives rather than partial derivatives.

Gradient

As the covariant derivative of a scalar is just the partial derivative the gradient of a scalar function ϕ using covariant derivatives is

$$\operatorname{grad}\phi = \nabla\phi = \phi_{;i}\boldsymbol{g}^i = \phi_{,i}\boldsymbol{g}^i \tag{3.47}$$

and

$$\nabla\phi = \phi_{,i}\boldsymbol{g}^i = \phi_{,i}g^{ij}\boldsymbol{g}_j \tag{3.48}$$

Divergence

The divergence of a vector using covariant derivatives is

$$\operatorname{div}\boldsymbol{\phi} = \nabla\cdot\boldsymbol{\phi} = \phi^i{}_{;i} = \frac{1}{\sqrt{|g|}}\left(\sqrt{|g|}\phi^i\right)_{,i} \tag{3.49}$$

where g is the determinant of the covariant metric tensor g_{ij}.

Curl

The curl of a vector using covariant derivatives is

$$\operatorname{curl}\boldsymbol{\phi} = \nabla\times\boldsymbol{\phi} = \frac{1}{\sqrt{g}}\left(\phi_{j;i} - \phi_{i;j}\right)\boldsymbol{g}_k \tag{3.50}$$

where g is the determinant of the covariant metric tensor g_{ij}.

Laplacian

The Laplacian of a scalar using covariant derivatives is

$$\nabla^2\phi = \operatorname{div}(\operatorname{grad}\phi) = \nabla\cdot\nabla\phi = \phi|_i^i = \frac{1}{\sqrt{g}}\left(\sqrt{g}g^{ij}\phi_{,j}\right)_{,i} \tag{3.51}$$

where g is the determinant of the covariant metric tensor g_{ij}.

The Laplacian of a vector using covariant derivatives is

$$\nabla^2 \boldsymbol{\phi} = \text{grad}\,(\text{div}\,\boldsymbol{\phi}) - \text{curl}\,(\text{curl}\,\boldsymbol{\phi}) == \boldsymbol{\phi}|_i^i \tag{3.52}$$

The Laplacian of a contravariant (rank (0,1)) tensor ϕ^k is

$$\nabla^2 \boldsymbol{\phi} = \left(\nabla^2 \phi_k - 2 g^{ij} \Gamma^K_{jH} \frac{\partial \phi^h}{\partial x^i} + \phi^h \frac{\partial g^{ij} \Gamma^K_{ij}}{\partial x^h} \right) \boldsymbol{e}^k \tag{3.53}$$

and the covariant derivative of a covariant tensor (rank (1,0)) ϕ_k is

$$\nabla^2 \boldsymbol{\phi} = \left(\nabla^2 \phi_k - 2 g^{ij} \Gamma^h_{jk} \frac{\partial \phi_h}{\partial x^i} + \phi_h g^{ij} \frac{\partial \Gamma^h_{ij}}{\partial x^i} \right) \boldsymbol{e}_k \tag{3.54}$$

3.6.6 Coordinate Systems

Rectangular Cartesian

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \boldsymbol{i}_1 \\ \boldsymbol{i}_2 \\ \boldsymbol{i}_3 \end{bmatrix} \tag{3.55}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.56}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.57}$$

The Christoffel symbols of the second kind are all zero.

Cylindrical Polar

The global coordinates (x, y, z) with respect to the cylindrical polar coordinates (r, θ, z) are defined by

$$\begin{aligned} x &= r\cos\theta & r &\geq 0 \\ y &= r\sin\theta & 0 &\leq \theta \leq 2\pi \\ z &= z & -\infty &< z < \infty \end{aligned} \tag{3.58}$$

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \cos\theta\boldsymbol{i}_1 + \sin\theta\boldsymbol{i}_2 \\ -r\sin\theta\boldsymbol{i}_1 + r\cos\theta\boldsymbol{i}_2 \\ \boldsymbol{i}_3 \end{bmatrix} \tag{3.59}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.60}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.61}$$

The Christoffell symbols of the second kind are

$$\Gamma^r_{\theta\theta} = -r \tag{3.62}$$

$$\Gamma^\theta_{r\theta} = \Gamma^\theta_{\theta r} = \frac{1}{r} \tag{3.63}$$

with all other Christoffell symbols zero.

Spherical Polar

The global coordinates (x, y, z) with respect to the cylindrical polar coordinates (r, θ, ϕ) are defined by

$$\begin{aligned} x &= r\cos\theta\sin\phi & r &\geq 0 \\ y &= r\sin\theta\sin\phi & 0 &\leq \theta \leq 2\pi \\ z &= r\cos\phi & 0 &\leq \phi \leq \pi \end{aligned} \tag{3.64}$$

The base vectors with respect to the spherical polar coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \cos\theta\sin\phi\boldsymbol{i}_1 + \sin\theta\sin\phi\boldsymbol{i}_2 + \cos\phi\boldsymbol{i}_3 \\ -r\sin\theta\sin\phi\boldsymbol{i}_1 + r\cos\theta\sin\phi\boldsymbol{i}_2 \\ r\cos\theta\cos\phi\boldsymbol{i}_1 + r\sin\theta\cos\phi\boldsymbol{i}_2 - r\sin\phi\boldsymbol{i}_3 \end{bmatrix} \tag{3.65}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^2\sin^2\phi & 0 \\ 0 & 0 & r^2 \end{bmatrix} \tag{3.66}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2\sin^2\phi} & 0 \\ 0 & 0 & \frac{1}{r^2} \end{bmatrix} \tag{3.67}$$

The Christoffell symbols of the second kind are

$$\Gamma^{r}_{\theta\theta} = -r\sin^2\phi \tag{3.68}$$

$$\Gamma^{r}_{\phi\phi} = -r \tag{3.69}$$

$$\Gamma^{\phi}_{\theta\theta} = -\sin\phi\cos\phi \tag{3.70}$$

$$\Gamma^{\theta}_{r\theta} = \Gamma^{\theta}_{\theta r} = \frac{1}{r} \tag{3.71}$$

$$\Gamma^{\phi}_{r\phi} = \Gamma^{\phi}_{\phi r} = \frac{1}{r} \tag{3.72}$$

$$\Gamma^{\theta}_{\theta\phi} = \Gamma^{\theta}_{\phi\theta} = \cot\phi \tag{3.73}$$

with all other Christoffel symbols zero.

Prolate Spheroidal

The global coordinates (x, y, z) with respect to the prolate spheroidal coordinates (λ, μ, θ) are defined by

$$\begin{aligned} x &= a\sinh\lambda\sin\mu\cos\theta & \lambda &\geq 0 \\ y &= a\sinh\lambda\sin\mu\sin\theta & 0 &\leq \mu \leq \pi \\ z &= a\cosh\lambda\cos\mu & 0 &\leq \theta \leq 2\pi \end{aligned} \tag{3.74}$$

where $a \geq 0$ is the focus.

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} a\cosh\lambda\sin\mu\cos\theta\boldsymbol{i}_1 + a\cosh\lambda\sin\mu\sin\theta\boldsymbol{i}_2 + a\sinh\lambda\cos\mu\boldsymbol{i}_3 \\ a\sinh\lambda\cos\mu\cos\theta\boldsymbol{i}_1 + a\sinh\lambda\cos\mu\sin\theta\boldsymbol{i}_2 - a\cosh\lambda\sin\mu\boldsymbol{i}_3 \\ -a\sinh\lambda\sin\mu\sin\theta\boldsymbol{i}_1 + a\sinh\lambda\sin\mu\cos\theta\boldsymbol{i}_2 \end{bmatrix} \tag{3.75}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 & 0 \\ 0 & a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 \\ 0 & 0 & a^2\sinh^2\lambda\sin^2\mu \end{bmatrix} \tag{3.76}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 & 0 \\ 0 & \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 \\ 0 & 0 & \frac{1}{a^2\sinh^2\lambda\sin^2\mu} \end{bmatrix} \tag{3.77}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\lambda}_{\lambda\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda+\sin^2\mu} \tag{3.78}$$

$$\Gamma^{\lambda}_{\mu\mu} = \frac{-\sinh\lambda\cosh\lambda}{\sinh^2\lambda+\sin^2\mu} \tag{3.79}$$

$$\Gamma^{\lambda}_{\theta\theta} = \frac{-\sinh\lambda\cosh\lambda\sin^2\mu}{\sinh^2\lambda+\sin^2\mu} \tag{3.80}$$

$$\Gamma^{\lambda}_{\lambda\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda+\sin^2\mu} \tag{3.81}$$

$$\Gamma^{\mu}_{\mu\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda+\sin^2\mu} \tag{3.82}$$

$$\Gamma^{\mu}_{\lambda\lambda} = \frac{-\sin\mu\cos\mu}{\sinh^2\lambda+\sin^2\mu} \tag{3.83}$$

$$\Gamma^{\mu}_{\theta\theta} = \frac{-\sinh^2\lambda\sin\mu\cos\mu}{\sinh^2\lambda+\sin^2\mu} \tag{3.84}$$

$$\Gamma^{\mu}_{\mu\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda+\sin^2\mu} \tag{3.85}$$

$$\Gamma^{\theta}_{\theta\lambda} = \frac{\cosh\lambda}{\sinh\lambda} \tag{3.86}$$

$$\Gamma^{\theta}_{\theta\mu} = \frac{\cos\mu}{\sin\mu} \tag{3.87}$$

$$\tag{3.88}$$

with all other Christoffel symbols zero.

Oblate Spheroidal

The global coordinates (x, y, z) with respect to the oblate spheroidal coordinates (λ, μ, θ) are defined by

$$\begin{aligned} x &= a\cosh\lambda\cos\mu\cos\theta && \lambda \geq 0 \\ y &= a\cosh\lambda\cos\mu\sin\theta && \frac{-\pi}{2} \leq \mu \leq \frac{\pi}{2} \\ z &= a\sinh\lambda\sin\mu && 0 \leq \theta \leq 2\pi \end{aligned} \tag{3.89}$$

where $a \geq 0$ is the focus.

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} a\sinh\lambda\cos\mu\cos\theta\boldsymbol{i}_1 + a\sinh\lambda\cos\mu\sin\theta\boldsymbol{i}_2 + a\cosh\lambda\sin\mu\boldsymbol{i}_3 \\ -a\cosh\lambda\sin\mu\cos\theta\boldsymbol{i}_1 - a\cosh\lambda\sin\mu\sin\theta\boldsymbol{i}_2 + a\sinh\lambda\cos\mu\boldsymbol{i}_3 \\ -a\cosh\lambda\cos\mu\sin\theta\boldsymbol{i}_1 + a\cosh\lambda\cos\mu\cos\theta\boldsymbol{i}_2 \end{bmatrix} \tag{3.90}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 & 0 \\ 0 & a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 \\ 0 & 0 & a^2\cosh^2\lambda\cos^2\mu \end{bmatrix} \tag{3.91}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 & 0 \\ 0 & \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 \\ 0 & 0 & \frac{1}{a^2\cosh^2\lambda\cos^2\mu} \end{bmatrix} \tag{3.92}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\lambda}_{\lambda\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{3.93}$$

$$\Gamma^{\lambda}_{\mu\mu} = \frac{-\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{3.94}$$

$$\Gamma^{\lambda}_{\theta\theta} = \frac{-\sinh\lambda\cosh\lambda\cos^2\mu}{\sinh^2\lambda + \sin^2\mu} \tag{3.95}$$

$$\Gamma^{\lambda}_{\lambda\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{3.96}$$

$$\Gamma^{\mu}_{\mu\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{3.97}$$

$$\Gamma^{\mu}_{\lambda\lambda} = \frac{-\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{3.98}$$

$$\Gamma^{\mu}_{\theta\theta} = \frac{\cosh^2\lambda\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{3.99}$$

$$\Gamma^{\mu}_{\mu\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{3.100}$$

$$\Gamma^{\theta}_{\theta\lambda} = \frac{\sinh\lambda}{\cosh\lambda} \tag{3.101}$$

$$\Gamma^{\theta}_{\theta\mu} = \frac{-\sin\mu}{\cos\mu} \tag{3.102}$$

$$\tag{3.103}$$

with all other Christoffel symbols zero.

Cylindrical parabolic

The global coordinates (x, y, z) with respect to the cylindrical parabolic coordinates (ξ, η, z) are defined by

$$\begin{aligned} x &= \xi\eta \qquad && -\infty < \xi < \infty \\ y &= \frac{1}{2}\left(\xi^2 - \eta^2\right) \qquad && \eta \geq 0 \\ z &= z \qquad && -\infty < z < \infty \end{aligned} \tag{3.104}$$

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \eta\boldsymbol{i}_1 + \xi\boldsymbol{i}_2 \\ \xi\boldsymbol{i}_1 - \eta\boldsymbol{i}_2 \\ \boldsymbol{i}_3 \end{bmatrix} \tag{3.105}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} \xi^2+\eta^2 & 0 & 0 \\ 0 & \xi^2+\eta^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.106}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{\xi^2+\eta^2} & 0 & 0 \\ 0 & \frac{1}{\xi^2+\eta^2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.107}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\xi}_{\xi\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{3.108}$$

$$\Gamma^{\eta}_{\eta\eta} = \frac{\eta}{\xi^2+\eta^2} \tag{3.109}$$

$$\Gamma^{\eta}_{\xi\xi} = \frac{-\eta}{\xi^2+\eta^2} \tag{3.110}$$

$$\Gamma^{\xi}_{\eta\eta} = \frac{-\xi}{\xi^2+\eta^2} \tag{3.111}$$

$$\Gamma^{\xi}_{\xi\eta} = \Gamma^{\xi}_{\eta\xi} = \frac{\eta}{\xi^2+\eta^2} \tag{3.112}$$

$$\Gamma^{\eta}_{\xi\eta} = \Gamma^{\eta}_{\eta\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{3.113}$$

$$\tag{3.114}$$

with all other Christoffel symbols zero.

Parabolic polar

The global coordinates (x, y, z) with respect to the cylindrical parabolic coordinates (ξ, η, θ) are defined by

$$\begin{aligned} x &= \xi\eta\cos\theta & \xi &\geq 0 \\ y &= \xi\eta\sin\theta & \eta &\geq 0 \\ z &= \frac{1}{2}\left(\xi^2 - \eta^2\right) & 0 &\leq \theta < 2\pi \end{aligned} \tag{3.115}$$

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \eta\cos\theta\boldsymbol{i}_1 + \eta\sin\theta\boldsymbol{i}_3 + \xi\boldsymbol{i}_3 \\ \xi\cos\theta\boldsymbol{i}_1 + \xi\sin\theta\boldsymbol{i}_3 - \eta\boldsymbol{i}_3 \\ -\xi\eta\sin\theta\boldsymbol{i}_1 + \xi\eta\cos\theta\boldsymbol{i}_2 \end{bmatrix} \tag{3.116}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} \xi^2 + \eta^2 & 0 & 0 \\ 0 & \xi^2 + \eta^2 & 0 \\ 0 & 0 & \xi\eta \end{bmatrix} \tag{3.117}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{\xi^2+\eta^2} & 0 & 0 \\ 0 & \frac{1}{\xi^2+\eta^2} & 0 \\ 0 & 0 & \frac{1}{\xi\eta} \end{bmatrix} \tag{3.118}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\xi}_{\xi\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{3.119}$$

$$\Gamma^{\eta}_{\eta\eta} = \frac{\eta}{\xi^2+\eta^2} \tag{3.120}$$

$$\Gamma^{\xi}_{\eta\eta} = \frac{-\xi}{\xi^2+\eta^2} \tag{3.121}$$

$$\Gamma^{\eta}_{\xi\xi} = \frac{-\eta}{\xi^2+\eta^2} \tag{3.122}$$

$$\Gamma^{\eta}_{\theta\theta} = \frac{-\xi^2\eta}{\xi^2+\eta^2} \tag{3.123}$$

$$\Gamma^{\xi}_{\theta\theta} = \frac{-\xi\eta^2}{\xi^2+\eta^2} \tag{3.124}$$

$$\Gamma^{\xi}_{\xi\eta} = \Gamma^{\xi}_{\eta\xi} = \frac{\eta}{\xi^2+\eta^2} \tag{3.125}$$

$$\Gamma^{\eta}_{\xi\eta} = \Gamma^{\eta}_{\eta\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{3.126}$$

$$\Gamma^{\theta}_{\xi\theta} = \Gamma^{\theta}_{\theta\xi} = \frac{1}{\xi} \tag{3.127}$$

$$\Gamma^{\theta}_{\eta\theta} = \Gamma^{\theta}_{\theta\eta} = \frac{1}{\eta} \tag{3.128}$$

$$ \tag{3.129}$$

with all other Christoffel symbols zero.

3.7 Curves

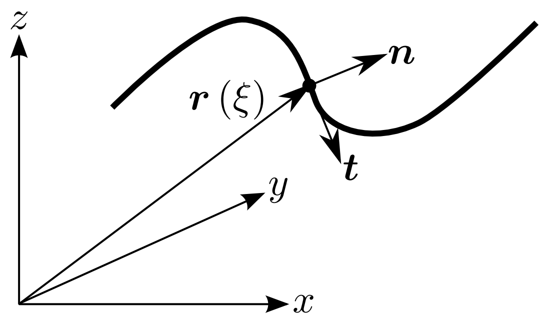

FIGURE 3.1: A parametised curve in space. The position, $\boldsymbol{r}(\xi)$ is given in terms of the curve parameter ξ. $\boldsymbol{t}$ is the tangent vector to the curve and $\boldsymbol{n}$ is the normal vector to the curve.

Chapter 4

Theory

4.1 Tensor Analysis

4.1.1 Base vectors

Now, if we have a vector, $\boldsymbol{v}$ we can write

$$\boldsymbol{v} = v^i \boldsymbol{g}_i \tag{4.1}$$

where v^i are the components of the contravariant vector, and $\boldsymbol{g}_i$ are the covariant base vectors.

Similarly, the vector $\boldsymbol{v}$ can also be written as

$$\boldsymbol{v} = v_i \boldsymbol{g}^i \tag{4.2}$$

where v_i are the components of the covariant vector, and $\boldsymbol{g}^i$ are the contravariant base vectors.

We now note that

$$\boldsymbol{v} = v^i \boldsymbol{g}_i = v^i \sqrt{g_{ii}} \hat{\boldsymbol{g}}_i \tag{4.3}$$

where $v^i \sqrt{g_{ii}}$ are the physical components of the vector and $\hat{\boldsymbol{g}}_i$ are the unit vectors given by

$$\hat{\boldsymbol{g}}_i = \frac{\boldsymbol{g}_i}{\sqrt{g_{ii}}} \tag{4.4}$$

4.1.2 Metric Tensors

Metric tensors are the inner product of base vectors. If $\boldsymbol{g}_i$ are the covariant base vectors then the covariant metric tensor is given by

$$g_{ij} = \boldsymbol{g}_i \cdot \boldsymbol{g}_j \tag{4.5}$$

Similarily if $\boldsymbol{g}^i$ are the contravariant base vectors then the contravariant metric tensor is given by

$$g^{ij} = \boldsymbol{g}^i \cdot \boldsymbol{g}^j \tag{4.6}$$

We can also form a mixed metric tensor from the dot product of a contravariant and a covariant base vector *i.e.,*

$$g^i_{.j} = \boldsymbol{g}^i \cdot \boldsymbol{g}_j \tag{4.7}$$

and

$$g_i^{.j} = \boldsymbol{g}_i \cdot \boldsymbol{g}^j \tag{4.8}$$

Note that for mixed tensors the "." indicates the order of the index *i.e.,* $g^i_{.j}$ indicates that the first index is contravariant and the second index is covariant whereas $g_i^{.j}$ indicates that the first index is covariant and the second index is contravariant.

If the base vectors are all mutually orthogonal and constant then $\boldsymbol{g}_i = \boldsymbol{g}^i$ and $g_{ij} = g^{ij}$.

The metric tensors generalise (Euclidean) distance *i.e.,*

$$ds^2 = g_{ij} dx^i dx^j \tag{4.9}$$

Note that multiplying by the covariant metric tensor lowers indices *i.e.,*

$$\begin{aligned} \boldsymbol{A}_i &= g_{ij} \boldsymbol{A}^j \\ A_{ij} &= g_{ik} g_{jl} A^{kl} = g_{jk} A_i^{.k} = g_{ik} A^k_{.j} \end{aligned} \tag{4.10}$$

and that multiplying by the contravariant metric tensor raises indices *i.e.,*

$$\begin{aligned} \boldsymbol{A}^i &= g^{ij} \boldsymbol{A}_j \\ A^{ij} &= g^{ik} g^{jl} A_{kl} = g^{ik} A_k^{.j} = g^{jk} A^i_{.k} \end{aligned} \tag{4.11}$$

and for the mixed tensors

$$\begin{aligned} A_i^{\cdot j} &= g^{jk} A_{ik} = g_{ik} A^{kj} \\ A_{\cdot j}^{i} &= g^{ik} A_{kj} = g_{jk} A^{ik} \end{aligned} \tag{4.12}$$

4.1.3 Transformations

The transformation rules for tensors in going from a $\boldsymbol{\nu}$ coordinate system to a $\boldsymbol{\xi}$ coordinate system are as follows:

For a covariant vector (a rank (0,1) tensor)

$$\tilde{a}_i = \frac{\partial \nu^a}{\partial \xi^i} a_a \tag{4.13}$$

For a contravariant vector (a rank (1,0) tensor)

$$\tilde{a}^i = \frac{\partial \xi^i}{\partial \nu^a} a^a \tag{4.14}$$

For a covariant tensor (a rank (0,2) tensor)

$$\tilde{A}_{ij} = \frac{\partial \nu^a}{\partial \xi^i} \frac{\partial \nu^b}{\partial \xi^j} A_{ab} \tag{4.15}$$

For a contravariant tensor (a rank (2,0) tensor)

$$\tilde{A}^{ij} = \frac{\partial \xi^i}{\partial \nu^a} \frac{\partial \xi^j}{\partial \nu^b} A^{ab} \tag{4.16}$$

and for Mixed tensors (rank (1,1) tensors)

$$\tilde{A}^{i}_{\cdot j} = \frac{\partial \xi^i}{\partial \nu^a} \frac{\partial \nu^b}{\partial \xi^j} A^{a}_{\cdot b} \tag{4.17}$$

and

$$\tilde{A}_i^{\cdot j} = \frac{\partial \nu^a}{\partial \xi^i} \frac{\partial \xi^j}{\partial \nu^b} A_a^{\cdot b} \tag{4.18}$$

4.1.4 Derivatives

Scalars

We note that a scalar quantity $u(\boldsymbol{\xi})$ has derivatives

$$\frac{\partial u}{\partial \xi^i} = u_{,i} \tag{4.19}$$

Or more formally, the covariant derivative $(\cdot_{;\cdot})$ of a rank 0 tensor u is

$$u_{;i} = \frac{\partial u}{\partial \xi^i} = u_{,i} \tag{4.20}$$

Vectors

The derivatives of a vector $\boldsymbol{v}$ are given by

$$\begin{aligned} \frac{\partial \boldsymbol{v}}{\partial \xi^i} &= \frac{\partial}{\partial \xi^i}\left(v^k \boldsymbol{g}_k\right) \\ &= \frac{\partial v^k}{\partial \xi^i}\boldsymbol{g}_k + v^k \frac{\partial \boldsymbol{g}_k}{\partial \xi^i} \\ &= v^k{}_{,i}\boldsymbol{g}_k + v^k \boldsymbol{g}_{k,i} \end{aligned} \tag{4.21}$$

Now introducing the notation

$$\Gamma^i_{jk} = \boldsymbol{g}^i \cdot \frac{\partial \boldsymbol{g}_j}{\partial x^k} \tag{4.22}$$

where Γ^i_{jk} are the Christoffel symbols of the second kind.

Note that the Christoffel symbols of the first kind are given by

$$\Gamma_{ijk} = \boldsymbol{g}_i \cdot \frac{\partial \boldsymbol{g}_j}{\partial x^k} \tag{4.23}$$

Note that

$$\begin{aligned} \Gamma^i_{jk} &= \boldsymbol{g}^i \cdot \boldsymbol{g}_{j,k} \\ &= \boldsymbol{g}^i \cdot \Gamma^l_{jk}\boldsymbol{g}_l \\ &= \Gamma^i_{jl} g^i_{.l} \end{aligned} \tag{4.24}$$

The Christoffel symbols of the first kind are also given by

$$\Gamma_{ijk} = \frac{1}{2}\left(\frac{\partial g_{ij}}{\partial \xi^k} + \frac{\partial g_{ik}}{\partial \xi^j} - \frac{\partial g_{jk}}{\partial \xi^i}\right) \tag{4.25}$$

and that Christoffel symbols of the second kind are given by

$$\begin{aligned}\Gamma^i_{jk} &= g^{il}\Gamma_{ljk} \\ &= \frac{1}{2}g^{il}\left(\frac{\partial g_{lj}}{\partial \xi^k} + \frac{\partial g_{lk}}{\partial \xi^j} - \frac{\partial g_{jk}}{\partial \xi^l}\right)\end{aligned} \tag{4.26}$$

Note that Christoffel symbols are not tensors and the have the following transformation laws from $\boldsymbol{\nu}$ to $\boldsymbol{\xi}$ coordinates

$$\Gamma_{ijk} = \Gamma_{abc}\frac{\partial \nu^b}{\partial \xi^j}\frac{\partial \nu^c}{\partial \xi^k}\frac{\partial \nu^a}{\partial \xi^i} + g_{ab}\frac{\partial \nu^c}{\partial \xi^i}\frac{\partial^2 \nu^c}{\partial \xi^j \partial \xi^k} \tag{4.27}$$

$$\Gamma^i_{jk} = \Gamma^a_{bc}\frac{\partial \xi^i}{\partial \nu^a}\frac{\partial \nu^b}{\partial \xi^k}\frac{\partial \nu^c}{\partial \xi^j} + \frac{\partial \xi^i}{\partial \nu^a}\frac{\partial^2 \nu^a}{\partial \xi^j \partial \xi^k} \tag{4.28}$$

$$\tag{4.29}$$

We can now write (BELOW SEEMS WRONG - CHECK)

$$\begin{aligned}\boldsymbol{v}_{,i} &= v^k{}_{,i}\boldsymbol{g}_k + \Gamma^k_{ij}v^j\boldsymbol{g}_j \\ &= v^k{}_{,i}\boldsymbol{g}_k + \Gamma^j_{ik}v^k\boldsymbol{g}_k \\ &= \left(v^k{}_{,i} + \Gamma^j_{ik}v^k\right)\boldsymbol{g}_k \\ &= v^k{}_{;i}\boldsymbol{g}_k\end{aligned} \tag{4.30}$$

where $v^k{}_{;i}$ is the covariant derivative of v^k .

The covariant derivative of a contravariant (rank (0,1)) tensor v^k is

$$v^k{}_{;i} = v^k{}_{,i} + \Gamma^k_{ij}v^j \tag{4.31}$$

and the covariant derivative of a covariant tensor (rank (1,0)) v_k is

$$v_{k;i} = v_{k,i} - \Gamma^j_{ki}v_j \tag{4.32}$$

Tensors

The covariant derivative of a contravariant (rank (0,2)) tensor W^{mn} is

$$W^{mn}{}_{;i} = W^{mn}{}_{,i} + \Gamma^m_{ji} W^{jn} + \Gamma^n_{ji} W^{mj} \tag{4.33}$$

and the covariant derivative of a covariant (rank (2,0)) tensor W_{mn} is

$$W_{mn;i} = W_{mn,i} - \Gamma^j_{mi} W_{jn} - \Gamma^j_{ni} W_{mj} \tag{4.34}$$

and the covariant derivative of a mixed (rank (1,1)) tensor $W^m_{.n}$ is

$$W^m_{.n\,;i} = W^m_{.n\,,i} + \Gamma^m_{ji} W^j_{.n} - \Gamma^j_{ni} W^m_{.j} \tag{4.35}$$

4.1.5 Common Operators

For tensor equations to hold in any coordinate system the equations must involve tensor quantities *i.e.,* covariant derivatives rather than partial derivatives.

Gradient

As the covariant derivative of a scalar is just the partial derivative the gradient of a scalar function ϕ using covariant derivatives is

$$\text{grad}\,\phi = \nabla\phi = \phi_{;i}\boldsymbol{g}^i = \phi_{,i}\boldsymbol{g}^i \tag{4.36}$$

and

$$\nabla\phi = \phi_{,i}\boldsymbol{g}^i = \phi_{,i} g^{ij}\boldsymbol{g}_j \tag{4.37}$$

Divergence

The divergence of a vector using covariant derivatives is

$$\text{div}\,\boldsymbol{\phi} = \nabla\cdot\boldsymbol{\phi} = \phi^i{}_{;i} = \frac{1}{\sqrt{|g|}}\left(\sqrt{|g|}\phi^i\right)_{,i} \tag{4.38}$$

where g is the determinant of the covariant metric tensor g_{ij}.

Curl

The curl of a vector using covariant derivatives is

$$\text{curl}\,\boldsymbol{\phi} = \nabla \times \boldsymbol{\phi} = \frac{1}{\sqrt{g}}\left(\phi_{j;i} - \phi_{i;j}\right)\boldsymbol{g}_k \tag{4.39}$$

where g is the determinant of the covariant metric tensor g_{ij}.

Laplacian

The Laplacian of a scalar using covariant derivatives is

$$\nabla^2_{\phi} = \text{div}\,(\text{grad}\ \phi) = \nabla \cdot \nabla\phi = \phi|^i_i = \frac{1}{\sqrt{g}}\left(\sqrt{g}g^{ij}\phi_{,j}\right)_{,i} \tag{4.40}$$

where g is the determinant of the covariant metric tensor g_{ij}.

The Laplacian of a vector using covariant derivatives is

$$\nabla^2_{\boldsymbol{\phi}} = \text{grad}\ \ (\text{div}\ \boldsymbol{\phi}) - \text{curl}\ \ (\text{curl}\ \boldsymbol{\phi}) == \boldsymbol{\phi}|^i_i \tag{4.41}$$

The Laplacian of a contravariant (rank (0,1)) tensor ϕ^k is

$$\nabla^2_{\boldsymbol{\phi}} = \left(\nabla^2_{\phi_k} - 2g^{ij}\Gamma^K_{jH}\frac{\partial\phi^h}{\partial x^i} + \phi^h\frac{\partial g^{ij}\Gamma^K_{ij}}{\partial x^h}\right)\boldsymbol{e}^k \tag{4.42}$$

and the covariant derivative of a covariant tensor (rank (1,0)) ϕ_k is

$$\nabla^2_{\boldsymbol{\phi}} = \left(\nabla^2_{\phi_k} - 2g^{ij}\Gamma^h_{jk}\frac{\partial\phi_h}{\partial x^i} + \phi_h g^{ij}\frac{\partial\Gamma^h_{ij}}{\partial x^i}\right)\boldsymbol{e}_k \tag{4.43}$$

4.1.6 Coordinate Systems

Rectangular Cartesian

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \boldsymbol{i}_1 \\ \boldsymbol{i}_2 \\ \boldsymbol{i}_3 \end{bmatrix} \tag{4.44}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.45}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.46}$$

The Christoffel symbols of the second kind are all zero.

Cylindrical Polar

The global coordinates (x, y, z) with respect to the cylindrical polar coordinates (r, θ, z) are defined by

$$\begin{aligned} x &= r\cos\theta \qquad && r \geq 0 \\ y &= r\sin\theta \qquad && 0 \leq \theta \leq 2\pi \\ z &= z \qquad && -\infty < z < \infty \end{aligned} \tag{4.47}$$

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \cos\theta\boldsymbol{i}_1 + \sin\theta\boldsymbol{i}_2 \\ -r\sin\theta\boldsymbol{i}_1 + r\cos\theta\boldsymbol{i}_2 \\ \boldsymbol{i}_3 \end{bmatrix} \tag{4.48}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.49}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.50}$$

The Christoffell symbols of the second kind are

$$\Gamma^r_{\theta\theta} = -r \tag{4.51}$$

$$\Gamma^\theta_{r\theta} = \Gamma^\theta_{\theta r} = \frac{1}{r} \tag{4.52}$$

with all other Christoffell symbols zero.

Spherical Polar

The global coordinates (x, y, z) with respect to the cylindrical polar coordinates (r, θ, ϕ) are defined by

$$\begin{aligned} x &= r\cos\theta\sin\phi && r \geq 0 \\ y &= r\sin\theta\sin\phi && 0 \leq \theta \leq 2\pi \\ z &= r\cos\phi && 0 \leq \phi \leq \pi \end{aligned} \tag{4.53}$$

The base vectors with respect to the spherical polar coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \cos\theta\sin\phi\boldsymbol{i}_1 + \sin\theta\sin\phi\boldsymbol{i}_2 + \cos\phi\boldsymbol{i}_3 \\ -r\sin\theta\sin\phi\boldsymbol{i}_1 + r\cos\theta\sin\phi\boldsymbol{i}_2 \\ r\cos\theta\cos\phi\boldsymbol{i}_1 + r\sin\theta\cos\phi\boldsymbol{i}_2 - r\sin\phi\boldsymbol{i}_3 \end{bmatrix} \tag{4.54}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^2 \sin^2 \phi & 0 \\ 0 & 0 & r^2 \end{bmatrix} \tag{4.55}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{r^2 \sin^2 \phi} & 0 \\ 0 & 0 & \frac{1}{r^2} \end{bmatrix} \tag{4.56}$$

The Christoffell symbols of the second kind are

$$\Gamma^{r}_{\theta\theta} = -r \sin^2 \phi \tag{4.57}$$

$$\Gamma^{r}_{\phi\phi} = -r \tag{4.58}$$

$$\Gamma^{\phi}_{\theta\theta} = -\sin \phi \cos \phi \tag{4.59}$$

$$\Gamma^{\theta}_{r\theta} = \Gamma^{\theta}_{\theta r} = \frac{1}{r} \tag{4.60}$$

$$\Gamma^{\phi}_{r\phi} = \Gamma^{\phi}_{\phi r} = \frac{1}{r} \tag{4.61}$$

$$\Gamma^{\theta}_{\theta\phi} = \Gamma^{\theta}_{\phi\theta} = \cot \phi \tag{4.62}$$

with all other Christofell symbols zero.

Prolate Spheroidal

The global coordinates (x, y, z) with respect to the prolate spheroidal coordinates (λ, μ, θ) are defined by

$$\begin{aligned} x &= a \sinh \lambda \sin \mu \cos \theta & \lambda &\geq 0 \\ y &= a \sinh \lambda \sin \mu \sin \theta & 0 &\leq \mu \leq \pi \\ z &= a \cosh \lambda \cos \mu & 0 &\leq \theta \leq 2\pi \end{aligned} \tag{4.63}$$

where $a \geq 0$ is the focus.

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} a\cosh\lambda\sin\mu\cos\theta\boldsymbol{i}_1 + a\cosh\lambda\sin\mu\sin\theta\boldsymbol{i}_2 + a\sinh\lambda\cos\mu\boldsymbol{i}_3 \\ a\sinh\lambda\cos\mu\cos\theta\boldsymbol{i}_1 + a\sinh\lambda\cos\mu\sin\theta\boldsymbol{i}_2 - a\cosh\lambda\sin\mu\boldsymbol{i}_3 \\ -a\sinh\lambda\sin\mu\sin\theta\boldsymbol{i}_1 + a\sinh\lambda\sin\mu\cos\theta\boldsymbol{i}_2 \end{bmatrix} \tag{4.64}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 & 0 \\ 0 & a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 \\ 0 & 0 & a^2\sinh^2\lambda\sin^2\mu \end{bmatrix} \tag{4.65}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 & 0 \\ 0 & \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 \\ 0 & 0 & \frac{1}{a^2\sinh^2\lambda\sin^2\mu} \end{bmatrix} \tag{4.66}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\lambda}_{\lambda\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{4.67}$$

$$\Gamma^{\lambda}_{\mu\mu} = \frac{-\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{4.68}$$

$$\Gamma^{\lambda}_{\theta\theta} = \frac{-\sinh\lambda\cosh\lambda\sin^2\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.69}$$

$$\Gamma^{\lambda}_{\lambda\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.70}$$

$$\Gamma^{\mu}_{\mu\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.71}$$

$$\Gamma^{\mu}_{\lambda\lambda} = \frac{-\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.72}$$

$$\Gamma^{\mu}_{\theta\theta} = \frac{-\sinh^2\lambda\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.73}$$

$$\Gamma^{\mu}_{\mu\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{4.74}$$

$$\Gamma^{\theta}_{\theta\lambda} = \frac{\cosh\lambda}{\sinh\lambda} \tag{4.75}$$

$$\Gamma^{\theta}_{\theta\mu} = \frac{\cos\mu}{\sin\mu} \tag{4.76}$$

$$\tag{4.77}$$

with all other Christofell symbols zero.

Oblate Spheroidal

The global coordinates (x, y, z) with respect to the oblate spheroidal coordinates (λ, μ, θ) are defined by

$$\begin{aligned} x &= a\cosh\lambda\cos\mu\cos\theta & \lambda &\geq 0 \\ y &= a\cosh\lambda\cos\mu\sin\theta & \frac{-\pi}{2} &\leq \mu \leq \frac{\pi}{2} \\ z &= a\sinh\lambda\sin\mu & 0 &\leq \theta \leq 2\pi \end{aligned} \tag{4.78}$$

where $a \geq 0$ is the focus.

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} a\sinh\lambda\cos\mu\cos\theta\boldsymbol{i}_1 + a\sinh\lambda\cos\mu\sin\theta\boldsymbol{i}_2 + a\cosh\lambda\sin\mu\boldsymbol{i}_3 \\ -a\cosh\lambda\sin\mu\cos\theta\boldsymbol{i}_1 - a\cosh\lambda\sin\mu\sin\theta\boldsymbol{i}_2 + a\sinh\lambda\cos\mu\boldsymbol{i}_3 \\ -a\cosh\lambda\cos\mu\sin\theta\boldsymbol{i}_1 + a\cosh\lambda\cos\mu\cos\theta\boldsymbol{i}_2 \end{bmatrix} \tag{4.79}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 & 0 \\ 0 & a^2\left(\sinh^2\lambda + \sin^2\mu\right) & 0 \\ 0 & 0 & a^2\cosh^2\lambda\cos^2\mu \end{bmatrix} \tag{4.80}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 & 0 \\ 0 & \frac{1}{a^2\left(\sinh^2\lambda+\sin^2\mu\right)} & 0 \\ 0 & 0 & \frac{1}{a^2\cosh^2\lambda\cos^2\mu} \end{bmatrix} \tag{4.81}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\lambda}_{\lambda\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{4.82}$$

$$\Gamma^{\lambda}_{\mu\mu} = \frac{-\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{4.83}$$

$$\Gamma^{\lambda}_{\theta\theta} = \frac{-\sinh\lambda\cosh\lambda\cos^2\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.84}$$

$$\Gamma^{\lambda}_{\lambda\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.85}$$

$$\Gamma^{\mu}_{\mu\mu} = \frac{\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.86}$$

$$\Gamma^{\mu}_{\lambda\lambda} = \frac{-\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.87}$$

$$\Gamma^{\mu}_{\theta\theta} = \frac{\cosh^2\lambda\sin\mu\cos\mu}{\sinh^2\lambda + \sin^2\mu} \tag{4.88}$$

$$\Gamma^{\mu}_{\mu\lambda} = \frac{\sinh\lambda\cosh\lambda}{\sinh^2\lambda + \sin^2\mu} \tag{4.89}$$

$$\Gamma^{\theta}_{\theta\lambda} = \frac{\sinh\lambda}{\cosh\lambda} \tag{4.90}$$

$$\Gamma^{\theta}_{\theta\mu} = \frac{-\sin\mu}{\cos\mu} \tag{4.91}$$

$$\tag{4.92}$$

with all other Christofell symbols zero.

Cylindrical parabolic

The global coordinates (x, y, z) with respect to the cylindrical parabolic coordinates (ξ, η, z) are defined by

$$\begin{aligned} x &= \xi\eta && -\infty < \xi < \infty \\ y &= \frac{1}{2}\left(\xi^2 - \eta^2\right) && \eta \geq 0 \\ z &= z && -\infty < z < \infty \end{aligned} \tag{4.93}$$

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \eta\boldsymbol{i}_1 + \xi\boldsymbol{i}_2 \\ \xi\boldsymbol{i}_1 - \eta\boldsymbol{i}_2 \\ \boldsymbol{i}_3 \end{bmatrix} \tag{4.94}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} \xi^2+\eta^2 & 0 & 0 \\ 0 & \xi^2+\eta^2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.95}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{\xi^2+\eta^2} & 0 & 0 \\ 0 & \frac{1}{\xi^2+\eta^2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.96}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\xi}_{\xi\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{4.97}$$

$$\Gamma^{\eta}_{\eta\eta} = \frac{\eta}{\xi^2+\eta^2} \tag{4.98}$$

$$\Gamma^{\eta}_{\xi\xi} = \frac{-\eta}{\xi^2+\eta^2} \tag{4.99}$$

$$\Gamma^{\xi}_{\eta\eta} = \frac{-\xi}{\xi^2+\eta^2} \tag{4.100}$$

$$\Gamma^{\xi}_{\xi\eta} = \Gamma^{\xi}_{\eta\xi} = \frac{\eta}{\xi^2+\eta^2} \tag{4.101}$$

$$\Gamma^{\eta}_{\xi\eta} = \Gamma^{\eta}_{\eta\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{4.102}$$

$$\tag{4.103}$$

with all other Christofell symbols zero.

Parabolic polar

The global coordinates (x, y, z) with respect to the cylindrical parabolic coordinates (ξ, η, θ) are defined by

$$\begin{aligned} x &= \xi\eta\cos\theta && \xi \geq 0 \\ y &= \xi\eta\sin\theta && \eta \geq 0 \\ z &= \frac{1}{2}\left(\xi^2 - \eta^2\right) && 0 \leq \theta < 2\pi \end{aligned} \tag{4.104}$$

The base vectors with respect to the global coordinate system are

$$\boldsymbol{g}_i = \begin{bmatrix} \eta\cos\theta\boldsymbol{i}_1 + \eta\sin\theta\boldsymbol{i}_3 + \xi\boldsymbol{i}_3 \\ \xi\cos\theta\boldsymbol{i}_1 + \xi\sin\theta\boldsymbol{i}_3 - \eta\boldsymbol{i}_3 \\ -\xi\eta\sin\theta\boldsymbol{i}_1 + \xi\eta\cos\theta\boldsymbol{i}_2 \end{bmatrix} \tag{4.105}$$

The covariant metric tensor is

$$g_{ij} = \begin{bmatrix} \xi^2+\eta^2 & 0 & 0 \\ 0 & \xi^2+\eta^2 & 0 \\ 0 & 0 & \xi\eta \end{bmatrix} \tag{4.106}$$

and the contravariant metric tensor is

$$g^{ij} = \begin{bmatrix} \frac{1}{\xi^2+\eta^2} & 0 & 0 \\ 0 & \frac{1}{\xi^2+\eta^2} & 0 \\ 0 & 0 & \frac{1}{\xi\eta} \end{bmatrix} \tag{4.107}$$

The Christoffell symbols of the second kind are

$$\Gamma^{\xi}_{\xi\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{4.108}$$

$$\Gamma^{\eta}_{\eta\eta} = \frac{\eta}{\xi^2+\eta^2} \tag{4.109}$$

$$\Gamma^{\xi}_{\eta\eta} = \frac{-\xi}{\xi^2+\eta^2} \tag{4.110}$$

$$\Gamma^{\eta}_{\xi\xi} = \frac{-\eta}{\xi^2+\eta^2} \tag{4.111}$$

$$\Gamma^{\eta}_{\theta\theta} = \frac{-\xi^2\eta}{\xi^2+\eta^2} \tag{4.112}$$

$$\Gamma^{\xi}_{\theta\theta} = \frac{-\xi\eta^2}{\xi^2+\eta^2} \tag{4.113}$$

$$\Gamma^{\xi}_{\xi\eta} = \Gamma^{\xi}_{\eta\xi} = \frac{\eta}{\xi^2+\eta^2} \tag{4.114}$$

$$\Gamma^{\eta}_{\xi\eta} = \Gamma^{\eta}_{\eta\xi} = \frac{\xi}{\xi^2+\eta^2} \tag{4.115}$$

$$\Gamma^{\theta}_{\xi\theta} = \Gamma^{\theta}_{\theta\xi} = \frac{1}{\xi} \tag{4.116}$$

$$\Gamma^{\theta}_{\eta\theta} = \Gamma^{\theta}_{\theta\eta} = \frac{1}{\eta} \tag{4.117}$$

$$\tag{4.118}$$

with all other Christofell symbols zero.

4.2 Equation set types

4.2.1 Static Equations

The general form for static equations is

4.2.2 Dynamic Equations

The general form for dynamic equations is

$$\boldsymbol{M}\ddot{\boldsymbol{u}}\left(t\right)+\boldsymbol{C}\dot{\boldsymbol{u}}\left(t\right)+\boldsymbol{K}\boldsymbol{u}\left(t\right)+\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)+\boldsymbol{f}\left(t\right)=\boldsymbol{0} \tag{4.119}$$

where $\boldsymbol{u}\left(t\right)$ is the unknown "displacement vector", $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{C}$ is the damping matrix, $\boldsymbol{K}$ is the stiffness matrix, $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ a non-linear vector function and $\boldsymbol{f}\left(t\right)$ the forcing vector.

From (**?**) we now expand the unknown vector $\boldsymbol{u}\left(t\right)$ in terms of a polynomial of degree p. With the known values of $\boldsymbol{u}_n$, $\dot{\boldsymbol{u}}_n$, $\ddot{\boldsymbol{u}}_n$ up to $\overset{p-1}{\boldsymbol{u}}_n$ at the beginning of the time step Δt we can write the polynomial expansion as

$$\boldsymbol{u}\left(t_n+\tau\right)\approx\tilde{\boldsymbol{u}}\left(t_n+\tau\right)=\boldsymbol{u}_n+\tau\dot{\boldsymbol{u}}_n+\frac{1}{2!}\tau^2\ddot{\boldsymbol{u}}_n+\cdots+\frac{1}{(p-1)!}\tau^{p-1}\overset{p-1}{\boldsymbol{u}}_n+\frac{1}{p!}\tau^p\boldsymbol{\alpha}_n^p \tag{4.120}$$

where the only unknown is the the vector $\boldsymbol{\alpha}_n^p$,

$$\boldsymbol{\alpha}_n^p\approx\overset{p}{\boldsymbol{u}}\equiv\frac{d^p\boldsymbol{u}}{dt^p} \tag{4.121}$$

A recurrance relationship can be established by substituting Equation (4.120) into Equation (6.384) and taking a weighted residual approach *i.e.,*

$$\begin{aligned}\int_0^{\Delta t}W\left(\tau\right)\Bigg[\boldsymbol{M}\left(\ddot{\boldsymbol{u}}_n+\tau\dddot{\boldsymbol{u}}_n+\cdots+\frac{1}{(p-2)!}\tau^{p-2}\boldsymbol{\alpha}_n^p\right)&\\+\boldsymbol{C}\left(\dot{\boldsymbol{u}}_n+\tau\ddot{\boldsymbol{u}}_n+\cdots+\frac{1}{(p-1)!}\tau^{p-1}\boldsymbol{\alpha}_n^p\right)&\\+\boldsymbol{K}\left(\boldsymbol{u}_n+\tau\dot{\boldsymbol{u}}_n+\cdots+\frac{1}{p!}\tau^p\boldsymbol{\alpha}_n^p\right)&\\+\boldsymbol{g}\left(\boldsymbol{u}_n+\tau\dot{\boldsymbol{u}}_n+\cdots+\frac{1}{p!}\tau^p\boldsymbol{\alpha}_n^p\right)+\boldsymbol{f}\left(t_n+\tau\right)\Bigg]d\tau=\boldsymbol{0}&\end{aligned} \tag{4.122}$$

where $W\left(\tau\right)$ is some weight function, $\tau=t-t_n$ and $\Delta t=t_{n+1}-t_n$. Dividing by $\int_0^{\Delta t}W\left(\tau\right)\,\mathbf{d}\tau$

we obtain

$$\frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{M}\left(\ddot{\boldsymbol{u}}_n + \tau\dddot{\boldsymbol{u}}_n + \cdots + \frac{1}{(p-2)!}\tau^{p-2}\boldsymbol{\alpha}_n^p\right)\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{C}\left(\dot{\boldsymbol{u}}_n + \tau\ddot{\boldsymbol{u}}_n + \cdots + \frac{1}{(p-1)!}\tau^{p-1}\boldsymbol{\alpha}_n^p\right)\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{K}\left(\boldsymbol{u}_n + \tau\dot{\boldsymbol{u}}_n + \cdots + \frac{1}{p!}\tau^{p}\boldsymbol{\alpha}_n^p\right)\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}\left(\boldsymbol{u}_n + \tau\dot{\boldsymbol{u}}_n + \cdots + \frac{1}{p!}\tau^{p}\boldsymbol{\alpha}_n^p\right)\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{f}(t_n+\tau)\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} = \mathbf{0} \quad (4.123)$$

Now we define

$$\bar{\boldsymbol{g}} = \frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}\left(\boldsymbol{u}(t_n+\tau)\right)\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \quad (4.124)$$

and

$$\bar{\boldsymbol{f}} = \frac{\int\limits_0^{\Delta t} W(\tau)\, \boldsymbol{f}(t_n+\tau)\ \mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\ \mathbf{d}\tau} \tag{4.125}$$

and

$$\theta_k = \frac{\int\limits_0^{\Delta t} W(\tau)\, \tau^k\, \mathbf{d}\tau}{\Delta t^k \int\limits_0^{\Delta t} W(\tau)\ \mathbf{d}\tau} \text{ for } k = 0, 1, \ldots, p \tag{4.126}$$

We can now write

$$\boldsymbol{M}\left(\ddot{\bar{\boldsymbol{u}}}_{n+1} + \frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{\alpha}_n^p\right) + \boldsymbol{C}\left(\dot{\bar{\boldsymbol{u}}}_{n+1} + \frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{\alpha}_n^p\right) + \boldsymbol{K}\left(\bar{\boldsymbol{u}}_{n+1} + \frac{\theta_p\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right) + \\ + \frac{\int\limits_0^{\Delta t} W(\tau)\, \boldsymbol{g}\left(\boldsymbol{u}_n + \tau\dot{\boldsymbol{u}}_n + \cdots + \frac{1}{p!}\tau^p\boldsymbol{\alpha}_n^p\right)\ \mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\ \mathbf{d}\tau} + \bar{\boldsymbol{f}} = \mathbf{0} \tag{4.127}$$

where

$$\begin{aligned} \bar{\boldsymbol{u}}_{n+1} &= \sum_{q=0}^{p-1} \frac{\theta_q\Delta t^q}{q!}\ \overset{q}{\boldsymbol{u}}_n \\ \dot{\bar{\boldsymbol{u}}}_{n+1} &= \sum_{q=1}^{p-1} \frac{\theta_{q-1}\Delta t^{q-1}}{(q-1)!}\ \overset{q}{\boldsymbol{u}}_n \\ \ddot{\bar{\boldsymbol{u}}}_{n+1} &= \sum_{q=2}^{p-1} \frac{\theta_{q-2}\Delta t^{q-2}}{(q-2)!}\ \overset{q}{\boldsymbol{u}}_n \end{aligned} \tag{4.128}$$

We note that as $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ is nonlinear we need to evaluate an integral of the form

$$\int_{0}^{\Delta t} W\left(\tau\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\,\mathbf{d}\tau \tag{4.129}$$

In order to ensure that the accuracy of the integration scheme is not compromised we need to evaluate Equation (4.129) with p^{th} order accuracy in time. For a first order in time approximation we form Taylor's series expansions for $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ about the point $\boldsymbol{u}\left(t_n+\tau\right)$ *i.e.,*

$$\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)-\tau\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\tau^2\right) \tag{4.130}$$

and

$$\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\left(\Delta t-\tau\right)\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\left(\Delta t-\tau\right)^2\right) \tag{4.131}$$

Now if we sum Equation (4.130) times $\dfrac{\Delta t-\tau}{\Delta t}$ and Equation (4.131) times $\dfrac{\tau}{\Delta t}$ we obtain

$$\frac{\Delta t-\tau}{\Delta t}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau}{\Delta t}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\tau^2\right) \tag{4.132}$$

Equation (4.129) now becomes

$$\int_{0}^{\Delta t} W\left(\tau\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\,\mathbf{d}\tau=\int_{0}^{\Delta t} W\left(\tau\right)\left(\frac{\Delta t-\tau}{\Delta t}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau}{\Delta t}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)-\mathrm{O}\left(\tau^2\right)\right)\,\mathbf{d}\tau \tag{4.133}$$

or

$$\int_{0}^{\Delta t} W\left(\tau\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\,\mathbf{d}\tau=\int_{0}^{\Delta t} W\left(\tau\right)\left(\left(1-\frac{\tau}{\Delta t}\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau}{\Delta t}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)-\mathrm{O}\left(\tau^2\right)\right)\,\mathbf{d}\tau \tag{4.134}$$

and so

$$\frac{\int\limits_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n+\tau))\,\mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} = \frac{\int\limits_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} - \frac{\int\limits_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{\Delta t\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$$
$$+ \frac{\int\limits_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))\,\mathbf{d}\tau}{\Delta t\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} - \frac{\int\limits_0^{\Delta t} W(\tau)\,\mathrm{O}\left(\tau^2\right)\,\mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \qquad (4.135)$$

Now if we recall that $\theta_0 = \frac{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} = 1$ and $\theta_1 = \frac{\int\limits_0^{\Delta t} W(\tau)\,\tau\,\mathbf{d}\tau}{\Delta t\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$ we can write

$$\bar{\boldsymbol{g}} = (1-\theta_1)\,\boldsymbol{g}(\boldsymbol{u}(t_n)) + \theta_1\boldsymbol{g}(\boldsymbol{u}(t_{n+1})) - \text{Error} \qquad (4.136)$$

where

$$\text{Error} = \frac{\int\limits_0^{\Delta t} W(\tau)\,\mathrm{O}\left(\tau^2\right)\,\mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \qquad (4.137)$$

For a second order in time approximation we need to consider three points. We form Taylor's series expansions for $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ about the point $\boldsymbol{u}\left(t_n+\tau\right)$ ***i.e.,***

$$\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)-\left(\Delta t+\tau\right)\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) \\ +\frac{\left(\Delta t+\tau\right)^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\left(\Delta t+\tau\right)^3\right) \quad (4.138)$$

and

$$\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)-\tau\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\frac{\tau^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\tau^3\right) \quad (4.139)$$

and

$$\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\left(\Delta t-\tau\right)\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) \\ +\frac{\left(\Delta t-\tau\right)^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\left(\Delta t-\tau\right)^3\right) \quad (4.140)$$

If we sum the equations Equation (4.138) times $\frac{-\tau\left(\Delta t-\tau\right)}{2\Delta t^2}$, Equation (4.139) times $\frac{2\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)}{2\Delta t^2}$, and Equation (4.140) times $\frac{\tau\left(\Delta t+\tau\right)}{2\Delta t^2}$ we obtain

$$\frac{-\tau\left(\Delta t-\tau\right)}{2\Delta t^2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right)+\frac{2\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)}{2\Delta t^2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau\left(\Delta t+\tau\right)}{2\Delta t^2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right) \\ =\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\tau^3\right) \quad (4.141)$$

Equation (4.129) now becomes

$$\int_0^{\Delta t} W\left(\tau\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\,\mathbf{d}\tau=\int_0^{\Delta t} W\left(\tau\right)\left(\frac{-\tau\left(\Delta t-\tau\right)}{2\Delta t^2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right)+\right. \\ \left.\frac{2\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)}{2\Delta t^2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau\left(\Delta t+\tau\right)}{2\Delta t^2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)-\mathrm{O}\left(\tau^3\right)\right)\mathbf{d}\tau \quad (4.142)$$

or

$$\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n+\tau))\,\mathbf{d}\tau = \int_0^{\Delta t} W(\tau)\left(\left(\frac{-\tau}{2\Delta t}+\frac{\tau^2}{2\Delta t^2}\right)\boldsymbol{g}(\boldsymbol{u}(t_{n-1}))\right.$$
$$\left.+\left(1-\frac{\tau^2}{\Delta t^2}\right)\boldsymbol{g}(\boldsymbol{u}(t_n))+\left(\frac{\tau}{2\Delta t}+\frac{\tau^2}{2\Delta t^2}\right)\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))-\mathrm{O}\left(\tau^3\right)\right)\mathbf{d}\tau \quad (4.143)$$

and so

$$\frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n+\tau))\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} = \frac{\int_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_{n-1}))\,\mathbf{d}\tau}{2\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} + \frac{\int_0^{\Delta t} W(\tau)\,\tau^2\boldsymbol{g}(\boldsymbol{u}(t_{n-1}))\,\mathbf{d}\tau}{2\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} +$$
$$\frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} - \frac{\int_0^{\Delta t} W(\tau)\,\tau^2\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} + \frac{\int_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))\,\mathbf{d}\tau}{2\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} +$$
$$\frac{\int_0^{\Delta t} W(\tau)\,\tau^2\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))\,\mathbf{d}\tau}{2\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} - \frac{\int_0^{\Delta t} W(\tau)\,\mathrm{O}\left(\tau^3\right)\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \quad (4.144)$$

Now if we recall $\theta_0 = \frac{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} = 1, \theta_1 = \frac{\int_0^{\Delta t} W(\tau)\,\tau\,\mathbf{d}\tau}{\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$, and $\theta_2 = \frac{\int_0^{\Delta t} W(\tau)\,\tau^2\,\mathbf{d}\tau}{\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$

we can write

$$\bar{\boldsymbol{g}} = \frac{(\theta_2 - \theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right) + \left(1 - \theta_2\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right) + \frac{(\theta_2 + \theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right) - \text{Error} \tag{4.145}$$

where

$$\text{Error} = \frac{\int\limits_0^{\Delta t} W(\tau)\,\mathrm{O}\left(\tau^3\right)\,\mathbf{d}\tau}{\int\limits_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \tag{4.146}$$

For a third order in time approximation we need to consider four points. We form Taylor's series expansions for $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ about the point $\boldsymbol{u}\left(t_n + \tau\right)$ *i.e.*,

$$\begin{aligned}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-2}\right)\right) = \boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) - \left(2\Delta t+\tau\right)\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) + \frac{\left(2\Delta t+\tau\right)^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\\ - \frac{\left(2\Delta t+\tau\right)^3}{6}\dddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) + \mathrm{O}\left(\left(2\Delta t+\tau\right)^4\right)\end{aligned} \tag{4.147}$$

and

$$\begin{aligned}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right) = \boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) - \left(\Delta t+\tau\right)\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) + \frac{\left(\Delta t+\tau\right)^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\\ - \frac{\left(\Delta t+\tau\right)^3}{6}\dddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) + \mathrm{O}\left(\left(\Delta t+\tau\right)^4\right)\end{aligned} \tag{4.148}$$

and

$$\begin{aligned}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right) = \boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) - \tau\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) + \frac{\tau^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\\ - \frac{\tau^3}{6}\dddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right) + \mathrm{O}\left(\tau^4\right)\end{aligned} \tag{4.149}$$

and

$$\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\left(\Delta t-\tau\right)\dot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\frac{\left(\Delta t-\tau\right)^2}{2}\ddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\frac{\left(\Delta t-\tau\right)^3}{6}\dddot{\boldsymbol{g}}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\left(\Delta t-\tau\right)^4\right) \quad (4.150)$$

If we sum Equation (4.147) times $\frac{\tau\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)}{6\Delta t^3}$, Equation (4.148) times $\frac{-3\tau\left(\Delta t-\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}$, Equation (4.149) times $\frac{3\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}$, and Equation (4.150) times $\frac{\tau\left(\Delta t+\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}$ we obtain

$$\frac{\tau\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-2}\right)\right)-\frac{3\tau\left(\Delta t-\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right)+\frac{3\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau\left(\Delta t+\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)=\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)+\mathrm{O}\left(\tau^4\right) \quad (4.151)$$

Equation (4.129) now becomes

$$\int_0^{\Delta t} W\left(\tau\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\,\mathbf{d}\tau=\int_0^{\Delta t} W\left(\tau\right)\left(\frac{\tau\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-2}\right)\right)-\frac{3\tau\left(\Delta t-\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right)+\frac{3\left(\Delta t-\tau\right)\left(\Delta t+\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\frac{\tau\left(\Delta t+\tau\right)\left(2\Delta t+\tau\right)}{6\Delta t^3}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)-\mathrm{O}\left(\tau^4\right)\right)\mathbf{d}\tau \quad (4.152)$$

or

$$\int_0^{\Delta t} W\left(\tau\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n+\tau\right)\right)\,\mathbf{d}\tau=\int_0^{\Delta t} W\left(\tau\right)\left(\left(\frac{\tau}{6\Delta t}-\frac{\tau^3}{6\Delta t^3}\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-2}\right)\right)+\left(\frac{-\tau}{\Delta t}+\frac{\tau^2}{2\Delta t^2}+\frac{\tau^3}{2\Delta t^3}\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right)+\left(1+\frac{\tau}{2\Delta t}-\frac{\tau^2}{\Delta t^2}-\frac{\tau^3}{2\Delta t^3}\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right)+\left(\frac{\tau}{3\Delta t}+\frac{\tau^2}{2\Delta t^2}+\frac{\tau^3}{6\Delta t^3}\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right)-\mathrm{O}\left(\tau^4\right)\right)\mathbf{d}\tau \quad (4.153)$$

and so

$$
\begin{aligned}
\frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n+\tau))\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
&= \frac{\int_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_{n-2}))\,\mathbf{d}\tau}{6\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
- \frac{\int_0^{\Delta t} W(\tau)\,\tau^3\boldsymbol{g}(\boldsymbol{u}(t_{n-2}))\,\mathbf{d}\tau}{6\Delta t^3\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \\
&- \frac{\int_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_{n-1}))\,\mathbf{d}\tau}{\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\tau^2\boldsymbol{g}(\boldsymbol{u}(t_{n-1}))\,\mathbf{d}\tau}{2\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\tau^3\boldsymbol{g}(\boldsymbol{u}(t_{n-1}))\,\mathbf{d}\tau}{2\Delta t^3\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \\
&+ \frac{\int_0^{\Delta t} W(\tau)\,\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{2\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
- \frac{\int_0^{\Delta t} W(\tau)\,\tau^2\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \\
&- \frac{\int_0^{\Delta t} W(\tau)\,\tau^3\boldsymbol{g}(\boldsymbol{u}(t_n))\,\mathbf{d}\tau}{2\Delta t^3\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\tau\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))\,\mathbf{d}\tau}{3\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
+ \frac{\int_0^{\Delta t} W(\tau)\,\tau^2\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))\,\mathbf{d}\tau}{2\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \\
&+ \frac{\int_0^{\Delta t} W(\tau)\,\tau^3\boldsymbol{g}(\boldsymbol{u}(t_{n+1}))\,\mathbf{d}\tau}{6\Delta t^3\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
- \frac{\int_0^{\Delta t} W(\tau)\,\mathrm{O}\left(\tau^4\right)\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}
\end{aligned}
\tag{4.154}
$$

If we recall that $\theta_0 = \dfrac{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} = 1$, $\theta_1 = \dfrac{\int_0^{\Delta t} W(\tau)\,\tau\,\mathbf{d}\tau}{\Delta t\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$, $\theta_2 = \dfrac{\int_0^{\Delta t} W(\tau)\,\tau^2\,\mathbf{d}\tau}{\Delta t^2\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$, and $\theta_3 = \dfrac{\int_0^{\Delta t} W(\tau)\,\tau^3\,\mathbf{d}\tau}{\Delta t^3\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau}$ we can write

$$\bar{\boldsymbol{g}} = \frac{(\theta_1 - \theta_3)}{6}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-2}\right)\right) + \frac{(\theta_3 + \theta_2 - 2\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n-1}\right)\right) + \left(1 - \frac{(\theta_3 + 2\theta_2 - \theta_1)}{2}\right)\boldsymbol{g}\left(\boldsymbol{u}\left(t_n\right)\right) + \frac{(\theta_3 + 3\theta_2 + 2\theta_1)}{6}\boldsymbol{g}\left(\boldsymbol{u}\left(t_{n+1}\right)\right) - \text{Error} \quad (4.155)$$

where

$$\text{Error} = \frac{\int_0^{\Delta t} W(\tau)\,\mathrm{O}\left(\tau^4\right)\,\mathbf{d}\tau}{\int_0^{\Delta t} W(\tau)\,\mathbf{d}\tau} \quad (4.156)$$

Equation (4.127) now becomes

$$\boldsymbol{M}\left(\ddot{\boldsymbol{u}}_{n+1} + \frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{\alpha}_n^p\right) + \boldsymbol{C}\left(\dot{\boldsymbol{u}}_{n+1} + \frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{\alpha}_n^p\right) + \boldsymbol{K}\left(\bar{\boldsymbol{u}}_{n+1} + \frac{\theta_p\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right) + (1-\theta_1)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \theta_1\boldsymbol{g}\left(\boldsymbol{u}_{n+1}\right) + \bar{\boldsymbol{f}} + \text{Error} = \boldsymbol{0} \quad (4.157)$$

with a first order approximation in time or

$$\boldsymbol{M}\left(\ddot{\boldsymbol{u}}_{n+1}+\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{\alpha}_n^p\right)+\boldsymbol{C}\left(\dot{\boldsymbol{u}}_{n+1}+\frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{\alpha}_n^p\right)+\boldsymbol{K}\left(\bar{\boldsymbol{u}}_{n+1}+\frac{\theta_p\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)$$
$$+\frac{(\theta_2-\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right)+(1-\theta_2)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right)+\frac{(\theta_2+\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n+1}\right)+\bar{\boldsymbol{f}}-\text{Error}=\boldsymbol{0} \quad (4.158)$$

with a second order approximation in time or

$$\boldsymbol{M}\left(\ddot{\boldsymbol{u}}_{n+1}+\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{\alpha}_n^p\right)+\boldsymbol{C}\left(\dot{\boldsymbol{u}}_{n+1}+\frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{\alpha}_n^p\right)+\boldsymbol{K}\left(\bar{\boldsymbol{u}}_{n+1}+\frac{\theta_p\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)$$
$$+\frac{(\theta_1-\theta_3)}{6}\boldsymbol{g}\left(\boldsymbol{u}_{n-2}\right)+\frac{(\theta_3+\theta_2-2\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right)+\left(1-\frac{(\theta_3+2\theta_2-\theta_1)}{2}\right)\boldsymbol{g}\left(\boldsymbol{u}_n\right)$$
$$+\frac{(\theta_3+3\theta_2+2\theta_1)}{6}\boldsymbol{g}\left(\boldsymbol{u}_{n+1}\right)+\bar{\boldsymbol{f}}-\text{Error}=\boldsymbol{0} \quad (4.159)$$

with a third order approximation in time as $\boldsymbol{u}\left(t_{n-2}\right)=\boldsymbol{u}_{n-2}$, $\boldsymbol{u}\left(t_{n-1}\right)=\boldsymbol{u}_{n-1}$, $\boldsymbol{u}\left(t_n\right)=\boldsymbol{u}_n$ and $\boldsymbol{u}\left(t_{n+1}\right)=\boldsymbol{u}_{n+1}=\hat{\boldsymbol{u}}_{n+1}+\frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p$ where $\hat{\boldsymbol{u}}_{n+1}$ is the *predicted displacement* at the new time step and is given by

$$\hat{\boldsymbol{u}}_{n+1}=\sum_{q=0}^{p-1}\frac{\Delta t^q}{q!}\overset{q}{\boldsymbol{u}}_n \quad (4.160)$$

Rearranging gives

$$\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^p\right)=\left(\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M}+\frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C}+\frac{\theta_p\Delta t^p}{p!}\boldsymbol{K}\right)\boldsymbol{\alpha}_n^p+\theta_1\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1}+\frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)$$
$$+(1-\theta_1)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right)+\left(\boldsymbol{M}\ddot{\boldsymbol{u}}_{n+1}+\boldsymbol{C}\dot{\boldsymbol{u}}_{n+1}+\boldsymbol{K}\bar{\boldsymbol{u}}_{n+1}+\bar{\boldsymbol{f}}\right)=\boldsymbol{0} \quad (4.161)$$

for a first order approximation or

$$\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^p\right)=\left(\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M}+\frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C}+\frac{\theta_p\Delta t^p}{p!}\boldsymbol{K}\right)\boldsymbol{\alpha}_n^p+\frac{(\theta_2+\theta_1)}{2}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1}+\frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)$$
$$+(1-\theta_2)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right)+\frac{(\theta_2-\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right)+\left(\boldsymbol{M}\ddot{\boldsymbol{u}}_{n+1}+\boldsymbol{C}\dot{\boldsymbol{u}}_{n+1}+\boldsymbol{K}\bar{\boldsymbol{u}}_{n+1}+\bar{\boldsymbol{f}}\right)=\boldsymbol{0} \quad (4.162)$$

for a second order approximation or

$$\begin{aligned}\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^p\right) = \left(\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M} + \frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C} + \frac{\theta_p \Delta t^p}{p!}\boldsymbol{K}\right)\boldsymbol{\alpha}_n^p + \frac{(\theta_3+3\theta_2+2\theta_1)}{6}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right) \\ + \left(1 - \frac{(\theta_3+2\theta_2-\theta_1)}{2}\right)\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \frac{(\theta_3+\theta_2-2\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right) + \frac{(\theta_1-\theta_3)}{6}\boldsymbol{g}\left(\boldsymbol{u}_{n-2}\right) \\ + \left(\boldsymbol{M}\ddot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{C}\dot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{K}\bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}}\right) = \boldsymbol{0} \quad (4.163)\end{aligned}$$

for a third order approximation.

If we now define the *Amplification matrix* $\boldsymbol{A}$ as

$$\boldsymbol{A} = \frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M} + \frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C} + \frac{\theta_p \Delta t^p}{p!}\boldsymbol{K} \quad (4.164)$$

and the right hand side vector $\boldsymbol{b}$ as

$$\boldsymbol{b} = \boldsymbol{M}\ddot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{C}\dot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{K}\bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}} \quad (4.165)$$

We can now write Equation (4.161) as

$$\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^p\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^p + \theta_1\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right) + (1-\theta_1)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \boldsymbol{b} = \boldsymbol{0} \quad (4.166)$$

and Equation (4.162) as

$$\begin{aligned}\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^p\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^p + \frac{(\theta_2+\theta_1)}{2}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right) \\ + (1-\theta_2)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \frac{(\theta_2-\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right) + \boldsymbol{b} = \boldsymbol{0} \quad (4.167)\end{aligned}$$

and Equation (4.163) as

$$\begin{aligned}\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^p\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^p + \frac{(\theta_3 + 3\theta_2 + 2\theta_1)}{6}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right) \\ + \left(1 - \frac{(\theta_3 + 2\theta_2 - \theta_1)}{2}\right)\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \frac{(\theta_3 + \theta_2 - 2\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right) \\ + \frac{(\theta_1 - \theta_3)}{6}\boldsymbol{g}\left(\boldsymbol{u}_{n-2}\right) + \boldsymbol{b} = \boldsymbol{0} \end{aligned} \quad (4.168)$$

If $\boldsymbol{g}\left(\boldsymbol{u}\right) \equiv \boldsymbol{0}$ then Equation (4.161) or Equation (4.162) is linear in $\boldsymbol{\alpha}_n^p$ and $\boldsymbol{\alpha}_n^p$ can be found by solving the linear equation

$$\boldsymbol{\alpha}_n^p = -\left(\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M} + \frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C} + \frac{\theta_p\Delta t^p}{p!}\boldsymbol{K}\right)^{-1}\left(\boldsymbol{M}\ddot{\boldsymbol{u}}_{n+1} + \boldsymbol{C}\dot{\boldsymbol{u}}_{n+1} + \boldsymbol{K}\bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}}\right) \quad (4.169)$$

or

$$\boldsymbol{\alpha}_n^p = -\boldsymbol{A}^{-1}\boldsymbol{b} \quad (4.170)$$

If $\boldsymbol{g}\left(\boldsymbol{u}\right)$ is not $\equiv \boldsymbol{0}$ then Equation (4.161) or Equation (4.162) is nonlinear in $\boldsymbol{\alpha}_n^p$. To solve this equation we use Newton's method *i.e.,*

$$\begin{aligned} &1.\ \boldsymbol{J}\left(\boldsymbol{\alpha}_{n(i)}^p\right).\delta\boldsymbol{\alpha}_{n(i)}^p = -\boldsymbol{\psi}\left(\boldsymbol{\alpha}_{n(i)}^p\right) \\ &2.\ \boldsymbol{\alpha}_{n(i+1)}^p = \boldsymbol{\alpha}_{n(i)}^p + \delta\boldsymbol{\alpha}_{n(i)}^p \end{aligned} \quad (4.171)$$

where $\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right)$ is the Jacobian and is given by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right) = \frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M} + \frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C} + \frac{\theta_p\Delta t^p}{p!}\boldsymbol{K} + \frac{\theta_1\Delta t^p}{p!}\frac{\partial\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)}{\partial\boldsymbol{\alpha}_n^p} \quad (4.172)$$

or

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right) = \boldsymbol{A} + \frac{\theta_1\Delta t^p}{p!}\frac{\partial\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)}{\partial\boldsymbol{\alpha}_n^p} \quad (4.173)$$

for a first order approximation or by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right)=\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M}+\frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C}+\frac{\theta_p\Delta t^p}{p!}\boldsymbol{K}+\frac{(\theta_2+\theta_1)\,\Delta t^p}{2p!}\frac{\partial\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1}+\dfrac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)}{\partial\boldsymbol{\alpha}_n^p} \tag{4.174}$$

or

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right)=\boldsymbol{A}+\frac{(\theta_2+\theta_1)\,\Delta t^p}{2p!}\frac{\partial\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1}+\dfrac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)}{\partial\boldsymbol{\alpha}_n^p} \tag{4.175}$$

for a second order approximation or by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right)=\frac{\theta_{p-2}\Delta t^{p-2}}{(p-2)!}\boldsymbol{M}+\frac{\theta_{p-1}\Delta t^{p-1}}{(p-1)!}\boldsymbol{C}+\frac{\theta_p\Delta t^p}{p!}\boldsymbol{K}+\frac{(\theta_3+3\theta_2+2\theta_1)\,\Delta t^p}{6p!}\frac{\partial\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1}+\dfrac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)}{\partial\boldsymbol{\alpha}_n^p} \tag{4.176}$$

or

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^p\right)=\boldsymbol{A}+\frac{(\theta_3+3\theta_2+2\theta_1)\,\Delta t^p}{6p!}\frac{\partial\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1}+\dfrac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\right)}{\partial\boldsymbol{\alpha}_n^p} \tag{4.177}$$

for a third order approximation.

Once $\boldsymbol{\alpha}_n^p$ has been obtained the values at the next time step can be obtained from

$$\begin{aligned}
\boldsymbol{u}_{n+1}&=\boldsymbol{u}_n+\Delta t\dot{\boldsymbol{u}}_n+\cdots+\frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p=\hat{\boldsymbol{u}}_{n+1}+\frac{\Delta t^p}{p!}\boldsymbol{\alpha}_n^p\\
\dot{\boldsymbol{u}}_{n+1}&=\dot{\boldsymbol{u}}_n+\Delta t\ddot{\boldsymbol{u}}_n+\cdots+\frac{\Delta t^{p-1}}{(p-1)!}\boldsymbol{\alpha}_n^p=\dot{\hat{\boldsymbol{u}}}_{n+1}+\frac{\Delta t^{p-1}}{(p-1)!}\boldsymbol{\alpha}_n^p\\
&\vdots\\
\overset{p-1}{\boldsymbol{u}}_{n+1}&=\overset{p-1}{\boldsymbol{u}}_{n}+\Delta t\boldsymbol{\alpha}_n^p
\end{aligned} \tag{4.178}$$

For algorithms in which the degree of the polynomial, p, is higher than the order we require the algorithm to be initialised so that the initial velocity or acceleration can be computed. The initial velocity or acceleration values can be obtained by substituting the initial displacement or initial displacement and velocity values into Equation (6.384), rearranging and solving. For example consider the case of a second degree polynomial and a first order system. Substituing the initial displacement $\boldsymbol{u}_0$ into Equation (6.384) gives

$$\boldsymbol{C}\dot{\boldsymbol{u}}_0 + \boldsymbol{K}\boldsymbol{u}_0 + \boldsymbol{g}\left(\boldsymbol{u}_0\right) + \bar{\boldsymbol{f}}_0 = \boldsymbol{0} \tag{4.179}$$

and therefore an approximation to the initial velocity can be found from

$$\dot{\boldsymbol{u}}_0 = -\boldsymbol{C}^{-1}\left(\boldsymbol{K}\boldsymbol{u}_0 + \boldsymbol{g}\left(\boldsymbol{u}_0\right) + \bar{\boldsymbol{f}}_0\right) \tag{4.180}$$

Similarily for a third degree polynomial and a second order system the initial acceleration can be found from

$$\ddot{\boldsymbol{u}}_0 = -\boldsymbol{M}^{-1}\left(\boldsymbol{C}\dot{\boldsymbol{u}}_0 + \boldsymbol{K}\boldsymbol{u}_0 + \boldsymbol{g}\left(\boldsymbol{u}_0\right) + \bar{\boldsymbol{f}}_0\right) \tag{4.181}$$

To evaluate the mean weighted load vector, $\bar{\boldsymbol{f}}$, we need to evaluate the integral in Equation (4.125). In a similar fashion to how Equation (4.124) is evaluated we can calculate

$$\bar{\boldsymbol{f}} = \theta_1\boldsymbol{f}_{n+1} + \left(1-\theta_1\right)\boldsymbol{f}_n \tag{4.182}$$

for a first order approximation or

$$\bar{\boldsymbol{f}} = \frac{\left(\theta_2+\theta_1\right)}{2}\boldsymbol{f}_{n+1} + \left(1-\theta_2\right)\boldsymbol{f}_n + \frac{\left(\theta_2-\theta_1\right)}{2}\boldsymbol{f}_{n-1} \tag{4.183}$$

for a second order approximation or

$$\begin{aligned}\bar{\boldsymbol{f}} = \frac{\left(\theta_3+3\theta_2+2\theta_1\right)}{6}\boldsymbol{f}_{n+1} &+ \left(1-\frac{\left(\theta_3+2\theta_2-\theta_1\right)}{2}\right)\boldsymbol{f}_n \\ &+ \frac{\left(\theta_3+\theta_2-2\theta_1\right)}{2}\boldsymbol{f}_{n-1} + \frac{\left(\theta_1-\theta_3\right)}{6}\boldsymbol{f}_{n-2}\end{aligned} \tag{4.184}$$

for a third order approximation.

Special SN11 case, p=1

For this special case, the mean predicited values are given by

$$\bar{\boldsymbol{u}}_{n+1} = \boldsymbol{u}_n \tag{4.185}$$

The predicted displacement values are given by

$$\hat{\boldsymbol{u}}_{n+1} = \boldsymbol{u}_n \tag{4.186}$$

The amplification matrix is given by

$$\boldsymbol{A} = \boldsymbol{C} + \theta_1 \Delta t \boldsymbol{K} \tag{4.187}$$

The right hand side vector is given by

$$\boldsymbol{b} = \boldsymbol{K} \bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}} \tag{4.188}$$

The load vector is given by

$$\bar{\boldsymbol{f}} = \theta_1 \boldsymbol{f}_{n+1} + (1 - \theta_1) \boldsymbol{f}_n \tag{4.189}$$

The nonlinear function is given by

$$\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^1\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^1 + \theta_1 \boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \Delta t \boldsymbol{\alpha}_n^1\right) + (1 - \theta_1)\, \boldsymbol{g}\left(\boldsymbol{u}_n\right) + \boldsymbol{b} = \boldsymbol{0} \tag{4.190}$$

The Jacobian matrix is given by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^1\right) = \boldsymbol{A} + \theta_1 \Delta t \frac{\partial \boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \Delta t \boldsymbol{\alpha}_n^1\right)}{\partial \boldsymbol{\alpha}_n^1} \tag{4.191}$$

And the time step update is given by

$$\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \Delta t \boldsymbol{\alpha}_n^1 \tag{4.192}$$

Special SN21 case, p=2

For this special case, the mean predicited values are given by

$$\begin{aligned} \bar{\boldsymbol{u}}_{n+1} &= \boldsymbol{u}_n + \theta_1 \Delta t \dot{\boldsymbol{u}}_n \\ \dot{\bar{\boldsymbol{u}}}_{n+1} &= \dot{\boldsymbol{u}}_n \end{aligned} \tag{4.193}$$

where

$$\dot{\boldsymbol{u}}_0 = -\boldsymbol{C}^{-1}\left(\boldsymbol{K}\boldsymbol{u}_0 + \boldsymbol{g}\left(\boldsymbol{u}_0\right) + \bar{\boldsymbol{f}}_0\right) \tag{4.194}$$

The predicted displacement values are given by

$$\hat{\boldsymbol{u}}_{n+1} = \boldsymbol{u}_n + \Delta t \dot{\boldsymbol{u}}_n \tag{4.195}$$

The amplification matrix is given by

$$\boldsymbol{A} = \theta_1 \Delta t \boldsymbol{C} + \frac{\theta_2 \Delta t^2}{2}\boldsymbol{K} \tag{4.196}$$

The right hand side vector is given by

$$\boldsymbol{b} = \boldsymbol{C}\dot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{K}\bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}} \tag{4.197}$$

The load vector is given by

$$\bar{\boldsymbol{f}} = \frac{(\theta_2 + \theta_1)}{2}\boldsymbol{f}_{n+1} + (1 - \theta_2)\,\boldsymbol{f}_n + \frac{(\theta_2 - \theta_1)}{2}\boldsymbol{f}_{n-1} \tag{4.198}$$

The nonlinear function is given by

$$\begin{aligned}\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^2\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^2 + \frac{(\theta_2 + \theta_1)}{2}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^2\right) + \\ (1 - \theta_2)\,\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \frac{(\theta_2 - \theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right) + \boldsymbol{b} = \boldsymbol{0}\end{aligned} \tag{4.199}$$

The Jacobian matrix is given by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^2\right) = \boldsymbol{A} + \frac{(\theta_2 + \theta_1)\,\Delta t^2}{4}\frac{\partial \boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^2\right)}{\partial \boldsymbol{\alpha}_n^2} \tag{4.200}$$

And the time step update is given by

$$\begin{aligned}\boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + \Delta t\dot{\boldsymbol{u}}_n + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^2 \\ \dot{\boldsymbol{u}}_{n+1} &= \dot{\boldsymbol{u}}_n + \Delta t\boldsymbol{\alpha}_n^2\end{aligned} \tag{4.201}$$

Special SN22 case, p=2

For this special case, the mean predicited values are given by

$$\begin{aligned}\bar{\boldsymbol{u}}_{n+1} &= \boldsymbol{u}_n + \theta_1\Delta t\dot{\boldsymbol{u}}_n \\ \dot{\bar{\boldsymbol{u}}}_{n+1} &= \dot{\boldsymbol{u}}_n\end{aligned} \tag{4.202}$$

The predicted displacement values are given by

$$\hat{\boldsymbol{u}}_{n+1} = \boldsymbol{u}_n + \Delta t\dot{\boldsymbol{u}}_n \tag{4.203}$$

The amplification matrix is given by

$$\boldsymbol{A} = \boldsymbol{M} + \theta_1\Delta t\boldsymbol{C} + \frac{\theta_2\Delta t^2}{2}\boldsymbol{K} \tag{4.204}$$

The right hand side vector is given by

$$\boldsymbol{b} = \boldsymbol{M}\ddot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{C}\dot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{K}\bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}} \tag{4.205}$$

The load vector is given by

$$\bar{\boldsymbol{f}} = \frac{(\theta_2+\theta_1)}{2}\boldsymbol{f}_{n+1} + (1-\theta_2)\boldsymbol{f}_n + \frac{(\theta_2-\theta_1)}{2}\boldsymbol{f}_{n-1} \tag{4.206}$$

The nonlinear function is given by

$$\begin{aligned}\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^2\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^2 + \frac{(\theta_2+\theta_1)}{2}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^2\right) + \\ (1-\theta_2)\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \frac{(\theta_2-\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right) + \boldsymbol{b} = \boldsymbol{0}\end{aligned} \tag{4.207}$$

The Jacobian matrix is given by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^2\right) = \boldsymbol{A} + \frac{(\theta_2+\theta_1)\,\Delta t^2}{4}\frac{\partial \boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^2\right)}{\partial \boldsymbol{\alpha}_n^2} \tag{4.208}$$

And the time step update is given by

$$\begin{aligned}\boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + \Delta t\dot{\boldsymbol{u}}_n + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^2\\ \dot{\boldsymbol{u}}_{n+1} &= \dot{\boldsymbol{u}}_n + \Delta t\boldsymbol{\alpha}_n^2\end{aligned} \tag{4.209}$$

Special SN32 case, p=3

For this special case, the mean predicited values are given by

$$\begin{aligned}\bar{\boldsymbol{u}}_{n+1} &= \boldsymbol{u}_n + \theta_1\Delta t\dot{\boldsymbol{u}}_n + \frac{\theta_2\Delta t^2}{2}\ddot{\boldsymbol{u}}_n\\ \dot{\bar{\boldsymbol{u}}}_{n+1} &= \dot{\boldsymbol{u}}_n + \theta_1\Delta t\ddot{\boldsymbol{u}}_n\\ \ddot{\bar{\boldsymbol{u}}}_{n+1} &= \ddot{\boldsymbol{u}}_n\end{aligned} \tag{4.210}$$

The predicted displacement values are given by

$$\hat{\boldsymbol{u}}_{n+1} = \boldsymbol{u}_n + \Delta t\dot{\boldsymbol{u}}_n + \frac{\Delta t^2}{2}\ddot{\boldsymbol{u}}_n \tag{4.211}$$

The amplification matrix is given by

$$\boldsymbol{A} = \theta_1\Delta t\boldsymbol{M} + \frac{\theta_2\Delta t^2}{2}\boldsymbol{C} + \frac{\theta_3\Delta t^3}{6}\boldsymbol{K} \tag{4.212}$$

The right hand side vector is given by

$$\boldsymbol{b} = \boldsymbol{M}\ddot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{C}\dot{\bar{\boldsymbol{u}}}_{n+1} + \boldsymbol{K}\bar{\boldsymbol{u}}_{n+1} + \bar{\boldsymbol{f}} \tag{4.213}$$

The load vector is given by

$$\bar{\boldsymbol{f}} = \frac{(\theta_3 + 3\theta_2 + 2\theta_1)}{6}\boldsymbol{f}_{n+1} + \left(1 - \frac{(\theta_3 + 2\theta_2 - \theta_1)}{2}\right)\boldsymbol{f}_n + \frac{(\theta_3 + \theta_2 - 2\theta_1)}{2}\boldsymbol{f}_{n-1} + \frac{(\theta_1 - \theta_3)}{6}\boldsymbol{f}_{n-2} \quad (4.214)$$

The nonlinear function is given by

$$\boldsymbol{\psi}\left(\boldsymbol{\alpha}_n^3\right) = \boldsymbol{A}\boldsymbol{\alpha}_n^3 + \frac{(\theta_3 + 3\theta_2 + 2\theta_1)}{6}\boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^3}{6}\boldsymbol{\alpha}_n^3\right) + \left(1 - \frac{(\theta_3 + 2\theta_2 - \theta_1)}{2}\right)\boldsymbol{g}\left(\boldsymbol{u}_n\right) + \frac{(\theta_3 + \theta_2 - 2\theta_1)}{2}\boldsymbol{g}\left(\boldsymbol{u}_{n-1}\right) + \frac{(\theta_1 - \theta_3)}{6}\boldsymbol{g}\left(\boldsymbol{u}_{n-2}\right) + \boldsymbol{b} = \boldsymbol{0} \quad (4.215)$$

The Jacobian matrix is given by

$$\boldsymbol{J}\left(\boldsymbol{\alpha}_n^3\right) = \boldsymbol{A} + \frac{(\theta_3 + 3\theta_2 + 2\theta_1)\,\Delta t^3}{36}\frac{\partial \boldsymbol{g}\left(\hat{\boldsymbol{u}}_{n+1} + \frac{\Delta t^3}{6}\boldsymbol{\alpha}_n^3\right)}{\partial \boldsymbol{\alpha}_n^3} \quad (4.216)$$

And the time step update is given by

$$\begin{aligned}
\boldsymbol{u}_{n+1} &= \boldsymbol{u}_n + \Delta t\dot{\boldsymbol{u}}_n + \frac{\Delta t^2}{2}\ddot{\boldsymbol{u}}_n + \frac{\Delta t^3}{6}\boldsymbol{\alpha}_n^3 \\
\dot{\boldsymbol{u}}_{n+1} &= \dot{\boldsymbol{u}}_n + \Delta t\ddot{\boldsymbol{u}}_n + \frac{\Delta t^2}{2}\boldsymbol{\alpha}_n^3 \\
\ddot{\boldsymbol{u}}_{n+1} &= \ddot{\boldsymbol{u}}_n + \Delta t\boldsymbol{\alpha}_n^3
\end{aligned} \quad (4.217)$$

4.3 Interface Conditions

4.3.1 Variational principles

The branch of mathematics concerned with the problem of finding a function for which a certain integral of that function is either at its largest or smallest value is called the *calculus of variations*. When scientific laws are formulated in terms of the principles of the calculus of variations they are termed *variational principles*.

4.3.2 Lagrange Multipliers

4.4 Optimisation Problems

4.4.1 Unconstrained Optimisation

The general form of a unconstrained optimisation problem is

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f\left(\boldsymbol{x}\right) \\ \text{subject to} \quad & \boldsymbol{a} \leq \boldsymbol{x} \leq \boldsymbol{b} \end{aligned} \tag{4.218}$$

4.4.2 Linear Program

The general form of a linear programming problem is

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned} \tag{4.219}$$

4.4.3 Constrained Optimisation

The general form of a PDE constrained optimisation problem is

$$\begin{aligned} \min_{\boldsymbol{x}} \quad & f\left(\boldsymbol{x}\right) \\ \text{subject to} \quad & \boldsymbol{a} \leq \boldsymbol{x} \leq \boldsymbol{b} \\ & \boldsymbol{g}\left(\boldsymbol{x}\right) = \boldsymbol{c} \\ & \boldsymbol{h}\left(\boldsymbol{x}\right) \geq \boldsymbol{d} \end{aligned} \tag{4.220}$$

4.4.4 PDE Constrained Optimisation

The general form of a PDE constrained optimisation problem is

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{u}} \quad & f\left(\boldsymbol{x},\boldsymbol{u}\right) \\ \text{subject to} \quad & \boldsymbol{p}\left(\boldsymbol{x},\boldsymbol{u}\right)=\boldsymbol{0} \\ & \boldsymbol{a}\leq\boldsymbol{x}\leq\boldsymbol{b} \\ & \boldsymbol{g}\left(\boldsymbol{x}\right)=\boldsymbol{c} \\ & \boldsymbol{h}\left(\boldsymbol{x}\right)\geq\boldsymbol{d} \end{aligned} \tag{4.221}$$

Chapter 5

Mathematics

5.1 Notation

5.1.1 Vector and Tensors

The following formatting will be used for vectors and tensors.

$$\begin{array}{rl}
a, b, c, \ldots & \text{Scalars or } 0^{th} \text{ order tensors} \\
\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}, \ldots & \text{Vectors or } 1^{st} \text{ order tensors} \\
\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \ldots & \text{Dyadics or } 2^{nd} \text{ order tensors} \\
\mathrm{A}, \mathrm{B}, \mathrm{C}, \ldots & \text{Triadics or } 3^{rd} \text{ order tensors} \\
\boldsymbol{\mathsf{A}}, \boldsymbol{\mathsf{B}}, \boldsymbol{\mathsf{C}}, \ldots & \text{Tetradics or } 4^{th} \text{ order tensors}
\end{array} \tag{5.1}$$

5.2 Vector Calculus

5.2.1 Vector Identities

If $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{d}$ are all vectors and ϕ and ψ are scalars then

$$\begin{aligned}
\boldsymbol{a}\cdot(\boldsymbol{b}\times\boldsymbol{c}) &= \boldsymbol{b}\cdot(\boldsymbol{c}\times\boldsymbol{a}) = \boldsymbol{c}\cdot(\boldsymbol{a}\times\boldsymbol{b})\\
\boldsymbol{a}\times(\boldsymbol{b}\times\boldsymbol{c}) &= (\boldsymbol{a}\cdot\boldsymbol{c})\,\boldsymbol{b} - (\boldsymbol{a}\cdot\boldsymbol{b})\,\boldsymbol{c}\\
(\boldsymbol{a}\times\boldsymbol{b})\cdot(\boldsymbol{c}\times\boldsymbol{d}) &= (\boldsymbol{a}\cdot\boldsymbol{c})\,(\boldsymbol{b}\cdot\boldsymbol{d}) - (\boldsymbol{a}\cdot\boldsymbol{d})\,(\boldsymbol{b}\cdot\boldsymbol{c})\\
\nabla(\phi\psi) &= \phi\nabla\psi + \psi\nabla\phi\\
\nabla\times\nabla\phi &= \boldsymbol{0}\\
\nabla\cdot\nabla\phi &= \nabla^2\phi\\
\nabla\cdot(\nabla\times\boldsymbol{a}) &= 0\\
\nabla\times(\nabla\times\boldsymbol{a}) &= \nabla(\nabla\cdot a) - \nabla^2\boldsymbol{a}\\
\nabla\cdot(\phi\boldsymbol{a}) &= \boldsymbol{a}\cdot\nabla\phi + \phi\nabla\cdot\boldsymbol{a}\\
\nabla\times\phi\boldsymbol{a} &= \nabla\phi\times\boldsymbol{a} + \phi\nabla\times\boldsymbol{a}\\
\nabla(\boldsymbol{a}\cdot\boldsymbol{b}) &= (\boldsymbol{a}\cdot\nabla)\,\boldsymbol{b} + (\boldsymbol{b}\cdot\nabla)\,\boldsymbol{a} + \boldsymbol{a}\times(\nabla\times\boldsymbol{b}) + \boldsymbol{b}\times(\nabla\times\boldsymbol{a})\\
\nabla\cdot(\boldsymbol{a}\times\boldsymbol{b}) &= \boldsymbol{b}\cdot(\nabla\times\boldsymbol{a}) - \boldsymbol{a}\cdot(\nabla\times\boldsymbol{b})\\
\nabla\times(\boldsymbol{a}\times\boldsymbol{b}) &= \boldsymbol{a}\,(\nabla\cdot\boldsymbol{b}) - \boldsymbol{b}\,(\nabla\cdot\boldsymbol{a}) + (\boldsymbol{b}\cdot\nabla)\,\boldsymbol{a} - (\boldsymbol{a}\cdot\nabla)\,\boldsymbol{b}
\end{aligned} \tag{5.2}$$

If $\boldsymbol{A}$ is a second order tensor, $\boldsymbol{b}$ is a vector and ϕ is a scalar then

$$\begin{aligned}
\nabla\cdot(\phi\boldsymbol{A}) &= \nabla\phi\cdot\boldsymbol{A} + \phi\nabla\cdot\boldsymbol{A}\\
\nabla\cdot(\boldsymbol{A}\cdot\boldsymbol{b}) &= (\nabla\cdot\boldsymbol{A})\cdot\boldsymbol{b} + \boldsymbol{A}^T : \nabla\boldsymbol{b}
\end{aligned} \tag{5.3}$$

5.2.2 Vector Integral Theorems

For a domain Ω with boundary $\Gamma = \partial\Omega$ and normal $\boldsymbol{n}$ and a vector field $\boldsymbol{a}$ then *the divergence theorem* is

$$\int_\Omega \nabla\cdot\boldsymbol{a}\,\mathrm{d}\Omega = \int_\Gamma \boldsymbol{a}\cdot\boldsymbol{n}\,\mathrm{d}\Gamma \tag{5.4}$$

5.3 Operations

5.3.1 Linearisation

A linearisation of a function $f(\boldsymbol{x})$ in the direction of $\Delta\boldsymbol{x}$ is

$$L_{\boldsymbol{x}}(f)[\Delta\boldsymbol{x}] = \frac{d}{d\epsilon} f(\boldsymbol{x} + \epsilon\Delta\boldsymbol{x})\big|_{\epsilon=0} = f(\boldsymbol{x}) + D(f(\boldsymbol{x}))[\Delta\boldsymbol{x}] \tag{5.5}$$

x

Chapter 6

Equation Sets

6.1 Classical Field Class

6.1.1 Generalised Laplace Equation

Governing equations:

The generalised Laplace's equation on a domain Ω with boundary Γ in OpenCMISS can be stated as

$$\boxed{\nabla \cdot \left(\boldsymbol{\sigma}\left(\boldsymbol{x}\right) \nabla u\left(\boldsymbol{x}\right)\right) = 0} \tag{6.1}$$

where $\boldsymbol{x} \in \Omega$, $u\left(\boldsymbol{x}\right)$ is the potential and $\boldsymbol{\sigma}\left(\boldsymbol{x}\right)$ is the (positive definite) conductivity tensor throughout the domain. Note that $\boldsymbol{\sigma} = \boldsymbol{I}$ gives the standard form of Laplace's equation *i.e.,* $\nabla^2 u = 0$.

To complete the description of the boundary value problem, the governing Equation (6.1) is supplemented by suitable boundary conditions on the domain boundary Γ. These boundary conditions can either be of Dirichlet type and specify a solution value, d *i.e.,*

$$u\left(\boldsymbol{x}\right) = d\left(\boldsymbol{x}\right) \quad \boldsymbol{x} \in \Gamma_D, \tag{6.2}$$

and/or of Neumann type and specify the solution gradient in normal direction, e *i.e.,*

$$q\left(\boldsymbol{x}, t\right) = \left(\boldsymbol{\sigma}\left(\boldsymbol{x}\right) \nabla u\left(\boldsymbol{x}, t\right)\right) \cdot \boldsymbol{n} = e\left(\boldsymbol{x}, t\right) \quad \boldsymbol{x} \in \Gamma_N, \tag{6.3}$$

where $q\left(\boldsymbol{x},t\right)$ is the flux in the normal direction, $\boldsymbol{n}$ is the normal vector to the boudary and $\Gamma = \Gamma_D \cup \Gamma_N$.

Weak formulation:

The corresponding weak form of Equation (6.1) is

$$\int\limits_{\Omega} \nabla \cdot \left(\boldsymbol{\sigma} \nabla \phi\right) w \, \mathbf{d}\Omega = 0 \tag{6.4}$$

where w is a suitable test function (For a definition of what constitutes a “suitable” test and trial function, see Section **??** - still has to be written).

Applying Green’s theorem to Equation (6.4) gives

$$\int\limits_{\Omega} \nabla \cdot \left(\boldsymbol{\sigma} \nabla \phi\right) w \, \mathbf{d}\Omega = - \int\limits_{\Omega} \left(\boldsymbol{\sigma} \nabla \phi\right) \cdot \nabla w \, \mathbf{d}\Omega + \int\limits_{\Gamma} \left(\boldsymbol{\sigma} \nabla \phi\right) \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma = 0 \tag{6.5}$$

or

$$\int\limits_{\Omega} \left(\boldsymbol{\sigma} \nabla \phi\right) \cdot \nabla w \, \mathbf{d}\Omega = \int\limits_{\Gamma} \left(\boldsymbol{\sigma} \nabla \phi\right) \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma \, , \tag{6.6}$$

Tensor notation:

If we now consider the integrand of the left hand side of Equation (6.6) in tensorial form with covariant derivatives indicated by a semi-colon in the index (see Section 4.1.4 for details) then

$$\boldsymbol{\sigma} \nabla u = \sigma^{i}_{.j} u_{;i} \tag{6.7}$$

and

$$\nabla w = w_{;k} \tag{6.8}$$

Now, Equations (6.7) and (6.8) represent vectors that are covariant and therefore we must convert Equation (6.7) to a contravariant vector by multiplying by the contravariant metric tensor (from $\boldsymbol{i}$ to $\boldsymbol{x}$ coordinates; see Sections 4.1.2 and 4.1.6) so that we can take the dot product.

The final tensorial form for the left hand integral is

$$(\boldsymbol{\sigma}\nabla u)\cdot\nabla w = G^{jk}\sigma^{i}_{.j}\phi_{;i}w_{;k} \tag{6.9}$$

and thus Equation (6.6) becomes

$$\int\limits_{\Omega} G^{jk}\sigma^{i}_{.j}u_{;i}w_{;k}\,\mathbf{d}\Omega = \int\limits_{\Gamma} G^{jk}\sigma^{i}_{.j}u_{;i}n_{k}w\,\mathbf{d}\Gamma \tag{6.10}$$

or

$$\int\limits_{\Omega} G^{jk}\sigma^{i}_{.j}u_{;i}w_{;k}\,\mathbf{d}\Omega = \int\limits_{\Gamma} qw\,\mathbf{d}\Gamma \tag{6.11}$$

Finite element formulation:

We can now discretise the domain into finite elements *i.e.,* $\Omega = \bigcup\limits_{e=1}^{E}\Omega_e$ with $\Gamma = \bigcup\limits_{f=1}^{F}\Gamma_f$, Equation (6.11) becomes

$$\sum_{e=1}^{E}\int\limits_{\Omega_e} G^{jk}\sigma^{i}_{j}u_{;i}w_{;k}\,\mathbf{d}\Omega = \sum_{f=1}^{F}\int\limits_{\Gamma_f} qw\,\mathbf{d}\Omega \tag{6.12}$$

The next step is to approximate the dependent variable, u, using basis functions. To simplify this for different types of basis functions an *interpolation notation* is adopted. This interpolation is such that $\psi^{\beta}_{n}(\boldsymbol{\xi})$ are the appropriate basis functions for the type of element (*e.g.,* bicubic Hermite, Hermite-sector, *etc.*) and dimension of the problem (one, two or three-dimensional). For example if $\boldsymbol{\xi} = \{\xi\}$ and the element is a cubic Hermite element (Section 2.3.2) then $\psi^{\beta}_{n}(\boldsymbol{\xi})$ are the cubic Hermite basis functions where the local node number n ranges from 1 to 2 and the derivative β ranges from 0 to 1. If, however, $\boldsymbol{\xi} = \{\xi_1, \xi_2\}$ and the element is a bicubic Hermite element then n ranges from 1 to 4, β ranges from 0 to 3 and $\psi^{\beta}_{n}(\boldsymbol{\xi})$ is the appropriate basis function for the nodal variable $u^{n}_{,\beta}$. The nodal variables are defined as follows: $u^{n}_{,0} = u^{n}$, $u^{n}_{,1} = \dfrac{\partial u^{n}}{\partial s_1}$, $u^{n}_{,2} = \dfrac{\partial u^{n}}{\partial s_2}$, $u^{n}_{,3} = \dfrac{\partial^2 u^{n}}{\partial s_1 \partial s_2}$, *etc.* Hence for the bicubic Hermite element the interpolation function $\psi^{3}_{2}(\boldsymbol{\xi})$ multiplies the nodal variable $u^{2}_{,3} = \dfrac{\partial^2 u^{2}}{\partial s_1 \partial s_2}$ and thus the interpolation function is $\Psi^{1}_{2}(\xi_1)\,\Psi^{1}_{1}(\xi_2)$. The scale factors for the Hermite interpolation are handled by the introduction of an interpolation scale factor $\mathrm{S}(n,\beta)$ defined as follows: $\mathrm{S}(n,0) = 1$, $\mathrm{S}(n,1) = \mathcal{S}(n,1)$, $\mathrm{S}(n,2) = \mathcal{S}(n,2)$, $\mathrm{S}(n,3) = \mathcal{S}(n,1)\,\mathcal{S}(n,2)$, *etc.* For Lagrangian basis functions the interpolation scale factors are all one. The general form of the interpolation notation for u is then

$$u(\boldsymbol{\xi}) = \psi_n^{\beta}(\boldsymbol{\xi})\, u^n_{,\beta}\mathrm{S}(n,\beta) \tag{6.13}$$

We can also interpolate the other variables in a similar manner *i.e.,*

$$\begin{aligned} q(\boldsymbol{\xi}) &= \psi_o^{\gamma}\left(\boldsymbol{\xi} q^o_{,\gamma}\right)\mathrm{S}(o,\gamma) \\ \boldsymbol{\sigma}(\boldsymbol{\xi}) &= \psi_p^{\delta}(\boldsymbol{\xi})\,\boldsymbol{\sigma}^p_{,\delta}\mathrm{S}(p,\delta) \end{aligned} \tag{6.14}$$

where $q^o_{,\gamma}$ and $\boldsymbol{\sigma}^p_{,\delta}$ are the nodal degrees-of-freedom for the flux variable and conductivity tensor.

Using a Galerkin finite element approach (where the weighting functions are chosento be the interpolating basis functions) we have

$$w(\boldsymbol{\xi}) = \psi_m^{\alpha}(\boldsymbol{\xi})\,\mathrm{S}(m,\alpha) \tag{6.15}$$

Spatial integration:

When dealing with integrals a similar interpolation notation is adopted in that $d\boldsymbol{\xi}$ is the appropriate infinitesimal for the dimension of the problem. The limits of the integral are also written in bold font and indicate the appropriate number of integrals for the dimension of the problem. For example if $\boldsymbol{\xi} = \{\xi_1, \xi_2\}$ then $\int\limits_{\mathbf{0}}^{\mathbf{1}} x\,\mathbf{d}\boldsymbol{\xi} = \int\limits_0^1\int\limits_0^1 x\,\mathbf{d}\xi_1 \wedge \mathbf{d}\xi_2$, but if $\boldsymbol{\xi} = \{\xi_1, \xi_2, \xi_3\}$ then $\int\limits_{\mathbf{0}}^{\mathbf{1}} x\,\mathbf{d}\boldsymbol{\xi} = \int\limits_0^1\int\limits_0^1\int\limits_0^1 x\,d\xi_1 d\xi_2 d\xi_3$.

In order to integrate in $\boldsymbol{\xi}$ coordinates we must convert the spatial derivatives to derivatives with respect to $\boldsymbol{\xi}$. Using the tensor transformation equations for a covariant vector we have

$$u_{;i} = \frac{\partial \xi^s}{\partial x^i} u_{;s} = \frac{\partial \xi^s}{\partial x^i}\frac{\partial u}{\partial \xi^s} \tag{6.16}$$

and

$$w_{;k} = \frac{\partial \xi^r}{\partial x^k} w_{;r} = \frac{\partial \xi^r}{\partial x^k}\frac{\partial w}{\partial \xi^r} \tag{6.17}$$

As we only know $\tilde{\boldsymbol{\sigma}}$, the conductivity in the $\boldsymbol{\nu}$ (fibre) coordinate system rather than $\boldsymbol{\sigma}$, the conductivity in the $\boldsymbol{x}$ (geometric) coordinate system, we must transform the mixed tensor $\tilde{\sigma}^a_{.b}$ from $\boldsymbol{\nu}$ to $\boldsymbol{x}$ coordinates. However, as the fibre coordinate system is defined in terms of $\boldsymbol{\xi}$ coordinates we first transform the conductivity tensor from $\boldsymbol{\nu}$ to $\boldsymbol{\xi}$ coordinates *i.e.,*

$$\sigma^d_{.e}\left(\boldsymbol{\xi}\right) = \frac{\partial \xi^d}{\partial \nu^a}\frac{\partial \nu^b}{\partial \xi^e}\tilde{\sigma}^a_{.b}\left(\boldsymbol{\xi}\right) \tag{6.18}$$

with $.b$ indicating that b is the "second" index (Section **??**), and then transform the conductivity from $\boldsymbol{\xi}$ coordinates to $\boldsymbol{x}$ coordinates *i.e.,*

$$\begin{aligned}\sigma^i_{.j}\left(\boldsymbol{\xi}\right) &= \frac{\partial x^i}{\partial \xi^d}\frac{\partial \xi^e}{\partial x^j}\sigma^d_{.e}\left(\boldsymbol{\xi}\right)\\ &= \frac{\partial x^i}{\partial \xi^d}\frac{\partial \xi^e}{\partial x^j}\frac{\partial \xi^d}{\partial \nu^a}\frac{\partial \nu^b}{\partial \xi^e}\tilde{\sigma}^a_{.b}\left(\boldsymbol{\xi}\right)\end{aligned} \tag{6.19}$$

where $\tilde{\boldsymbol{\sigma}}$ is interpolated in the normal way *i.e.,*

$$\tilde{\boldsymbol{\sigma}}\left(\boldsymbol{\xi}\right) = \psi^\delta_p\left(\boldsymbol{\xi}\right)\tilde{\boldsymbol{\sigma}}^p_{,\delta}\mathrm{S}\left(p,\delta\right) \tag{6.20}$$

Using an interpolated variables yields

$$\begin{aligned}&\sum_{e=1}^{E}\int_{\mathbf{0}}^{\mathbf{1}} G^{jk}\frac{\partial x^i}{\partial \xi^d}\frac{\partial \xi^e}{\partial x^j}\frac{\partial \xi^d}{\partial \nu^a}\frac{\partial \nu^b}{\partial \xi^e}\tilde{\sigma}^a_{.b}\left(\boldsymbol{\xi}\right)\frac{\partial \xi^s}{\partial x^i}\\ &\qquad\frac{\partial\left(\psi^\beta_n\left(\boldsymbol{\xi}\right)u^n_{,\beta}\mathrm{S}\left(n,\beta\right)\right)}{\partial \xi^s}\frac{\partial \xi^r}{\partial x^k}\frac{\partial\left(\psi^\alpha_m\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\right)}{\partial \xi^r}\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}\\ &\qquad= \sum_{f=1}^{F}\int_{\Gamma_f}\psi^\gamma_o\left(\boldsymbol{\xi}\right)q^o_{,\gamma}\mathrm{S}\left(o,\gamma\right)\psi^\alpha_m\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\,\mathbf{d}\Gamma\end{aligned} \tag{6.21}$$

where $\boldsymbol{J}\left(\boldsymbol{\xi}\right)$ is the *Jacobian* of the transformation from the integration $\boldsymbol{x}$ to $\boldsymbol{\xi}$ coordinates.

Taking the fixed nodal degrees-of-freedom and scale factors outside the integral we have

$$\sum_{e=1}^{E} u^n_{,\beta} \mathrm{S}(m,\alpha)\,\mathrm{S}(n,\beta) \int_{\mathbf{0}}^{\mathbf{1}} G^{jk} \frac{\partial x^i}{\partial \xi^d} \frac{\partial \xi^e}{\partial x^j} \frac{\partial \xi^d}{\partial \nu^a} \frac{\partial \nu^b}{\partial \xi^e} \tilde{\sigma}^a_{.b}(\boldsymbol{\xi}) \frac{\partial \xi^r}{\partial x^i} \frac{\partial \xi^s}{\partial x^k} \frac{\partial \psi^\alpha_m(\boldsymbol{\xi})}{\partial \xi^s} \frac{\partial \psi^\beta_n(\boldsymbol{\xi})}{\partial \xi^r} \left|\boldsymbol{J}(\boldsymbol{\xi})\right| d\boldsymbol{\xi} = \sum_{f=1}^{F} q^o_{,\gamma} \mathrm{S}(m,\alpha)\,\mathrm{S}(o,\gamma) \int_{\Gamma_f} \psi^\gamma_o(\boldsymbol{\xi})\,\psi^\alpha_m(\boldsymbol{\xi})\,\mathbf{d}\Gamma \quad (6.22)$$

This is an equation of the form of

$$\boldsymbol{K}\boldsymbol{u} = \boldsymbol{f} \quad (6.23)$$

where

$$\boldsymbol{f} = \boldsymbol{N}\boldsymbol{q} \quad (6.24)$$

The elemental stiffness matrix, $K^{\alpha\beta}_{mn}$, is given by

$$K^{\alpha\beta}_{mn} = \mathrm{S}(m,\alpha)\,\mathrm{S}(m,\beta) \int_{\mathbf{0}}^{\mathbf{1}} \frac{\partial \psi^\alpha_m(\boldsymbol{\xi})}{\partial \xi^r} \frac{\partial \psi^\beta_n(\boldsymbol{\xi})}{\partial \xi^s} \gamma^{rs}(\boldsymbol{\xi}) \left|\boldsymbol{J}(\boldsymbol{\xi})\right| \mathbf{d}\boldsymbol{\xi} \quad (6.25)$$

where

$$\gamma^{rs}(\boldsymbol{\xi}) = G^{jk} \frac{\partial x^i}{\partial \xi^d} \frac{\partial \xi^e}{\partial x^j} \frac{\partial \xi^d}{\partial \nu^a} \frac{\partial \nu^b}{\partial \xi^e} \tilde{\sigma}^a_{.b}(\boldsymbol{\xi}) \frac{\partial \xi^r}{\partial x^i} \frac{\partial \xi^s}{\partial x^k} \quad (6.26)$$

Note that for Laplace's equation with no conductivity or fibre fields we have

$$\gamma^{rs}(\boldsymbol{\xi}) = G^{ik} \frac{\partial \xi^r}{\partial x^i} \frac{\partial \xi^s}{\partial x^k} \quad (6.27)$$

and that for rectangular-Cartesian coordinates systems $G^{ik} = \delta^{ik}$ and thus $i = k$. This now gives

$$\gamma^{rs}(\boldsymbol{\xi}) = \frac{\partial \xi^r}{\partial x^i} \frac{\partial \xi^s}{\partial x^i} = g^{rs} \quad (6.28)$$

where g^{rs} is the *contravariant metric tensor* from $\boldsymbol{x}$ to $\boldsymbol{\xi}$ coordinates. It may thus be helpful to think about γ^{rs} as a scaled/transformed contravariant metric tensor.

The right hand side flux matrix, $N_{mo}^{\alpha\gamma}$, is given by

$$N_{mo}^{\alpha\gamma} = \mathrm{S}(m,\alpha)\,\mathrm{S}(o,\gamma)\int\limits_{\Gamma_f} \psi_m^{\alpha}(\boldsymbol{\xi})\,\psi_o^{\gamma}(\boldsymbol{\xi})\ \mathbf{d}\Gamma \tag{6.29}$$

6.1.2 Poisson Equations

Generalised Poisson Equation

Governing equations:

The general form of Poisson's equation with source on a domain Ω with boundary Γ in OpenCMISS can be stated as

$$\boxed{\nabla\cdot\left(\boldsymbol{\sigma}(\boldsymbol{x})\,\nabla u(\boldsymbol{x})\right) + a(\boldsymbol{x}) = 0} \tag{6.30}$$

where $\boldsymbol{x}\in\Omega$, $u(\boldsymbol{x})$ is the dependent variable, $\boldsymbol{\sigma}(\boldsymbol{x})$ is the conductivity tensor throughout the domain and $a(\boldsymbol{x})$ is a source term.

Appropriate boundary conditions conditions for the diffusion equation are specification of Dirichlet boundary conditions on the solution, d *i.e.,*

$$u(\boldsymbol{x},t) = d(\boldsymbol{x},t) \quad \boldsymbol{x}\in\Gamma_D, \tag{6.31}$$

and/or Neumann conditions in terms of the solution flux in the normal direction, e *i.e.,*

$$q(\boldsymbol{x}) = \left(\boldsymbol{\sigma}(\boldsymbol{x})\,\nabla u(\boldsymbol{x})\right)\cdot\boldsymbol{n} = e(\boldsymbol{x}) \quad \boldsymbol{x}\in\Gamma_N, \tag{6.32}$$

where $q(\boldsymbol{x},t)$ is the flux in the normal direction, $\boldsymbol{n}$ is the normal vector to the boudary and $\Gamma=\Gamma_D\cup\Gamma_N$.

Weak formulation:

The corresponding weak form of Equation (6.30) can be found by integrating over the spatial domain with a test function *i.e.,*

$$\int\limits_{\Omega} \left(\nabla \cdot (\boldsymbol{\sigma} \nabla u) + a\right) w \,\mathbf{d}\Omega = 0 \tag{6.33}$$

where w is a suitable spatial test function.

Applying the divergence theorem in Equation (6.398) to Equation (6.33) gives

$$-\int\limits_{\Omega} \boldsymbol{\sigma} \nabla u \cdot \nabla w \,\mathbf{d}\Omega + \int\limits_{\Gamma} \left(\boldsymbol{\sigma} \nabla u \cdot \boldsymbol{n}\right) w \,\mathbf{d}\Gamma + \int\limits_{\Omega} a w \,\mathbf{d}\Omega = 0 \tag{6.34}$$

where $\boldsymbol{n}$ is the unit outward normal vector to the boundary Γ.

Tensor notation:

Equation (6.34) can be written in tensor notation as

$$-\int\limits_{\Omega} G^{jk} \sigma^i_j u_{;i} w_{;k} \,\mathbf{d}\Omega + \int\limits_{\Gamma} G^{jk} \sigma^i_j u_{;i} n_k w \,\mathbf{d}\Gamma + \int\limits_{\Omega} a w \,\mathbf{d}\Omega = 0 \tag{6.35}$$

or

$$\int\limits_{\Omega} G^{jk} \sigma^i_j u_{;i} w_{;k} \,\mathbf{d}\Omega = \int\limits_{\Gamma} q w \,\mathbf{d}\Gamma + \int\limits_{\Omega} a w \,\mathbf{d}\Omega = 0 \tag{6.36}$$

where G^{jk} is the contravariant metric tensor for the spatial coordinate system.

Finite element formulation:

We can now discretise the spatial domain into finite elements *i.e.,* $\Omega = \bigcup_{e=1}^{E} \Omega_e$ with $\Gamma = \bigcup_{f=1}^{F} \Gamma_f$. Equation (6.36) becomes

$$\sum_{e=1}^{E} \int_{\Omega_e} G^{jk} \sigma_j^i u_{;i} w_{;k} \, \mathbf{d}\Omega = \sum_{f=1}^{F} \int_{\Gamma_f} q w \, \mathbf{d}\Gamma + \sum_{e=1}^{E} \int_{\Omega_e} a w \, \mathbf{d}\Omega \tag{6.37}$$

We can now interpolate the variables with basis functions *i.e.,*

$$\begin{aligned} u(\boldsymbol{\xi}) &= \psi_n^{\beta}(\boldsymbol{\xi})\, u_{,\beta}^{n} \mathrm{S}(n, \beta) \\ q(\boldsymbol{\xi}) &= \psi_o^{\gamma}(\boldsymbol{\xi})\, q_{,\gamma}^{o} \mathrm{S}(o, \gamma) \\ \tilde{\boldsymbol{\sigma}}(\boldsymbol{\xi}) &= \psi_p^{\delta}(\boldsymbol{\xi})\, \tilde{\boldsymbol{\sigma}}_{,\delta}^{p} \mathrm{S}(p, \delta) \\ a(\boldsymbol{\xi}) &= \psi_p^{\delta}(\boldsymbol{\xi})\, a_{,\delta}^{p} \mathrm{S}(p, \delta) \end{aligned} \tag{6.38}$$

where $u_{,\beta}^{n}$, $q_{,\gamma}^{o}$, $\tilde{\boldsymbol{\sigma}}_{,\delta}^{p}$ and $a_{,\delta}^{p}$ are the nodal degrees-of-freedom for the variables.

For a Galerkin finite element formulation we also choose the spatial weighting function w to be equal to the basis fucntions *i.e.,*

$$w(\boldsymbol{\xi}) = \psi_m^{\alpha}(\boldsymbol{\xi})\, \mathrm{S}(m, \alpha) \tag{6.39}$$

Spatial integration:

Adopting the standard integration notation we can write the spatial integration term in Equation (6.37) as

$$\sum_{e=1}^{E}\int_{\mathbf{0}}^{\mathbf{1}} G^{jk}\frac{\partial x^i}{\partial \xi^d}\frac{\partial \xi^e}{\partial x^j}\frac{\partial \xi^d}{\partial \nu^a}\frac{\partial \nu^b}{\partial \xi^e}\tilde{\sigma}^{a}_{.b}\left(\boldsymbol{\xi}\right)\frac{\partial \xi^s}{\partial x^i}$$
$$\frac{\partial\left(\psi_n^{\beta}\left(\boldsymbol{\xi}\right)u^n_{,\beta}\mathrm{S}\left(n,\beta\right)\right)}{\partial \xi^s}\frac{\partial \xi^r}{\partial x^k}\frac{\partial\left(\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\right)}{\partial \xi^r}\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}$$
$$=\sum_{f=1}^{F}\int_{\Gamma_f}\psi_o^{\gamma}\left(\boldsymbol{\xi}\right)q^o_{,\gamma}\mathrm{S}\left(o,\gamma\right)\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\,\mathbf{d}\Gamma+$$
$$\sum_{e=1}^{E}\int_{\mathbf{0}}^{\mathbf{1}} a\left(\boldsymbol{\xi}\right)\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \tag{6.40}$$

where $\boldsymbol{J}\left(\boldsymbol{\xi}\right)$ is the *Jacobian* of the transformation from the integration $\boldsymbol{x}$ to $\boldsymbol{\xi}$ coordinates.

Taking values that are constant over the integration interval outside the integration gives

$$\sum_{e=1}^{E}u^n_{,\beta}\mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(n,\beta\right)\int_{\mathbf{0}}^{\mathbf{1}}\frac{\partial\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)}{\partial \xi^r}\frac{\partial\psi_n^{\beta}\left(\boldsymbol{\xi}\right)}{\partial \xi^s}\gamma^{rs}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}$$
$$=\sum_{f=1}^{F}q^o_{,\gamma}\mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(o,\gamma\right)\int_{\Gamma_f}\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\psi_o^{\gamma}\left(\boldsymbol{\xi}\right)\,\mathbf{d}\Gamma+$$
$$\sum_{e=1}^{E}a^p_{,\delta}\mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(p,\delta\right)\int_{\mathbf{0}}^{\mathbf{1}}\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\psi_p^{\delta}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \tag{6.41}$$

where $\gamma^{rs}\left(\boldsymbol{\xi}\right)$ is defined in Equations (6.26)–(6.28).

This is an equation of the form

$$\boldsymbol{K}\boldsymbol{u}=\boldsymbol{f} \tag{6.42}$$

where

$$\boldsymbol{f}=\boldsymbol{N}\boldsymbol{q}+\boldsymbol{R}\boldsymbol{a} \tag{6.43}$$

The elemental stiffness matrix, $K_{mn}^{\alpha\beta}$, is given by

$$K_{mn}^{\alpha\beta} = \mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(m,\beta\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \frac{\partial\psi_m^{\alpha}\left(\boldsymbol{\xi}\right)}{\partial\xi^r}\frac{\partial\psi_n^{\beta}\left(\boldsymbol{\xi}\right)}{\partial\xi^s}\gamma^{rs}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \tag{6.44}$$

The elemental flux matrix, $N_{mo}^{\alpha\gamma}$, is given by

$$N_{mo}^{\alpha\gamma} = \mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(o,\gamma\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\psi_o^{\gamma}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \tag{6.45}$$

and the elemental source matrix, $R_{mp}^{\alpha\delta}$,

$$R_{mp}^{\alpha\delta} = \mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(p,\delta\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \psi_m^{\alpha}\left(\boldsymbol{\xi}\right)\psi_p^{\delta}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \tag{6.46}$$

6.1.3 Diffusion Equations

6.1.4 General Diffusion Equation

Governing equations:

The general form of the diffusion equation with source on a domain Ω with boundary Γ in OpenCMISS can be stated as

$$\boxed{a\left(\boldsymbol{x}\right)\frac{\partial u\left(\boldsymbol{x},t\right)}{\partial t} + \nabla\cdot\left(\boldsymbol{\sigma}\left(\boldsymbol{x}\right)\nabla u\left(\boldsymbol{x},t\right)\right) + b\left(\boldsymbol{x},t\right) = 0} \tag{6.47}$$

where $\boldsymbol{x}\in\Omega$, $u\left(\boldsymbol{x},t\right)$ is the quatity that diffuses, $a\left(\boldsymbol{x}\right)$ is a temporal weighting, $\boldsymbol{\sigma}\left(\boldsymbol{x}\right)$ is the conductivity tensor throughout the domain and $b\left(\boldsymbol{x},t\right)$ is a source term.

Appropriate boundary conditions conditions for the diffusion equation are specification of Dirichlet boundary conditions on the solution, d *i.e.,*

$$u\left(\boldsymbol{x},t\right) = d\left(\boldsymbol{x},t\right) \quad \boldsymbol{x}\in\Gamma_D, \tag{6.48}$$

and/or Neumann conditions in terms of the solution flux in the normal direction, e *i.e.,*

$$q\left(\boldsymbol{x}\right) = \left(\boldsymbol{\sigma}\left(\boldsymbol{x}\right)\nabla u\left(\boldsymbol{x}\right)\right)\cdot\boldsymbol{n} = e\left(\boldsymbol{x}\right) \quad \boldsymbol{x}\in\Gamma_N, \tag{6.49}$$

where $q\left(\boldsymbol{x},t\right)$ is the flux in the normal direction, $\boldsymbol{n}$ is the normal vector to the boudary and $\Gamma = \Gamma_D\cup\Gamma_N$.

Appropriate initial conditions for the diffusion equation are the specification of an initial value of the solution, f *i.e.,*

$$u\left(\boldsymbol{x},0\right) = f\left(\boldsymbol{x}\right) \quad \boldsymbol{x}\in\Omega. \tag{6.50}$$

Weak formulation:

The corresponding weak form of Equation (6.47) can be found by integrating over the spatial domain with a test function *i.e.,*

$$\int\limits_{\Omega} a\frac{\partial u}{\partial t} + \left(\nabla\cdot\left(\boldsymbol{\sigma}\nabla u\right) + a\right) w\,\mathbf{d}\Omega = 0 \tag{6.51}$$

where w is a suitable spatial test function.

Applying the divergence theorem in Equation (6.398) to Equation (6.51) gives

$$\int\limits_{\Omega}\left(a\frac{\partial u}{\partial t}\right) w\,\mathbf{d}\Omega - \int\limits_{\Omega}\boldsymbol{\sigma}\nabla u\cdot\nabla w\,\mathbf{d}\Omega + \int\limits_{\Gamma}\left(\boldsymbol{\sigma}\nabla u\cdot\boldsymbol{n}\right) w\,\mathbf{d}\Gamma + \int\limits_{\Omega} bw\,\mathbf{d}\Omega = 0 \tag{6.52}$$

where $\boldsymbol{n}$ is the unit outward normal vector to the boundary Γ.

Tensor notation:

Equation (6.52) can be written in tensor notation as

$$\int\limits_{\Omega} a\dot{u}w\,\mathbf{d}\Omega - \int\limits_{\Omega} G^{jk}\sigma^i_j u_{;i} w_{;k}\,\mathbf{d}\Omega + \int\limits_{\Gamma} G^{jk}\sigma^i_j u_{;i} n_k w\,\mathbf{d}\Gamma + \int\limits_{\Omega} bw\,\mathbf{d}\Omega = 0 \tag{6.53}$$

or

$$\int\limits_{\Omega} a\dot{u}w\,\mathbf{d}\Omega - \int\limits_{\Omega} G^{jk}\sigma^i_j u_{;i} w_{;k}\,\mathbf{d}\Omega + \int\limits_{\Gamma} qw\,\mathbf{d}\Gamma + \int\limits_{\Omega} bw\,\mathbf{d}\Omega = 0 \tag{6.54}$$

where G^{jk} is the contravariant metric tensor for the spatial coordinate system.

Finite element formulation:

We can now discretise the spatial domain into finite elements *i.e.,* $\Omega = \bigcup_{e=1}^{E} \Omega_e$ with $\Gamma = \bigcup_{f=1}^{F} \Gamma_f$. Equation (6.54) becomes

$$\sum_{e=1}^{E} \int_{\Omega_e} a\dot{u}w \, \mathbf{d}\Omega - \sum_{e=1}^{E} \int_{\Omega_e} G^{jk} \sigma^i_j u_{;i} w_{;k} \, \mathbf{d}\Omega + \sum_{f=1}^{F} \int_{\Gamma_f} qw \, \mathbf{d}\Gamma + \sum_{e=1}^{E} \int_{\Omega_e} bw \, \mathbf{d}\Omega = 0 \tag{6.55}$$

If we now assume that the dependent variable u can be interpolated separately in space and in time we can write

$$u\left(\boldsymbol{x}, t\right) = \psi_n\left(\boldsymbol{x}\right) u^n\left(t\right) \tag{6.56}$$

or, in standard interpolation notation within an element,

$$u\left(\boldsymbol{\xi}, t\right) = \psi^{\beta}_{n}\left(\boldsymbol{\xi}\right) u^{n}_{,\beta}\left(t\right) \mathrm{S}\left(n, \beta\right) \tag{6.57}$$

where $u^n_{,\beta}\left(t\right)$ are the time varying nodal degrees-of-freedom for node n, global derivative β, $\psi^{\beta}_{n}\left(\boldsymbol{\xi}\right)$ are the corresponding basis functions and $\mathrm{S}\left(n, \beta\right)$ are the scale factors.

We can also interpolate the other variables in a similar manner *i.e.,*

$$\begin{aligned} q\left(\boldsymbol{\xi}, t\right) &= \psi^{\gamma}_{o}\left(\boldsymbol{\xi}\right) q^{o}_{,\gamma}\left(t\right) \mathrm{S}\left(o, \gamma\right) \\ a\left(\boldsymbol{\xi}\right) &= \psi^{\delta}_{p}\left(\boldsymbol{\xi}\right) a^{p}_{,\delta} \mathrm{S}\left(p, \delta\right) \\ \tilde{\boldsymbol{\sigma}}\left(\boldsymbol{\xi}\right) &= \psi^{\delta}_{p}\left(\boldsymbol{\xi}\right) \tilde{\boldsymbol{\sigma}}^{p}_{,\delta} \mathrm{S}\left(p, \delta\right) \\ b\left(\boldsymbol{\xi}, t\right) &= \psi^{\delta}_{p}\left(\boldsymbol{\xi}\right) b^{p}_{,\delta}\left(t\right) \mathrm{S}\left(p, \delta\right) \end{aligned} \tag{6.58}$$

where $q^o_{,\gamma}\left(t\right)$, $a^p_{,\delta}$, $\tilde{\boldsymbol{\sigma}}^p_{,\delta}$ and $b^p_{,\delta}\left(t\right)$ are the nodal degrees-of-freedom for the variables.

For a Galerkin finite element formulation we also choose the spatial weighting function w to be equal to the basis fucntions *i.e.,*

$$w\left(\boldsymbol{\xi}\right) = \psi^{\alpha}_{m}\left(\boldsymbol{\xi}\right) \mathrm{S}\left(m, \alpha\right) \tag{6.59}$$

Spatial integration:

Adopting the standard integration notation we can write the spatial integration term in Equation (6.55) as

$$\sum_{e=1}^{E}\int_{\mathbf{0}}^{\mathbf{1}} a\left(\boldsymbol{\xi}\right)\frac{\partial\left(\psi_{n}^{\beta}\left(\boldsymbol{\xi}\right)u_{,n}^{\beta}\left(t\right)\mathrm{S}\left(n,\beta\right)\right)}{\partial t}\psi_{m}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}-$$
$$\sum_{e=1}^{E}\int_{\mathbf{0}}^{\mathbf{1}} G^{jk}\frac{\partial x^{i}}{\partial\xi^{d}}\frac{\partial\xi^{e}}{\partial x^{j}}\frac{\partial\xi^{d}}{\partial\nu^{a}}\frac{\partial\nu^{b}}{\partial\xi^{e}}\tilde{\sigma}_{.b}^{a}\left(\boldsymbol{\xi}\right)\frac{\partial\xi^{s}}{\partial x^{i}}$$
$$\frac{\partial\left(\psi_{\beta}^{n}\left(\boldsymbol{\xi}\right)u_{,n}^{\beta}\left(t\right)\mathrm{S}\left(\beta,n\right)\right)}{\partial\xi^{s}}\frac{\partial\xi^{r}}{\partial x^{k}}\frac{\partial\left(\psi_{\alpha}^{m}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(\alpha,m\right)\right)}{\partial\xi^{r}}\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}+$$
$$\sum_{f=1}^{F}\int_{\mathbf{0}}^{\mathbf{1}}\psi_{o}^{\gamma}\left(\boldsymbol{\xi}\right)q_{,\gamma}^{o}\left(t\right)\mathrm{S}\left(o,\gamma\right)\psi_{m}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}+$$
$$\sum_{e=1}^{E}\int_{\mathbf{0}}^{\mathbf{1}} b\left(\boldsymbol{\xi},t\right)\psi_{m}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}=0 \quad (6.60)$$

where $\boldsymbol{J}\left(\boldsymbol{\xi}\right)$ is the *Jacobian* of the transformation from the integration $\boldsymbol{x}$ to $\boldsymbol{\xi}$ coordinates.

Taking values that are constant over the integration interval outside the integration gives

$$\sum_{e=1}^{E}\dot{u}_{,n}^{\beta}\left(t\right)\mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(n,\beta\right)\int_{\mathbf{0}}^{\mathbf{1}} a\left(\boldsymbol{\xi}\right)\psi_{m}^{\alpha}\left(\boldsymbol{\xi}\right)\psi_{n}^{\beta}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}-$$
$$\sum_{e=1}^{E}u_{,n}^{\beta}\left(t\right)\mathrm{S}\left(\alpha,m\right)\mathrm{S}\left(\beta,n\right)\int_{\mathbf{0}}^{\mathbf{1}}\frac{\partial\psi_{\alpha}^{m}\left(\boldsymbol{\xi}\right)}{\partial\xi^{r}}\frac{\partial\psi_{\beta}^{n}\left(\boldsymbol{\xi}\right)}{\partial\xi^{s}}\gamma^{rs}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}+$$
$$\sum_{f=1}^{F}q_{,\gamma}^{o}\left(t\right)\mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(o,\gamma\right)\int_{\mathbf{0}}^{\mathbf{1}}\psi_{m}^{\alpha}\left(\boldsymbol{\xi}\right)\psi_{o}^{\gamma}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}+$$
$$\sum_{e=1}^{E}b_{,\delta}^{p}\left(t\right)\mathrm{S}\left(m,\alpha\right)\mathrm{S}\left(p,\delta\right)\int_{\mathbf{0}}^{\mathbf{1}}\psi_{m}^{\alpha}\left(\boldsymbol{\xi}\right)\psi_{p}^{\delta}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}=0 \quad (6.61)$$

where $\gamma^{rs}\left(\boldsymbol{\xi}\right)$ is defined in Equations (6.26)–(6.28).

This is an equation of the form

$$\boldsymbol{C}\dot{\boldsymbol{u}}(t) + \boldsymbol{K}\boldsymbol{u}(t) + \boldsymbol{f}(t) = \mathbf{0} \tag{6.62}$$

where

$$\boldsymbol{f}(t) = \boldsymbol{N}\boldsymbol{q}(t) + \boldsymbol{R}\boldsymbol{b}(t) \tag{6.63}$$

The elemental damping matrix, $C_{mn}^{\alpha\beta}$, is given by

$$C_{mn}^{\alpha\beta} = \mathrm{S}(m,\alpha)\,\mathrm{S}(n,\beta)\int_{\mathbf{0}}^{\mathbf{1}} a(\boldsymbol{\xi})\,\psi_m^{\alpha}(\boldsymbol{\xi})\,\psi_n^{\beta}(\boldsymbol{\xi})\,|\boldsymbol{J}(\boldsymbol{\xi})|\;\mathbf{d}\boldsymbol{\xi} \tag{6.64}$$

The elemental stiffness matrix, $K_{mn}^{\alpha\beta}$, is given by

$$K_{mn}^{\alpha\beta} = -\mathrm{S}(m,\alpha)\,\mathrm{S}(n,\beta)\int_{\mathbf{0}}^{\mathbf{1}} \frac{\partial\psi_{\alpha}^{m}(\boldsymbol{\xi})}{\partial\xi^{r}}\frac{\partial\psi_{\beta}^{m}(\boldsymbol{\xi})}{\partial\xi^{s}}\gamma^{rs}(\boldsymbol{\xi})\,|\boldsymbol{J}(\boldsymbol{\xi})|\;\mathbf{d}\boldsymbol{\xi} \tag{6.65}$$

The elemental flux matrix, $N_{mo}^{\alpha\gamma}$, is given by

$$N_{mo}^{\alpha\gamma} = \mathrm{S}(m,\alpha)\,\mathrm{S}(o,\gamma)\int_{\mathbf{0}}^{\mathbf{1}} \psi_m^{\alpha}(\boldsymbol{\xi})\,\psi_o^{\gamma}(\boldsymbol{\xi})\,|\boldsymbol{J}(\boldsymbol{\xi})|\;\mathbf{d}\boldsymbol{\xi} \tag{6.66}$$

and the elemental source matrix, $R_{mp}^{\alpha\delta}$,

$$R_{mp}^{\alpha\delta} = \mathrm{S}(m,\alpha)\,\mathrm{S}(p,\delta)\int_{\mathbf{0}}^{\mathbf{1}} \psi_m^{\alpha}(\boldsymbol{\xi})\,\psi_p^{\delta}(\boldsymbol{\xi})\,|\boldsymbol{J}(\boldsymbol{\xi})|\;\mathbf{d}\boldsymbol{\xi} \tag{6.67}$$

6.1.5 Helmholtz Equation

6.1.6 Wave Equation

6.1.7 Advection-Diffusion Equation

6.1.8 Reaction-Diffusion Equation

6.1.9 Biharmonic Equation

6.2 Elasticity Class

6.2.1 Linear Elasticity

Finite element formulation of linear elasticity problems is

6.2.2 Finite Elasticity

Kinematics

As shown in Figure 6.1, consider a *material body* which is a three-dimensional smooth manifold with a boundary, $\mathfrak{B}$, which consists of a set of points which are refered to as *material points*. Consider also an ambient space manifold, $\mathfrak{S} \in \mathbb{R}^n$. The material body is only accessible to the observer when it moves through the ambient space. This motion is a time-dependent embedding on the material body into the ambient space. The embedding is known as a *placement of the body*. It is given by the mapping

$$\kappa\left(\mathcal{X},t\right) : \mathfrak{B} \to \mathfrak{S} \tag{6.68}$$

The embedded submanifold occupying a location in the ambient space is is called a *configuration* of $\mathfrak{B}$ and is given by

$$\mathcal{B}_t = \kappa_t\left(\mathfrak{B}\right) = \kappa\left(\mathfrak{B},t\right) \tag{6.69}$$

The customary (but not necessary) *reference placement* is given by

$$\kappa_0 : \mathfrak{B} \to \mathfrak{S} \tag{6.70}$$

and the region of space occupied by the reference placement *i.e.,* the *reference configuration* is given by

$$\mathcal{B}_0 = \kappa_0\left(\mathfrak{B}\right) \tag{6.71}$$

Points in $\mathcal{B}_0$ are denoted by capital letters *i.e.,* $X, Y, \ldots$. Points in $\mathcal{B}$ are denoted by lower case leters *i.e.,* $x, y, \ldots$.

A new configuration of $\mathfrak{B}$ is given by the deformation mapping

$$\chi : \mathcal{B} \to \mathbb{R}^3 \tag{6.72}$$

where a configuration represents a deformed state of the body. As the body moves we obtain a family of configurations. If we hold $X \in \mathcal{B}$ fixed can write $V_t\left(X\right) = V\left(X,t\right)$. We then have

$$V_t\left(X\right) = V\left(X,t\right) = \frac{\partial\chi\left(X,t\right)}{\partial t} = \frac{d\chi_X\left(t\right)}{dt} \tag{6.73}$$

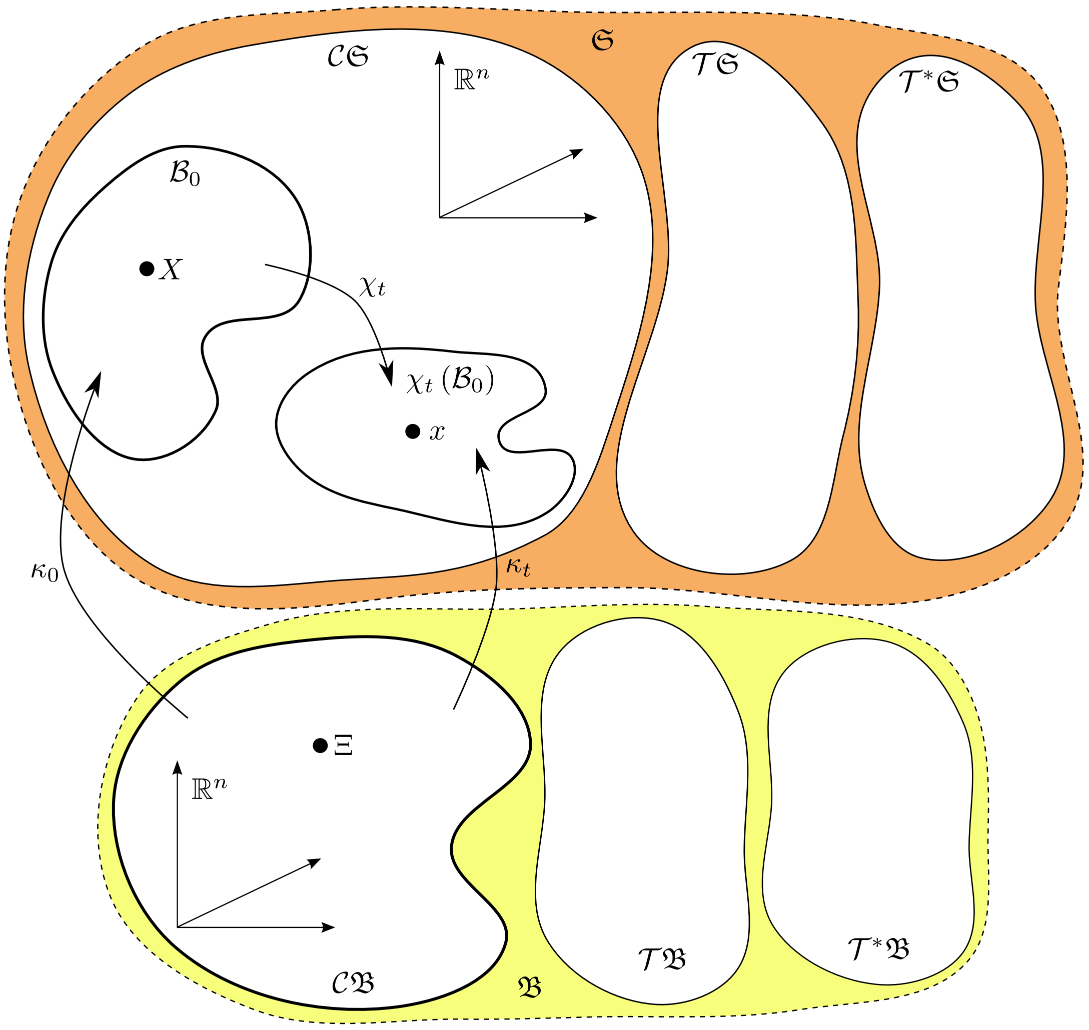
$\mathfrak{S}$
$\mathcal{C}\mathfrak{S}$
$\mathbb{R}^n$
$\mathcal{T}\mathfrak{S}$
$\mathcal{T}^*\mathfrak{S}$
$\mathcal{B}_0$
X
χ_t
$\chi_t\left(\mathcal{B}_0\right)$
x
κ_0
κ_t
Ξ
$\mathbb{R}^n$
$\mathcal{C}\mathfrak{B}$
$\mathfrak{B}$
$\mathcal{T}\mathfrak{B}$
$\mathcal{T}^*\mathfrak{B}$

FIGURE 6.1

Here V_t is called the *material velocity* of the motion. The *material acceleration* of the body is defined as

$$A_t(X) = A(X,t) = \frac{\partial V(X,t)}{\partial t} = \frac{dV_X(t)}{dt} \tag{6.74}$$

The *spatial velocity* of the motion is defined by v_t and the *spatial acceleration* of the motion is defined by a_t.

FIX BELOW

Consider the following line, area and volume forms in the reference configuration given by

$$\mathbf{d}\boldsymbol{X} = \sqrt{\det G_{IJ}} dX^1 \tag{6.75}$$

$$\mathbf{d}V = \sqrt{\det G_{IJ}} dX^1 \wedge dX^2 \wedge dX^3 \tag{6.76}$$

the corresponding volume form in the current configuration

$$\mathbf{d}\boldsymbol{v} = \sqrt{\det g_{ij}} dx^1 \wedge \cdots \wedge dx^n \tag{6.77}$$

is given by

$$\chi^* \mathbf{d}\boldsymbol{v} = J \mathbf{d}\boldsymbol{V} \tag{6.78}$$

where J is the *Jacobian* of the mapping and is given by

$$J(X) = \sqrt{\frac{\det g_{ij}}{\det G_{IJ}}} \det\left(\frac{\partial \chi^i(X)}{\partial X^I}\right) \tag{6.79}$$

Deformation Gradient

Let $\chi : \mathcal{B}_0 \to \chi(\mathcal{B}_0) \subset \mathfrak{S}$ be a deformation configuration of $\mathcal{B}$ in $\mathfrak{S}$. The tangent of the mapping *i.e.,* $\mathcal{T}\chi$ is denoted as $\boldsymbol{F}$ and is called the *deformation gradient* of χ *i.e.,* $\boldsymbol{F} = \mathcal{T}\chi$. For $X \in \mathcal{B}_0$ we have

$$\boldsymbol{F}_X = \boldsymbol{F}(X) : \mathcal{T}_X \mathcal{B}_\mathrm{o} \to \mathcal{T}_{\chi(X)} \mathfrak{S} \tag{6.80}$$

If X^I and x^i are the coordinates on $\mathcal{B}_0$ and $\mathfrak{S}$ then the deformation gradient tensor with respect to the coordinate bases are

$$\boldsymbol{F}(X) = F^i_I(X) \boldsymbol{G}^I \otimes \boldsymbol{g}_i = \frac{\partial \chi^i(X)}{\partial X^I} \boldsymbol{G}^I \otimes \boldsymbol{g}_i \tag{6.81}$$

Note that $\boldsymbol{F}$ is a two-point tensor.

The polar decomposition

$$\begin{tikzcd}
 & \mathcal{T}_X\mathfrak{B} \arrow[dr, "\boldsymbol{R}"] & \\
\mathcal{T}_X\mathfrak{B} \arrow[ur, "\boldsymbol{U}"] \arrow[rr, "\boldsymbol{F}"] \arrow[dr, "\boldsymbol{R}"'] & & \mathcal{T}_x\mathfrak{S} \\
 & \mathcal{T}_x\mathfrak{S} \arrow[ur, "\boldsymbol{V}"'] &
\end{tikzcd}$$

is given by

$$\boldsymbol{F} = \boldsymbol{R} \cdot \boldsymbol{U} = \boldsymbol{V} \cdot \boldsymbol{R} \tag{6.82}$$

where $\boldsymbol{U}$ is the *right stretch tensor*, $\boldsymbol{V}$ is the *left stretch tensor* and $\boldsymbol{R}$ is the *rotation tensor*.

If we let the deformed coordinates be given by the position vector, $\boldsymbol{z}(\boldsymbol{x}, t)$ then the deformation gradient tensor with respect to the undeformed $\boldsymbol{X}$ coordinates is given by

$$\boldsymbol{F}(\boldsymbol{X}) = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{X}} \tag{6.83}$$

or, in component form,

$$F^i_I = \frac{\partial z^i}{\partial X^I} = \frac{\partial z^i}{\partial \xi^r}\frac{\partial \xi^r}{\partial X^I} \tag{6.84}$$

The deformation gradient tensor can also be used to map lines, area and volume forms between the reference and current configurations and vice versa *i.e.,*

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{F}\mathrm{d}\boldsymbol{X} \tag{6.85}$$

$$\mathrm{d}\boldsymbol{a} = J\boldsymbol{F}^{-T}\mathrm{d}\boldsymbol{A} \tag{6.86}$$

$$\mathrm{d}v = J\mathrm{d}V \tag{6.87}$$

where $\mathrm{d}\boldsymbol{X}$, $\mathrm{d}\boldsymbol{A}$ and $\mathrm{d}V$ are the line, area and volume forms in the reference configuration and $\mathrm{d}\boldsymbol{x}$, $\mathrm{d}\boldsymbol{a}$ and $\mathrm{d}v$ are the line, area and volume forms in the current configuration. J is the

Jacobian and is given by

$$J = \sqrt{\frac{\det \boldsymbol{g}}{\det \boldsymbol{G}}} \det \boldsymbol{F} \tag{6.88}$$

The formula for mapping areas is known as *Nanson's formula* and is given by

$$\boldsymbol{n}\mathrm{d}\boldsymbol{a} = J\chi_{*}\boldsymbol{N}\mathrm{d}\boldsymbol{A} \tag{6.89}$$

or

$$\boldsymbol{n}\mathrm{d}\boldsymbol{a} = J\boldsymbol{F}^{-T}\boldsymbol{N}\mathrm{d}\boldsymbol{A} \tag{6.90}$$

where $\boldsymbol{N}$ and $\mathrm{d}\boldsymbol{A}$ are the unit normal and area form in the reference configuration, $\boldsymbol{n}$ and $\mathrm{d}\boldsymbol{a}$ are the unit normal and area form in the current configuration and J is the Jacobian.

Strain and deformation tensors

Strain and deformation tensors quantify the amount of strain or deformation *i.e.,* the amount of stretch or distance between material points.

There are three strain tensors in the *material* coordinate system. The *right Cauchy-Green (or Green) deformation tensor*, $\boldsymbol{C}$, is defined by

$$\boldsymbol{C}\left(X\right) : \mathcal{T}_X\mathcal{B} \to \mathcal{T}_X\mathcal{B} \tag{6.91}$$

as the pullback of the spatial metric tensor *i.e.,* $\boldsymbol{C}\left(X\right) = \boldsymbol{F}\left(X\right)^T\boldsymbol{g}\left(x\right)\boldsymbol{F}\left(X\right)$ or $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{g}\boldsymbol{F}$ where $x = \chi\left(X\right)$. In terms of coordinates we have

$$\boldsymbol{C} = C_{IJ}\boldsymbol{G}^I \otimes \boldsymbol{G}^J = g_{ij}F^i_IF^j_J\boldsymbol{G}^I \otimes \boldsymbol{G}^J \tag{6.92}$$

If $\boldsymbol{C}$ is invertible we also have $\boldsymbol{B} = \boldsymbol{C}^{-1}$ where $\boldsymbol{B}$ is called the *Piola deformation tensor*.

We also have the *Green-Lagrange strain tensor* is given by the difference in metric tensors

$$\boldsymbol{E}\left(X\right) = \frac{1}{2}\left(\boldsymbol{C}\left(X\right) - \boldsymbol{G}\right) \tag{6.93}$$

of, in component form

$$\boldsymbol{E} = E_{IJ}\boldsymbol{G}^I \otimes \boldsymbol{G}^J = \frac{1}{2}\left(C_{IJ} - G_{IJ}\right)\boldsymbol{G}^I \otimes \boldsymbol{G}^J \tag{6.94}$$

There are also three strain tensors in the *spatial* coordinate sytsem. The *left Cauchy-Green (or Finger) deformation tensor*, $\boldsymbol{b}$, is defined by

$$\boldsymbol{b}\left(x\right) : \mathcal{T}_x\chi\left(\mathcal{B}\right) \to \mathcal{T}_x\chi\left(\mathcal{B}\right) \tag{6.95}$$

as the push forward of the material metric tensor *i.e.,* $\boldsymbol{b}\left(x\right) = \boldsymbol{F}\left(X\right)\boldsymbol{G}\left(X\right)\boldsymbol{F}\left(X\right)^T$ or $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{G}\boldsymbol{F}^T$ where $X = \chi^{-1}\left(x\right)$. In terms of coordinates we have

$$\boldsymbol{b} = b^{ij}\boldsymbol{g}_i \otimes \boldsymbol{g}_j = G^{IJ}F^i_I F^j_J \boldsymbol{g}_i \otimes \boldsymbol{g}_j \tag{6.96}$$

The left and right Cauchy-Green deformation tensors get their left and right names from their relationship to the left and right stretch tensors *i.e.,* in cartesian coordinates

$$\boldsymbol{U} = \sqrt{\boldsymbol{C}} \tag{6.97}$$

and

$$\boldsymbol{V} = \sqrt{\boldsymbol{b}} \tag{6.98}$$

We also have $\boldsymbol{c} = \boldsymbol{b}^{-1}$ where $\boldsymbol{c}$ is the *Cauchy deformation tensor i.e.,*

$$\boldsymbol{c} = c_{ij}\boldsymbol{g}^i \otimes \boldsymbol{g}^j \tag{6.99}$$

The final spatial strain tensor is the *Euler-Almansi strain tensor*, $\boldsymbol{e}$, is defined as the difference in metrics by

$$\boldsymbol{e}\left(x\right) : \mathcal{T}_x\chi\left(\mathcal{B}\right) \to \mathcal{T}_x\chi\left(\mathcal{B}\right) \tag{6.100}$$

and

$$\begin{aligned} \boldsymbol{e}\left(x\right) &= \frac{1}{2}\left(\boldsymbol{g} - \boldsymbol{b}\left(x\right)^{-1}\right) \\ &= \frac{1}{2}\left(\boldsymbol{g} - \boldsymbol{c}\left(x\right)\right) \end{aligned} \tag{6.101}$$

where $\boldsymbol{c} = \boldsymbol{b}^{-1}$. In component form we have

$$\boldsymbol{e} = e_{ij}\boldsymbol{g}^i \otimes \boldsymbol{g}^j = \frac{1}{2}\left(g_{ij} - c_{ij}\right)\boldsymbol{g}^i \otimes \boldsymbol{g}^j \tag{6.102}$$

Note that we have the following relationships between the deformation tensors and metric tensors

$$\begin{aligned}
\boldsymbol{C}^\flat &= \chi^*\boldsymbol{g}^\flat & \boldsymbol{c}^\flat &= \chi_*\boldsymbol{G}^\flat && (6.103)\\
\boldsymbol{B}^\sharp &= \chi^*\boldsymbol{g}^\sharp & \boldsymbol{b}^\sharp &= \chi_*\boldsymbol{G}^\sharp && (6.104)\\
\boldsymbol{E}^\flat &= \chi^*\boldsymbol{e}^\flat & \boldsymbol{e}^\flat &= \chi_*\boldsymbol{E}^\flat && (6.105)\\
&= \frac{1}{2}\left(\chi^*\boldsymbol{g}^\flat - \boldsymbol{G}^\flat\right) & &= \frac{1}{2}\left(\boldsymbol{g}^\flat - \chi_*\boldsymbol{G}^\flat\right) && (6.106)
\end{aligned}$$

i.e.,

$$\begin{aligned}
\boldsymbol{C}^\flat &= \boldsymbol{F}^T\boldsymbol{g}^\flat\boldsymbol{F} & \boldsymbol{c}^\flat &= \boldsymbol{F}^{-T}\boldsymbol{G}^\flat\boldsymbol{F}^{-1} && (6.107)\\
\boldsymbol{B}^\sharp &= \boldsymbol{F}^{-1}\boldsymbol{g}^\sharp\boldsymbol{F}^{-T} & \boldsymbol{b}^\sharp &= \boldsymbol{F}\boldsymbol{G}^\sharp\boldsymbol{F}^T && (6.108)\\
\boldsymbol{E}^\flat &= \boldsymbol{F}^T\boldsymbol{e}^\flat\boldsymbol{F} & \boldsymbol{e}^\flat &= \boldsymbol{F}^{-T}\boldsymbol{E}^\flat\boldsymbol{F}^{-1} && (6.109)
\end{aligned}$$

or, in component form,

$$\begin{aligned}
C_{IJ} &= g_{ij}F^i_IF^j_J & c_{ij} &= G_{IJ}\left(F^{-1}\right)^I_i\left(F^{-1}\right)^J_j && (6.110)\\
B^{IJ} &= g^{ij}\left(F^{-1}\right)^I_i\left(F^{-1}\right)^J_j & b^{ij} &= G^{IJ}F^i_IF^j_J && (6.111)\\
E_{IJ} &= e_{ij}F^i_IF^j_J & e_{ij} &= E_{IJ}\left(F^{-1}\right)^I_i\left(F^{-1}\right)^J_j && (6.112)
\end{aligned}$$

Stress tensors

Stress is amount of force over an area. We can thus form a number of different stress tensors depending on what configuration we base the force and the area in.

The *Cauchy stress tensor*, $\boldsymbol{\sigma}$, has both the force and area referred to the *spatial* configuration. The Cauchy stress tensor is important as it quantifies the physical stress that exists in current configuration and can be measured. It is defined by

$$\boldsymbol{\sigma}(x) : \mathcal{T}_x\mathcal{B} \times \mathcal{T}_x\mathcal{B} \to \mathbb{R} \tag{6.113}$$

i.e.,

$$\boldsymbol{\sigma}(x) = \sigma^{ij}\boldsymbol{g}_i \otimes \boldsymbol{g}_j \tag{6.114}$$

Cauchy's law....

We can also define the *Kirchoff stress tensor*, $\boldsymbol{\tau}$, by scaling the Cauchy stress by the Jacobian *i.e.,*

$$\boldsymbol{\tau} = J\boldsymbol{\sigma} \tag{6.115}$$

To find other stress measures we need to make use of Piola's transform and idenity.

The *Piola transform* allows mapping of vector fields on the current configuration to equivalent vector fields in the reference configuration. The transform is given by

$$\boldsymbol{Y} = J\chi^*\boldsymbol{y} = J\boldsymbol{F}^{-1}\boldsymbol{y} \tag{6.116}$$

where $\boldsymbol{y}$ is the vector field on the current configuration and $\boldsymbol{Y}$ is the equivalent vector field on the reference configuration. In component form we have

$$Y^I = J\left(F^{-1}\right)^I_i y^i \tag{6.117}$$

Now if $\boldsymbol{Y}$ is the Piola transform of $\boldsymbol{y}$ then we have the *Piola identity* given by

$$\mathrm{Div}\,\boldsymbol{Y} = J\,\mathrm{div}\,\boldsymbol{y} \circ \chi \tag{6.118}$$

where Div is the divergence operator in the reference configuration and div is the divergence operator in the current configurtion.

Using the Piola transform we can construct two new stress tensors. If we apply the Piola transform to the second index of the Cauchy stress tensor we obtain the *First Piola-Kirchoff Stress Tensor i.e.,*

$$\boldsymbol{P} = J\boldsymbol{F}^{-T}\boldsymbol{\sigma} \tag{6.119}$$

or, in component form,

$$\boldsymbol{P} = P^{iI}\boldsymbol{g}_i \otimes \boldsymbol{G}_I = J\left(F^{-1}\right)^I_j \sigma^{ij}\boldsymbol{g}_i \otimes \boldsymbol{G}_I \tag{6.120}$$

The first Piola-Kirchoff stress tensor relates forces in the current configuration with areas in the reference configuration. Note that $\boldsymbol{P}$ is a two point tensor and is not symmetric.

The relationship between Cauchy stress and the First Piola-Kirchoff stress is given by

$$\boldsymbol{\sigma} = J^{-1}\boldsymbol{P}\boldsymbol{F}^T \tag{6.121}$$

If we now apply the Piola transform to the first index of the First Piola-Kirchoff stress tensor we obtain the *Second Piola-Kirchoff Stress Tensor i.e.,*

$$\boldsymbol{S} = J\boldsymbol{F}^{-1}\boldsymbol{P} \tag{6.122}$$

or, in terms of the Cauchy stress,

$$\boldsymbol{S} = J\boldsymbol{F}^{-1}\boldsymbol{\sigma}\boldsymbol{F}^{-T} \tag{6.123}$$

or, in terms of components

$$\boldsymbol{S} = S^{IJ}\boldsymbol{G}_I \otimes \boldsymbol{G}_J = J\left(F^{-1}\right)^I_i \left(F^{-1}\right)^J_j \sigma^{ij}\boldsymbol{G}_I \otimes \boldsymbol{G}_J \tag{6.124}$$

The second Piola-Kirchoff stress tensor relates forces in the reference configuration with areas in the reference configuration.

Note that the Kirchoff stress is just the push forward of the second Piola-Kirchoff stress *i.e.,*

$$\boldsymbol{\tau} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T \tag{6.125}$$

and

$$\boldsymbol{S} = \boldsymbol{F}^{-1}\boldsymbol{\tau}\boldsymbol{F}^{-T} \tag{6.126}$$

BIOT STRESS

COVECTED STRESSS

The relationships between the stress tensors are given in Table 6.1.

	$\boldsymbol{\sigma}$	$\boldsymbol{\tau}$	$\boldsymbol{P}$	$\boldsymbol{S}$
$\boldsymbol{\sigma}$	-	$J^{-1}\boldsymbol{\tau}$	$J^{-1}\boldsymbol{P}\boldsymbol{F}^T$	$J^{-1}\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T$
$\boldsymbol{\tau}$	$J\boldsymbol{\sigma}$	-	$\boldsymbol{P}\boldsymbol{F}^T$	$\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T$
$\boldsymbol{P}$	$J\boldsymbol{\sigma}\boldsymbol{F}^{-T}$	$\boldsymbol{\tau}\boldsymbol{F}^{-T}$	-	$\boldsymbol{F}\boldsymbol{S}$
$\boldsymbol{S}$	$\mathrm{J}\boldsymbol{F}^{-1}\boldsymbol{\sigma}\boldsymbol{F}^{-T}$	$\boldsymbol{F}^{-1}\boldsymbol{\sigma}\boldsymbol{F}^{-T}$	$\boldsymbol{F}^{-1}\boldsymbol{P}$	-

TABLE 6.1: Reltionships between stress tensors.

For incompressible materials we need an additional volumetric stress. The *hydrostatic stress* is a Cauchy stress and so we have

$$\boldsymbol{\sigma}_p = -p\boldsymbol{g} \tag{6.127}$$

or in component form

$$\boldsymbol{\sigma}_p = \sigma_p^{ij}\boldsymbol{g}_i \otimes \boldsymbol{g}_j = -pg^{ij}\boldsymbol{g}_i \otimes \boldsymbol{g}_j \tag{6.128}$$

where p is known as the *hydrostatic pressure*.

We can pull this stress back to give a second Piola-Kirchoff stress via the pullback operation for a second order tensor *i.e.,*

$$\boldsymbol{S}_p = J\boldsymbol{F}^{-1}\boldsymbol{\sigma}_p\boldsymbol{F}^{-T} = -pJ\boldsymbol{C}^{-1} \tag{6.129}$$

or, in component form,

$$\boldsymbol{S}_p = S_p^{IJ}\boldsymbol{G}_I \otimes \boldsymbol{G}_J = -JF_i^I pg^{ij}F_j^J\boldsymbol{G}_I \otimes \boldsymbol{G}_J = -pJ\left(C^{-1}\right)^{IJ}\boldsymbol{G}_I \otimes \boldsymbol{G}_J \tag{6.130}$$

RESTRICTIONS ON STRESS TENSOR E.G. SYMMETRY

Constituative Relationships

The relationship between stress and strain is known as a *consituative relationship*. Materials for which the constituative relationship is just a function of the current state of deformation are known as *elastic*.

Hyperelasticity

For the special case whereby the work done by stresses during deformation is just a function of the initial configuration and the current configuration are known as *hyperelastic*. As a consequence hyperelastic materials are independent of the path of deformation and just depend on a *stored energy function* or *elastic potential*, ψ.

WORK CONJUGATES: P and F, S and C/E.

The first Piola-Kirchoff stress tensor is thus a function of postion and the deformation gradient tensor *i.e.,*

$$\boldsymbol{P} = \boldsymbol{P}\left(\boldsymbol{X}, \boldsymbol{F}\left(\boldsymbol{X}\right)\right) \tag{6.131}$$

In terms of the stored energy function we have

$$\boldsymbol{P}\left(\boldsymbol{X}\right) = \frac{\partial \psi\left(\boldsymbol{F}\left(\boldsymbol{X}\right), \boldsymbol{X}\right)}{\partial \boldsymbol{F}\left(\boldsymbol{X}\right)} \tag{6.132}$$

or, in component form,

$$\boldsymbol{P} = P^{iI}\boldsymbol{g}_i \otimes \boldsymbol{G}_I = \frac{\partial \psi}{\partial F^i_I}\boldsymbol{g}_i \otimes \boldsymbol{G}_I \tag{6.133}$$

ABOVE IS NOT RIGHT IN TERMS OF POSITION OF INDICES.

The second Piola-Kirchoff stress tensor is thus a function of position and the right Cauchy-Green deformation tensor (or, equivalently, the Green-Lagrange strain tensor) *i.e.,*

$$\boldsymbol{S} = \boldsymbol{S}\left(\boldsymbol{X}, \boldsymbol{C}\left(\boldsymbol{X}\right)\right) \tag{6.134}$$

Viscoelasticity

Plasticity

Anisotropy

In order to deal with anisotropy we wish to base our stress and strain calculation on fibre, $\boldsymbol{\nu}$, coordinates. To change our reference coordinate system from $\boldsymbol{X}$ to $\boldsymbol{\nu}$ we need to transform $\boldsymbol{F}\left(\boldsymbol{X}\right)$. As $\boldsymbol{F}\left(\boldsymbol{X}\right)$ is a two point tensor the transformation rule for transforming just the reference coordinates is given by

$$\boldsymbol{F}\left(\boldsymbol{\nu}\right) = \boldsymbol{Q}\boldsymbol{F}\left(\boldsymbol{X}\right) \tag{6.135}$$

where $\boldsymbol{Q}$ is the rotation matrix from $\boldsymbol{X}$ to $\boldsymbol{\nu}$ *i.e.,*

$$F^i_A = \frac{\partial X^M}{\partial \nu^A}F^i_M = \frac{\partial X^M}{\partial \nu^A}\frac{\partial z^i}{\partial \xi^k}\frac{\partial \xi^k}{\partial X^M} \tag{6.136}$$

To allow for growth we use a multiplicative decomposition approach *i.e.*,

$$\boldsymbol{F}(\boldsymbol{\nu}) = \boldsymbol{F}_e(\boldsymbol{\nu})\,\boldsymbol{F}_g(\boldsymbol{\nu}) \tag{6.137}$$

where $\boldsymbol{F}_g(\boldsymbol{\nu})$ is the growth tensor with respect to fibre coordinates and $\boldsymbol{F}_e(\boldsymbol{\nu})$ is the elastic component of the deformation gradient tensor in fibre coordinates.

The elastic component of the deformation gradient tensor can be calculated from

$$\boldsymbol{F}_e(\boldsymbol{\nu}) = \boldsymbol{F}(\boldsymbol{\nu})\,\boldsymbol{F}_g^{-1}(\boldsymbol{\nu}) \tag{6.138}$$

In component form we have

$$F^i_A = (F_e)^i_B\,(F_g)^B_A \tag{6.139}$$

and

$$(F_e)^i_B = F^i_A\left(F_g^{-1}\right)^A_B \tag{6.140}$$

The Jacobian of the growth component of the deformation is given by $J_g = \det \boldsymbol{F}_g(\boldsymbol{\nu})$ and the Jacobian of the elastic component of the deformation is given by $J_e = \det \boldsymbol{F}_e(\boldsymbol{\nu})$.

The right Cauchy Green deformation tensor in fibre coordinates is now given by the pullback of the current configuration metric tensor, $\boldsymbol{g}$,

$$\boldsymbol{C}(\boldsymbol{\nu}) = \boldsymbol{F}_e^T(\boldsymbol{\nu})\,\boldsymbol{g}\boldsymbol{F}_e(\boldsymbol{\nu}) \tag{6.141}$$

In component form we have

$$C_{AB} = g_{ij}\,(F_e)^i_A\,(F_e)^j_B \tag{6.142}$$

and

$$E_{AB} = \frac{1}{2}\left(C_{AB} - G_{AB}\right) \tag{6.143}$$

To find the stress tensors in deformed coordinates we need to push the second Piola Kirchhoff tensor in the reference coordinates forward to the deformed coordinates, $\boldsymbol{x}$, to give the Kirchhoff stress tensor, $\boldsymbol{\tau}(\boldsymbol{x})$. The push foward is given by

$$\boldsymbol{\tau}(\boldsymbol{x}) = \boldsymbol{F}_e(\boldsymbol{\nu})\,\boldsymbol{S}(\boldsymbol{\nu})\,\boldsymbol{F}_e^T(\boldsymbol{\nu}) \tag{6.144}$$

The Cauchy stress tensor, $\boldsymbol{\sigma}\left(\boldsymbol{x}\right)$, can then be calculated from the Kirchhoff stress tensor using the Jacobian of the deformation *i.e.,*

$$\boldsymbol{\sigma}\left(\boldsymbol{x}\right) = {J_e}^{-1}\boldsymbol{\tau}\left(\boldsymbol{x}\right) = {J_e}^{-1}\boldsymbol{F}_e\left(\boldsymbol{\nu}\right)\boldsymbol{S}\left(\boldsymbol{\nu}\right){\boldsymbol{F}_e}^T\left(\boldsymbol{\nu}\right) \tag{6.145}$$

7

In component form we have

$$\tau^{ij} = \left(F_e\right)^i_B T^{BC}\left({F_e}^T\right)^j_C \tag{6.146}$$

and

$$\sigma^{ij} = {J_e}^{-1}\left(F_e\right)^i_B T^{BC}\left({F_e}^T\right)^j_C \tag{6.147}$$

Elasticity Tensors

If $\boldsymbol{P}\left(\boldsymbol{X},\boldsymbol{F}\right)$ is the first Piola-Kirchoff constituative function that depends on $\boldsymbol{X}$ and the deformation gradient tensor $\boldsymbol{F}$. The *first elasticity tensor*, A, is given by

$$\mathsf{A} = \frac{\partial\boldsymbol{P}}{\partial\boldsymbol{F}} \tag{6.148}$$

or

$$A^{iAB}_j = \frac{\partial P^{iA}}{\partial F^j_B} \tag{6.149}$$

Note that A is a two-point tensor.

If $\boldsymbol{S}\left(\boldsymbol{X},\boldsymbol{C}\right)$ is the second Piola-Kirchoff constitutive function that depends on $\boldsymbol{X}$ and the right Cauchy-Green strain tensor $\boldsymbol{C}$. The *second elasticity tensor*, C, is given by

$$\mathsf{C} = \frac{\partial\boldsymbol{S}}{\partial\boldsymbol{C}} \tag{6.150}$$

or

$$C^{ABCD} = \frac{\partial S^{AB}}{\partial C_{CD}} \tag{6.151}$$

We can derive a relationship between $\mathbf{A}$ and $\mathbf{C}$. Differentiating

$$\boldsymbol{P} = \boldsymbol{F}\boldsymbol{S} \tag{6.152}$$

or

$$P^{iA} = F^i_B S^{BA} \tag{6.153}$$

with respect to the deformation gradient tensor gives

$$\frac{\partial P^{iA}}{\partial F^j_B} = F^i_C \frac{\partial S^{CA}}{\partial C_{DE}} \frac{\partial C_{DE}}{\partial F^i_C} + \frac{\partial F^i_C}{\partial F^j_B} T^{CA} \tag{6.154}$$

Now

$$C_{DE} = F^l_D F^k_E g_{lk} \tag{6.155}$$

and so

$$\frac{\partial C_{DE}}{\partial F^j_B} = \delta^B_D F^k_E g_{jk} + F^l_D \delta^B_E g_{lj} \tag{6.156}$$

Substituting this into the above equation gives

$$\begin{aligned} \frac{\partial P^{iA}}{\partial F^j_B} &= C^{CADE} \left(\delta^B_D F^k_E g_{jk} + F^l_D \delta^B_E g_{lj} \right) F^i_C + T^{CA} \delta^j_i \delta^B_C \\ &= C^{CABE} F^k_E F^i_C g_{jk} + C^{CADE} F^l_D F^i_C g_{lj} + T^{BA} \delta^i_j \end{aligned} \tag{6.157}$$

Now, using the symmetries $C^{CABE} = C^{CAEB}$ and $T^{AB} = T^{BA}$ we have

$$\mathbf{A} = 2\mathbf{C} \cdot \boldsymbol{F} \cdot \boldsymbol{F} \cdot \boldsymbol{g} + \boldsymbol{S} \otimes \boldsymbol{I} \tag{6.158}$$

or

$$A^{iAB}_j = 2C^{CADB} F^k_D F^i_C g_{kj} + T^{AB} \delta^i_j \tag{6.159}$$

Now, we can also define the *first spatial elasticity tensor*, $\mathbf{a}$, and the *second spatial elasticity* *tensor*, c, using push forwards and Piola transforms of A and C respectively *i.e.,*

$$\begin{aligned} \mathsf{a} &= \frac{1}{J}\chi_*\mathsf{A} \\ \mathsf{c} &= \frac{2}{J}\chi_*\mathsf{C} \end{aligned} \tag{6.160}$$

or

$$\begin{aligned} a_j^{ikl} &= \frac{1}{J}F_A^kF_B^lA_j^{iAB} \\ c^{ijkl} &= \frac{2}{J}F_A^iF_B^jF_C^kF_D^lC^{ABCD} \end{aligned} \tag{6.161}$$

The relationship between a and c is given by

$$\mathsf{a} = 2\mathsf{c}\cdot\boldsymbol{g} + \boldsymbol{\sigma}\otimes\boldsymbol{I} \tag{6.162}$$

or

$$a_j^{ikl} = c^{ikml}g_{mj} + \sigma^{kl}\delta_j^i \tag{6.163}$$

There is also a relationship between A and a known as the *Piola identity i.e.,*

$$\nabla_{\boldsymbol{X}}\cdot(\mathsf{A}\cdot\boldsymbol{U}) = J\nabla_{\boldsymbol{x}}\cdot(\mathsf{a}\cdot\boldsymbol{u}) \tag{6.164}$$

where

$$\boldsymbol{u} = \chi_*\boldsymbol{U} \tag{6.165}$$

Note that the tensor c is different to C in that it is *not* given by $\dfrac{\partial\sigma^{ij}}{\partial c_{kl}}$ and $\sigma^{ij} \neq 2\dfrac{\partial W}{\partial c_{ij}}$. They are instead given by

$$c^{ijkl} = \frac{\partial\sigma^{ij}}{\partial g_{kl}} \tag{6.166}$$

and

$$\sigma^{ij} = 2\frac{\partial W}{\partial g_{ij}} \tag{6.167}$$

6.2.3 Principle of Virtual Work

Consider a configuration, defined by a displacement field $\boldsymbol{u}$, of some deformable body, $\mathcal{B}$, subject to some *displacement boundary conditions* $\boldsymbol{u} = \bar{\boldsymbol{u}}$ over some part of the boundary $\partial\mathcal{B}_u$ and some *traction boundary conditions* $\boldsymbol{t} = \boldsymbol{\sigma} \cdot \boldsymbol{n} = \bar{\boldsymbol{t}}$ over some part of the boundary $\partial\mathcal{B}_t$. Note that $\partial\mathcal{B}_u \cap \partial\mathcal{B}_t = \emptyset$ and $\partial\mathcal{B}_u \cup \partial\mathcal{B}_t \subseteq \partial\mathcal{B}$.

We wish to find the displacement field $\boldsymbol{u}$ which satisfies the deformation boundary conditions where they are applied and the equations of motion and the traction boundary conditions where they are applied. A displacement field $\boldsymbol{u}$ which satisfies the displacemnet boundary conditions where they are applied but whose resulting stress field does not necessarily satisfy the equations of motion or the traction boundary conditions where they are applied is known as a *kinematically admissable displacement field.* Similarily, a stress field $\boldsymbol{\sigma}$ which satisfies the equations of motion and the traction boundary conditions where they are applied but whost resulting displacement field does not necessarily satisfy the displacement boundary conditions where they are applied is known as a *statically admissable stress field.*

For a statically admissable stress field $\boldsymbol{\sigma}$ conservation of momentum gives us the equations of motion *i.e.,*

$$\int_{\mathcal{B}} \rho\boldsymbol{a}\,\mathrm{d}v = \int_{\mathcal{B}} (\nabla\cdot\boldsymbol{\sigma} + \boldsymbol{b})\;\mathrm{d}v \tag{6.168}$$

where $\boldsymbol{a}$ is the acceleration of the body and $\boldsymbol{b}$ are the body forces. If we now multiply by the equations of motion by a kinematically admissable displacement field $\boldsymbol{u}$ we obtain

$$\begin{aligned}\int_{\mathcal{B}} \rho\boldsymbol{a}\cdot\boldsymbol{u}\,\mathrm{d}v &= \int_{\mathcal{B}} (\nabla\cdot\boldsymbol{\sigma} + \boldsymbol{b})\cdot\boldsymbol{u}\,\mathrm{d}v \\ &= \int_{\mathcal{B}} (\nabla\cdot\boldsymbol{\sigma})\cdot\boldsymbol{u}\,\mathrm{d}v + \int_{\mathcal{B}} \boldsymbol{b}\cdot\boldsymbol{u}\,\mathrm{d}v\end{aligned} \tag{6.169}$$

Now, by the vector identity

$$\nabla\cdot(\boldsymbol{\sigma}\cdot\boldsymbol{u}) = (\nabla\cdot\boldsymbol{\sigma})\cdot\boldsymbol{u} + \boldsymbol{\sigma}^T : \nabla\boldsymbol{u} \tag{6.170}$$

we have

$$(\nabla \cdot \boldsymbol{\sigma}) \cdot \boldsymbol{u} = \nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{u}) - \boldsymbol{\sigma} : \nabla \boldsymbol{u} \tag{6.171}$$

as the stress tensor, $\boldsymbol{\sigma}$, symmetric and so $\boldsymbol{\sigma}^T = \boldsymbol{\sigma}$. Equation (6.169) becomes

$$\begin{aligned}\int\limits_{\mathcal{B}} \rho \boldsymbol{a} \cdot \boldsymbol{u} \,\mathrm{d}v &= \int\limits_{\mathcal{B}} \left(\nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{u}) - \boldsymbol{\sigma} : \nabla \boldsymbol{u}\right) \,\mathrm{d}v + \int\limits_{\mathcal{B}} \boldsymbol{b} \cdot \boldsymbol{u} \,\mathrm{d}v \\ &= \int\limits_{\mathcal{B}} \nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{u}) \,\mathrm{d}v - \int\limits_{\mathcal{B}} \boldsymbol{\sigma} : \nabla \boldsymbol{u} \,\mathrm{d}v + \int\limits_{\mathcal{B}} \boldsymbol{b} \cdot \boldsymbol{u} \,\mathrm{d}v\end{aligned} \tag{6.172}$$

Applying the divergence theorem to the first integral on the right hand side gives

$$\begin{aligned}\int\limits_{\mathcal{B}} \nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{u}) \,\mathrm{d}v &= \int\limits_{\partial\mathcal{B}} (\boldsymbol{\sigma} \cdot \boldsymbol{u}) \cdot \boldsymbol{n} \,\mathrm{d}a \\ &= \int\limits_{\partial\mathcal{B}} (\boldsymbol{\sigma} \cdot \boldsymbol{n}) \cdot \boldsymbol{u} \,\mathrm{d}a\end{aligned} \tag{6.173}$$

or, by Cauchy's law, $\boldsymbol{t} = \boldsymbol{\sigma} \cdot \boldsymbol{n}$, we have

$$\int\limits_{\mathcal{B}} \nabla \cdot (\boldsymbol{\sigma} \cdot \boldsymbol{u}) \,\mathrm{d}v = \int\limits_{\partial\mathcal{B}} \boldsymbol{t} \cdot \boldsymbol{u} \,\mathrm{d}a \tag{6.174}$$

Equation (6.169) thus becomes

$$\int\limits_{\mathcal{B}} \rho \boldsymbol{a} \cdot \boldsymbol{u} \,\mathrm{d}v = \int\limits_{\partial\mathcal{B}} \boldsymbol{t} \cdot \boldsymbol{u} \,\mathrm{d}a - \int\limits_{\mathcal{B}} \boldsymbol{\sigma} : \nabla \boldsymbol{u} \,\mathrm{d}v + \int\limits_{\mathcal{B}} \boldsymbol{b} \cdot \boldsymbol{u} \,\mathrm{d}v \tag{6.175}$$

or

$$\int\limits_{\mathcal{B}} \rho \boldsymbol{a} \cdot \boldsymbol{u} \,\mathrm{d}v + \int\limits_{\mathcal{B}} \boldsymbol{\sigma} : \nabla \boldsymbol{u} \,\mathrm{d}v = \int\limits_{\partial\mathcal{B}} \boldsymbol{t} \cdot \boldsymbol{u} \,\mathrm{d}a + \int\limits_{\mathcal{B}} \boldsymbol{b} \cdot \boldsymbol{u} \,\mathrm{d}v \tag{6.176}$$

Taking the boundary conditions into account we have

$$\int_{\mathcal{B}} \rho\boldsymbol{a}\cdot\boldsymbol{u}\,\mathrm{d}v + \int_{\mathcal{B}} \boldsymbol{\sigma} : \nabla\boldsymbol{u}\,\mathrm{d}v = \int_{\partial\mathcal{B}_u} \boldsymbol{t}\cdot\bar{\boldsymbol{u}}\,\mathrm{d}a + \int_{\partial\mathcal{B}_t} \bar{\boldsymbol{t}}\cdot\boldsymbol{u}\,\mathrm{d}a + \int_{\mathcal{B}} \boldsymbol{b}\cdot\boldsymbol{u}\,\mathrm{d}v \tag{6.177}$$

If we now consider a second kinematically admissable displacement field $\boldsymbol{u}^*$ then the above equation will also hold *i.e.,*

$$\int_{\mathcal{B}} \rho\boldsymbol{a}\cdot\boldsymbol{u}^*\,\mathrm{d}v + \int_{\mathcal{B}} \boldsymbol{\sigma} : \nabla\boldsymbol{u}^*\,\mathrm{d}v = \int_{\partial\mathcal{B}_u} \boldsymbol{t}\cdot\bar{\boldsymbol{u}}\,\mathrm{d}a + \int_{\partial\mathcal{B}_t} \bar{\boldsymbol{t}}\cdot\boldsymbol{u}^*\,\mathrm{d}a + \int_{\mathcal{B}} \boldsymbol{b}\cdot\boldsymbol{u}^*\,\mathrm{d}v \tag{6.178}$$

Now, subtracting Equation (6.178) from Equation (6.177) gives us

$$\int_{\mathcal{B}} \rho\boldsymbol{a}\cdot\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\,\mathrm{d}v + \int_{\mathcal{B}} \boldsymbol{\sigma} : \nabla\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\,\mathrm{d}v = \int_{\partial\mathcal{B}_u} \boldsymbol{t}\cdot\bar{\boldsymbol{u}}\,\mathrm{d}a + \int_{\partial\mathcal{B}_t} \bar{\boldsymbol{t}}\cdot\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\,\mathrm{d}a + \int_{\mathcal{B}} \boldsymbol{b}\cdot\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\,\mathrm{d}v \tag{6.179}$$

If we now define the *virtual displacements* as $\delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}^*$ and note that $\delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}^* = \bar{\boldsymbol{u}} - \bar{\boldsymbol{u}} = \boldsymbol{0}$ on $\partial\mathcal{B}_u$ then we have

$$\int_{\mathcal{B}} \rho\boldsymbol{a}\cdot\delta\boldsymbol{u}\,\mathrm{d}v + \int_{\mathcal{B}} \boldsymbol{\sigma} : \nabla\delta\boldsymbol{u}\,\mathrm{d}v = \int_{\partial\mathcal{B}_t} \bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}\,\mathrm{d}a + \int_{\mathcal{B}} \boldsymbol{b}\cdot\delta\boldsymbol{u}\,\mathrm{d}v \tag{6.180}$$

This is known as the *Principle of Virtual Work i.e.,* when a deformable body undergoes some virtual displacement, $\delta\boldsymbol{u}$, the *kinetic work*, $W_{kin}\left(\delta\boldsymbol{u}\right)$, plus the *internal work*, $W_{int}\left(\delta\boldsymbol{u}\right)$, is balanced by the *external work*, $W_{ext}\left(\delta\boldsymbol{u}\right)$, where

$$\begin{aligned} W_{kin}\left(\delta\boldsymbol{u}\right) &= \int\limits_{\mathcal{B}} \rho\boldsymbol{a}\cdot\delta\boldsymbol{u}\,\mathbf{d}v \\ W_{int}\left(\delta\boldsymbol{u}\right) &= \int\limits_{\mathcal{B}} \boldsymbol{\sigma} : \nabla\delta\boldsymbol{u}\,\mathbf{d}v \\ W_{ext}\left(\delta\boldsymbol{u}\right) &= W_{surf}\left(\delta\boldsymbol{u}\right) + W_{body}\left(\delta\boldsymbol{u}\right) \\ &= \int\limits_{\partial\mathcal{B}_t} \bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}\,\mathbf{d}a + \int\limits_{\partial\mathcal{B}_p} \bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}\,\mathbf{d}a + \int\limits_{\mathcal{B}} \boldsymbol{b}\left(\boldsymbol{u}\right)\cdot\delta\boldsymbol{u}\,\mathbf{d}v \end{aligned} \tag{6.181}$$

where $W_{surf}\left(\boldsymbol{u},\delta\boldsymbol{u}\right)$ is the external work due to surface forces and $W_{body}\left(\boldsymbol{u},\delta\boldsymbol{u}\right)$ is the external work due to body forces.

Note that the internal work is often writen in terms of *virtual strain* instead of virtual displacement. Consider

$$\begin{aligned} \nabla\delta\boldsymbol{u} &= \frac{1}{2}\left(\nabla\delta\boldsymbol{u} + \left(\nabla\delta\boldsymbol{u}\right)^T\right) + \frac{1}{2}\left(\nabla\delta\boldsymbol{u} - \left(\nabla\delta\boldsymbol{u}\right)^T\right) \\ &= \delta\left[\frac{1}{2}\left(\nabla\boldsymbol{u} + \left(\nabla\boldsymbol{u}\right)^T\right)\right] + \delta\left[\frac{1}{2}\left(\nabla\boldsymbol{u} - \left(\nabla\boldsymbol{u}\right)^T\right)\right] \\ &= \delta\boldsymbol{\epsilon} + \delta\boldsymbol{\omega} \end{aligned} \tag{6.182}$$

where $\boldsymbol{\epsilon}$ is the *small strain tensor* and $\boldsymbol{\omega}$ is the *small rotation tensor*. The internal work is now given by

$$\begin{aligned} W_{int}\left(\boldsymbol{u},\delta\boldsymbol{u}\right) &= \int\limits_{\mathcal{B}} \boldsymbol{\sigma}\left(\boldsymbol{u}\right) : \nabla\delta\boldsymbol{u}\,\mathbf{d}v \\ &= \int\limits_{\mathcal{B}} \boldsymbol{\sigma}\left(\boldsymbol{u}\right) : \left(\delta\boldsymbol{\epsilon} + \delta\boldsymbol{\omega}\right)\,\mathbf{d}v \\ &= \int\limits_{\mathcal{B}} \boldsymbol{\sigma}\left(\boldsymbol{u}\right) : \delta\boldsymbol{\epsilon}\,\mathbf{d}v + \int\limits_{\mathcal{B}} \boldsymbol{\sigma}\left(\boldsymbol{u}\right) : \delta\boldsymbol{\omega}\,\mathbf{d}v \end{aligned} \tag{6.183}$$

Therefore

$$W_{int}\left(\boldsymbol{u},\delta\boldsymbol{\epsilon}\right)=\int\limits_{\mathcal{B}}\boldsymbol{\sigma}\left(\boldsymbol{u}\right):\delta\boldsymbol{\epsilon}\,\mathrm{d}v \tag{6.184}$$

as the stress tensor is a symmetric tensor and the small rotation tensor is a skew-symmetric tensor and the double dot product between a symmetric and skew-symmetric tensor is always zero.

In terms of strain the principle of virtual work can thus be stated as

$$\int\limits_{\mathcal{B}}\rho\boldsymbol{a}\cdot\delta\boldsymbol{u}\,\mathrm{d}v+\int\limits_{\mathcal{B}}\boldsymbol{\sigma}:\delta\boldsymbol{\epsilon}\,\mathrm{d}v=\int\limits_{\partial\mathcal{B}_t}\bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}\,\mathrm{d}a+\int\limits_{\mathcal{B}}\boldsymbol{b}\cdot\delta\boldsymbol{u}\,\mathrm{d}v \tag{6.185}$$

TALK ABOUT WORK CONJUGACY AND SHOW OTHER CONJUGATE FORMS

The internal work can also be expressed in the undeformed manifold in terms of the 2^{nd} Piola-Kirchoff stress, $\boldsymbol{S}$ and the Green-Lagrange strain tensor, $\boldsymbol{E}$, *i.e.,*

$$W_{int}\left(\boldsymbol{u},\delta\boldsymbol{E}\right)=\int\limits_{\mathcal{B}_0}\boldsymbol{S}\left(\boldsymbol{u}\right):\delta\boldsymbol{E}\,\mathrm{d}V \tag{6.186}$$

Considering the external surface work it is often the case that this is due to an external pressure. Thus the traction is given by

$$\bar{\boldsymbol{t}}=p_{ext}\hat{\boldsymbol{n}} \tag{6.187}$$

where p_{ext} is the applied external pressure and $\hat{\boldsymbol{n}}$ is the unit normal vector to the surface element $\mathbf{d}\boldsymbol{a}$. Now if the surface is parameterised by the coordinates ξ and η then

$$\mathbf{d}\boldsymbol{a}=\left\|\frac{\partial\boldsymbol{x}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta}\right\|_2\mathbf{d}\xi\wedge\mathbf{d}\eta \tag{6.188}$$

and the unit normal is given by

$$\hat{\boldsymbol{n}}=\frac{\dfrac{\partial\boldsymbol{x}}{\partial\xi}\times\dfrac{\partial\boldsymbol{x}}{\partial\eta}}{\left\|\dfrac{\partial\boldsymbol{x}}{\partial\xi}\times\dfrac{\partial\boldsymbol{x}}{\partial\eta}\right\|_2} \tag{6.189}$$

where $\boldsymbol{x}$ are the coordinates in the current configuration *i.e.,* $\boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}$.

The external surface work is thus given by

$$\begin{aligned}
W_{surf}\left(\boldsymbol{u}, p_{ext}, \delta\boldsymbol{u}\right) &= \int\limits_{\partial\mathcal{B}_t} p_{ext}\hat{\boldsymbol{n}}\left(\boldsymbol{u}\right)\cdot\delta\boldsymbol{u}\,\mathbf{d}\boldsymbol{a} \\
&= \int\limits_{\xi}\int\limits_{\eta} p_{ext}\frac{\dfrac{\partial\boldsymbol{x}}{\partial\xi}\times\dfrac{\partial\boldsymbol{x}}{\partial\eta}}{\left\|\dfrac{\partial\boldsymbol{x}}{\partial\xi}\times\dfrac{\partial\boldsymbol{x}}{\partial\eta}\right\|_2}\cdot\delta\boldsymbol{u}\left\|\frac{\partial\boldsymbol{x}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta}\right\|_2\mathbf{d}\xi\wedge\mathbf{d}\eta \\
&= \int\limits_{\xi}\int\limits_{\eta} p_{ext}\left(\frac{\partial\boldsymbol{x}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta}\right)\cdot\delta\boldsymbol{u}\,\mathbf{d}\xi\wedge\mathbf{d}\eta \\
&= \int\limits_{\xi}\int\limits_{\eta} p_{ext}\boldsymbol{n}\left(\boldsymbol{u}\right)\cdot\delta\boldsymbol{u}\,\mathbf{d}\xi\wedge\mathbf{d}\eta
\end{aligned} \tag{6.190}$$

where $\boldsymbol{n}$ is the normal direction (non-normalised) *i.e.,*

$$\boldsymbol{n} = \frac{\partial\boldsymbol{x}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta} \tag{6.191}$$

The body force is due to the effect of gravity on the body *i.e.,*

$$\boldsymbol{b} = \rho\boldsymbol{g} \tag{6.192}$$

where ρ is the density of the current configuration and $\boldsymbol{g}$ is the acceleration vector due to gravity.

The external body work is thus given by

$$\begin{aligned} W_{body}\left(\boldsymbol{u}, \delta\boldsymbol{u}\right) &= \int\limits_{\mathcal{B}} \boldsymbol{b}\left(\boldsymbol{u}\right) \cdot \delta\boldsymbol{u}\,\mathbf{d}v \\ &= \int\limits_{\mathcal{B}} \rho\boldsymbol{g} \cdot \delta\boldsymbol{u}\,\mathbf{d}v \\ &= \int\limits_{\mathcal{B}_0} J\frac{\rho_0}{J}\boldsymbol{g} \cdot \delta\boldsymbol{u}\,\mathbf{d}V \\ &= \int\limits_{\mathcal{B}_0} \rho_0\boldsymbol{g} \cdot \delta\boldsymbol{u}\,\mathbf{d}V \end{aligned} \tag{6.193}$$

where ρ_0 is the density in the reference configuration.

$$c^{ijkl} = J^{-1}F^i_AF^j_BF^k_CF^l_DC^{ABCD} \tag{6.194}$$

If we now just consider the external traction part. For the case where the external traction is given by a fixed external pressure we have

$$\int\limits_{\partial\mathcal{B}_t} \bar{\boldsymbol{t}} \cdot \delta\boldsymbol{u}\,\mathbf{d}\boldsymbol{a} = \int\limits_{\partial\mathcal{B}_t} p_{ext}\hat{\boldsymbol{n}} \cdot \delta\boldsymbol{u}\,\mathbf{d}\boldsymbol{a} \tag{6.195}$$

where $\hat{\boldsymbol{n}}$ is the unit normal to the surface element $\mathbf{d}\boldsymbol{a}$. Now if the surface is parameterised by the coordinates ξ and η then

$$\mathbf{d}\boldsymbol{a} = \left\|\frac{\partial\boldsymbol{x}}{\partial\xi} \times \frac{\partial\boldsymbol{x}}{\partial\eta}\right\|_2 \mathbf{d}\xi \wedge \mathbf{d}\eta \tag{6.196}$$

and the unit normal is given by

$$\hat{\boldsymbol{n}} = \frac{\dfrac{\partial\boldsymbol{x}}{\partial\xi} \times \dfrac{\partial\boldsymbol{x}}{\partial\eta}}{\left\|\dfrac{\partial\boldsymbol{x}}{\partial\xi} \times \dfrac{\partial\boldsymbol{x}}{\partial\eta}\right\|_2} \tag{6.197}$$

We thus have

The directional derivative of $\delta W_{ext}\left(\boldsymbol{u}, p_{ext}, \delta\boldsymbol{u}\right)$ is given by

$$D\left(\delta W_{ext}\left(\boldsymbol{u}, p_{ext}, \delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}, \Delta p_{ext}\right] = \int\limits_{\xi}\int\limits_{\eta} \left(p_{ext}\boldsymbol{n}\cdot\delta\boldsymbol{u} + \Delta p_{ext}\boldsymbol{n}\cdot\delta\boldsymbol{u} + p_{ext}\Delta\boldsymbol{n}\cdot\delta\boldsymbol{u}\right) \mathbf{d}\xi\wedge\mathbf{d}\eta \tag{6.198}$$

For problems where p_{ext} is not part of the solution procedure then $\Delta p_{ext} = 0$ and thus

$$D\left(\delta W_{ext}\left(\boldsymbol{u}, p_{ext}, \delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] = \int\limits_{\xi}\int\limits_{\eta} \left(p_{ext}\boldsymbol{n}\cdot\delta\boldsymbol{u} + p_{ext}\Delta\boldsymbol{n}\cdot\delta\boldsymbol{u}\right) \mathbf{d}\xi\wedge\mathbf{d}\eta \tag{6.199}$$

The directional derivative of the normal vector is given by

$$D\left(\boldsymbol{n}\right)\left[\Delta\boldsymbol{u}\right] = \frac{\partial\Delta\boldsymbol{u}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta} - \frac{\partial\Delta\boldsymbol{u}}{\partial\eta}\times\frac{\partial\boldsymbol{x}}{\partial\xi} \tag{6.200}$$

and thus we have

$$D\left(\delta W_{ext}\left(\boldsymbol{u}, p_{ext}, \delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] = \int\limits_{\xi}\int\limits_{\eta} \left(p_{ext}\boldsymbol{n}\cdot\delta\boldsymbol{u} + p_{ext}\left(\frac{\partial\Delta\boldsymbol{u}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta} - \frac{\partial\Delta\boldsymbol{u}}{\partial\eta}\times\frac{\partial\boldsymbol{x}}{\partial\xi}\right)\cdot\delta\boldsymbol{u}\right) \mathbf{d}\xi\wedge\mathbf{d}\eta \tag{6.201}$$

The transformation rule for a triple product is

$$\boldsymbol{a}\cdot\left(\boldsymbol{b}\times\boldsymbol{c}\right) = \boldsymbol{b}\cdot\left(\boldsymbol{c}\times\boldsymbol{a}\right) = \boldsymbol{c}\cdot\left(\boldsymbol{a}\times\boldsymbol{b}\right) \tag{6.202}$$

and so we have

$$\begin{aligned} D\left(\delta W_{ext}\left(\boldsymbol{u}, p_{ext}, \delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] &= \int\limits_{\xi}\int\limits_{\eta} \left(p_{ext}\boldsymbol{n}\cdot\delta\boldsymbol{u} + p_{ext}\left(\frac{\partial\Delta\boldsymbol{u}}{\partial\xi}\times\frac{\partial\boldsymbol{x}}{\partial\eta} - \frac{\partial\Delta\boldsymbol{u}}{\partial\eta}\times\frac{\partial\boldsymbol{x}}{\partial\xi}\right)\cdot\delta\boldsymbol{u}\right) \mathbf{d}\xi\wedge\mathbf{d}\eta \\ &= p_{ext}\int\limits_{\xi}\int\limits_{\eta} \left(\boldsymbol{n}\cdot\delta\boldsymbol{u} + \left(\frac{\partial\boldsymbol{x}}{\partial\eta}\times\delta\boldsymbol{u}\right)\cdot\frac{\partial\Delta\boldsymbol{u}}{\partial\xi} - \left(\frac{\partial\boldsymbol{x}}{\partial\xi}\times\delta\boldsymbol{u}\right)\cdot\frac{\partial\Delta\boldsymbol{u}}{\partial\eta}\right) \mathbf{d}\xi\wedge\mathbf{d}\eta \end{aligned} \tag{6.203}$$

Now Equation (6.203) is, in general, non-symmetric in terms of $\Delta\boldsymbol{u}$ and $\delta\boldsymbol{u}$. This will equate to a non-conservative force.

Consider now the directional derivative of the incompressibility constraint *i.e.,*

$$D\left(\int_{\mathbf{0}}^{\mathbf{1}} \left(J\left(\delta\boldsymbol{u}\right)-1\right)\psi_m^{(N+1)\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(m,\alpha\right)J_{\mathcal{B}}\left(\boldsymbol{\xi}\right)\,\mathbf{d}\boldsymbol{\xi}\right)\left[\Delta\boldsymbol{u}\right]$$
$$=\mathrm{S}\left(m,\alpha\right)\int_{\mathbf{0}}^{\mathbf{1}}\left(J-1\right)\psi_m^{(N+1)\alpha}\left(\boldsymbol{\xi}\right)J_{\mathcal{B}}\left(\boldsymbol{\xi}\right)\,\mathbf{d}\boldsymbol{\xi} \tag{6.204}$$

6.2.4 Finite Element Formulation

Residual

The residual statement is given by

$$R=\delta\Pi\left(\boldsymbol{u},\delta\boldsymbol{u}\right)=0 \tag{6.205}$$

Firstly, consider the displacement, $\boldsymbol{u}$, between the reference/material coordinates, $\boldsymbol{x}$, and the current/spatial coordinates, $\boldsymbol{z}$. The displacement is defined by

$$\boldsymbol{u}=\boldsymbol{z}-\boldsymbol{x} \tag{6.206}$$

and thus

$$\begin{aligned}\delta\boldsymbol{u}&=\delta\left(\boldsymbol{z}-\boldsymbol{X}\right)\\&=\delta\boldsymbol{z}-\delta\boldsymbol{X}\\&=\delta\boldsymbol{z}\end{aligned} \tag{6.207}$$

as there is no variation in the original reference configuration $\boldsymbol{X}$.

The variational statement is

$$\int_{\mathcal{B}}\left(\boldsymbol{\sigma}+\boldsymbol{\sigma}_p\right):\nabla\delta\boldsymbol{u}\,\mathrm{d}v=\int_{\partial\mathcal{B}_t}\bar{\boldsymbol{t}}\cdot\delta\boldsymbol{u}\,\mathrm{d}a+\int_{\mathcal{B}}\boldsymbol{b}\cdot\delta\boldsymbol{u}\,\mathrm{d}v \tag{6.208}$$

or

$$\int\limits_{\mathcal{B}} \boldsymbol{\sigma} : \nabla \delta \boldsymbol{u} \,\mathbf{d}v + \int\limits_{\mathcal{B}} \boldsymbol{\sigma}_p : \nabla \delta \boldsymbol{u} \,\mathbf{d}v = \int\limits_{\partial \mathcal{B}_t} \bar{\boldsymbol{t}} \cdot \delta \boldsymbol{u} \,\mathbf{d}a + \int\limits_{\mathcal{B}} \boldsymbol{b} \cdot \delta \boldsymbol{u} \,\mathbf{d}v \tag{6.209}$$

In component form, we have

$$\int\limits_{\mathcal{B}} \sigma^{ij} \delta u_{j;i} \,\mathbf{d}v + \int\limits_{\mathcal{B}} \sigma_p^{ij} \delta u_{j;i} \,\mathbf{d}v = \int\limits_{\mathcal{B}} b^j \delta u_j \,\mathbf{d}v + \int\limits_{\partial \mathcal{B}} t^j \delta u_j \,\mathbf{d}a \tag{6.210}$$

The first integral on the left hand side of the virtual work statement is

$$\begin{aligned} \int\limits_{\mathcal{B}} \sigma^{ij} \delta u_{j;i} \,\mathbf{d}v &= \int\limits_{\mathcal{B}} \sigma^{ij} \left(\delta u_{j,i} - \Gamma^k_{ji} \delta u_k \right) \mathbf{d}v \\ &= \int\limits_{\mathcal{B}} \sigma^{ij} \left(\frac{\partial \delta u_j}{\partial x^i} - \Gamma^k_{ji} \delta u_k \right) \mathbf{d}v \end{aligned} \tag{6.211}$$

and thus the second integral on the left hand side is

$$\int\limits_{\mathcal{B}} \sigma_p^{ij} \delta u_{j;i} \,\mathbf{d}v = \int\limits_{\mathcal{B}} \sigma_p^{ij} \left(\frac{\partial \delta u_j}{\partial x^i} - \Gamma^k_{ji} \delta u_k \right) \mathbf{d}v \tag{6.212}$$

If we now substitute $\delta \boldsymbol{u} = \delta \boldsymbol{z}$ and convert the left hand side of the virtual work statement from an integral with respect to spatial coordinates to an integral with respect to $\boldsymbol{\xi}$ coordinates we obtain

$$\begin{aligned} \int\limits_{\mathcal{B}} \sigma^{ij} \left(\frac{\partial \delta u_j}{\partial x^i} - \Gamma^k_{ji} \delta u_k \right) \mathbf{d}v &= \int\limits_{\mathcal{B}} \sigma^{ij} \left(\frac{\partial \delta z_j}{\partial x^i} - \Gamma^k_{ji} \delta z_k \right) \mathbf{d}v \\ &= \int\limits_{\mathbf{0}}^{\mathbf{1}} \sigma^{ij} (\boldsymbol{\xi}) \left(\frac{\partial \xi_l}{\partial x^i} \frac{\partial \delta z_j (\boldsymbol{\xi})}{\partial \xi^l} - \Gamma^k_{ji} \delta z_k (\boldsymbol{\xi}) \right) J_{\mathcal{B}} (\boldsymbol{\xi}) \,\mathbf{d}\boldsymbol{\xi} \end{aligned} \tag{6.213}$$

Note that in rectangular cartesian coordinates $\Gamma^k_{ji} = 0 \; \forall i, j, k$. In addition it is not necessary to transform either the Cauchy stress tensor or gradient of the virtual displacements so that the components are with respect to $\boldsymbol{\xi}$ coordinates. What is important is that the stress and displacement are with respect to the same coordinate system. Because the gradient of $\delta\boldsymbol{z}$ is with respect to $\boldsymbol{x}$ coordinates then $\boldsymbol{\sigma}$ needs to be with respect to $\boldsymbol{x}$ coordinates. As there is no coordinate transformations the Christoffel symbols are all zero and can be dropped.

The second integral on the left hand side is thus

$$\int\limits_{\mathcal{B}} \sigma_p^{ij}\left(\frac{\partial \delta u_j}{\partial x^i} - \Gamma^k_{ji}\delta u_k\right)\mathbf{d}v = \int\limits_{\mathbf{0}}^{\mathbf{1}} \sigma_p^{ij}(\boldsymbol{\xi})\frac{\partial \xi_l}{\partial x^i}\frac{\partial \delta z_j(\boldsymbol{\xi})}{\partial \xi^l}J_{\mathcal{B}}(\boldsymbol{\xi})\,\mathbf{d}\boldsymbol{\xi} \tag{6.214}$$

The right hand side of the virtual work statement is

$$\begin{aligned}\int\limits_{\mathcal{B}} b^j\delta u_j\,\mathbf{d}v + \int\limits_{\partial\mathcal{B}_t} \bar{t}^j\delta u_j\,\mathbf{d}a &= \int\limits_{\mathcal{B}} b^j\delta z_j\,\mathbf{d}v + \int\limits_{\partial\mathcal{B}_t} t^j\delta z_j\,\mathbf{d}a + \int\limits_{\partial\mathcal{B}_P} Pn^j\delta z_j\,\mathbf{d}a\\ &= \int\limits_{\mathbf{0}}^{\mathbf{1}} b^j(\boldsymbol{\xi})\,\delta z_j(\boldsymbol{\xi})\,J_{\mathcal{B}}(\boldsymbol{\xi})\,\mathbf{d}\boldsymbol{\xi}\\ &\quad+\int\limits_{\mathbf{0}}^{\mathbf{1}} t^j(\boldsymbol{\xi})\,\delta z_j(\boldsymbol{\xi})\,J_{\mathcal{B}}(\boldsymbol{\xi})\,\mathbf{d}\boldsymbol{\xi}\\ &\quad+\int\limits_{\mathbf{0}}^{\mathbf{1}} P(\boldsymbol{\xi})\,n^j(\boldsymbol{\xi})\,\delta z_j(\boldsymbol{\xi})\,J_{\mathcal{B}}(\boldsymbol{\xi})\,\mathbf{d}\boldsymbol{\xi}\end{aligned} \tag{6.215}$$

where P is the applied surface pressure.

If we now use basis functions to interpolate the virtual displacements *i.e.,*

$$\delta z_j(\boldsymbol{\xi}) = \psi^{\alpha}_{jm}(\boldsymbol{\xi})\,\delta z^m_{j,\alpha}\mathrm{S}(m,\alpha) \tag{6.216}$$

which, assuming rectangular cartesian coordinates, gives for the first left hand side integral

$$\begin{aligned}\int_{\mathbf{0}}^{\mathbf{1}} \sigma^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \delta z_j(\boldsymbol{\xi})}{\partial \xi^l} J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} &= \int_{\mathbf{0}}^{\mathbf{1}} \sigma^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \left(\psi_{jm}^{\alpha}(\boldsymbol{\xi})\, \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha)\right)}{\partial \xi^l} J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \\ &= \int_{\mathbf{0}}^{\mathbf{1}} \sigma^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \psi_{jm}^{\alpha}(\boldsymbol{\xi})}{\partial \xi^l} \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha)\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \\ &= \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha) \int_{\mathbf{0}}^{\mathbf{1}} \sigma^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \psi_{m}^{j\alpha}(\boldsymbol{\xi})}{\partial \xi^l} J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi}\end{aligned} \tag{6.217}$$

and for the second left hand side integral

$$\int_{\mathbf{0}}^{\mathbf{1}} \sigma_p^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \delta z_j(\boldsymbol{\xi})}{\partial \xi^l} J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} = \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha) \int_{\mathbf{0}}^{\mathbf{1}} \sigma_p^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \psi_{m}^{j\alpha}(\boldsymbol{\xi})}{\partial \xi^l} J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \tag{6.218}$$

and for the first integral on the right hand side integral we have

$$\begin{aligned}\int_{\mathbf{0}}^{\mathbf{1}} b^j(\boldsymbol{\xi})\, \delta z_j(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} &= \int_{\mathbf{0}}^{\mathbf{1}} b^j(\boldsymbol{\xi})\, \psi_{jm}^{\alpha}(\boldsymbol{\xi})\, \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha)\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \\ &= \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha) \int_{\mathbf{0}}^{\mathbf{1}} b^j(\boldsymbol{\xi})\, \psi_{m}^{j\alpha}(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi}\end{aligned} \tag{6.219}$$

and for the second integral on the right hand side we have

$$\begin{aligned}\int_{\mathbf{0}}^{\mathbf{1}} t^j(\boldsymbol{\xi})\, \delta z_j(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} &= \int_{\mathbf{0}}^{\mathbf{1}} t^j(\boldsymbol{\xi})\, \psi_{jm}^{\alpha}(\boldsymbol{\xi})\, \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha)\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \\ &= \delta z_{j,\alpha}^{m} \mathrm{S}(m,\alpha) \int_{\mathbf{0}}^{\mathbf{1}} t^j(\boldsymbol{\xi})\, \psi_{m}^{j\alpha}(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi}\end{aligned} \tag{6.220}$$

and for the third integral on the right hand side we have

$$\begin{aligned}\int_{\mathbf{0}}^{\mathbf{1}} P(\boldsymbol{\xi})\, n^j(\boldsymbol{\xi})\, \delta z_j(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} &= \int_{\mathbf{0}}^{\mathbf{1}} P(\boldsymbol{\xi})\, n^j(\boldsymbol{\xi})\, \psi^{\alpha}_{jm}(\boldsymbol{\xi})\, \delta z^m_{j,\alpha} \mathrm{S}(m,\alpha)\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \\ &= \delta z^m_{j,\alpha} \mathrm{S}(m,\alpha) \int_{\mathbf{0}}^{\mathbf{1}} P(\boldsymbol{\xi})\, n^j(\boldsymbol{\xi})\, \psi^{j\alpha}_{m}(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi}\end{aligned} \tag{6.221}$$

This can be formulated as

$$r^{j\alpha}_m \delta z^m_{j,\alpha} = 0 \tag{6.222}$$

where the residual vector is thus given by

$$\begin{aligned} r^{j\alpha}_m = \mathrm{S}(m,\alpha) \Bigg(\int_{\mathbf{0}}^{\mathbf{1}} \left(\sigma^{ij}(\boldsymbol{\xi}) \frac{\partial \xi_l}{\partial x^i} \frac{\partial \psi^{j\alpha}_m(\boldsymbol{\xi})}{\partial \xi^l} + \rho \left(a^j(\boldsymbol{\xi}) - b^j(\boldsymbol{\xi}) \right) \psi^{j\alpha}_m(\boldsymbol{\xi}) \right) J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \\ - \int_{\mathbf{0}}^{\mathbf{1}} P(\boldsymbol{\xi})\, n^j(\boldsymbol{\xi})\, \psi^{j\alpha}_m(\boldsymbol{\xi})\, J_{\mathcal{B}}(\boldsymbol{\xi})\, \mathbf{d}\boldsymbol{\xi} \Bigg) \end{aligned} \tag{6.223}$$

Now, as the virtual displacements are arbitrary we have the residual statement

$$r^{j\alpha}_m = 0 \tag{6.224}$$

In order to handle incompressible materials we need an additional constraint which penalises change in volume. The change in volume is given by

$$\Delta V = \frac{\sqrt{\det \boldsymbol{g}}}{J_g \sqrt{\det \boldsymbol{G}}} \tag{6.225}$$

and the residual equation is

$$\begin{aligned} r_m^{(N+1)\alpha} &= \int_{\mathbf{0}}^{\mathbf{1}} \left(\Delta V\left(\boldsymbol{\xi}\right) - 1\right) \psi_m^{(N+1)\alpha}\left(\boldsymbol{\xi}\right) \mathrm{S}\left(m, \alpha\right) J_{\mathcal{B}}\left(\boldsymbol{\xi}\right) \, \mathbf{d}\boldsymbol{\xi} \\ &= \mathrm{S}\left(m, \alpha\right) \int_{\mathbf{0}}^{\mathbf{1}} \left(\frac{\sqrt{\det \boldsymbol{g}\left(\boldsymbol{\xi}\right)}}{J_g\left(\boldsymbol{\xi}\right) \sqrt{\det \boldsymbol{G}\left(\boldsymbol{\xi}\right)}} - 1 \right) \psi_m^{(N+1)\alpha}\left(\boldsymbol{\xi}\right) J_{\mathcal{B}}\left(\boldsymbol{\xi}\right) \, \mathbf{d}\boldsymbol{\xi} \end{aligned} \tag{6.226}$$

where N is the number of dimensions.

Jacobian

In order to solve the nonlinear system of equations a Newton scheme can be used. To calculate the Jacobian of the system we need to calculate the variation of the virtual work statement. The Jacobian is given by the derivative of the residual equation in the direction of a change in $\boldsymbol{u}$, $\Delta \boldsymbol{u}$ *i.e.,* the Lie derivative of the residual vector in the direction $\Delta \boldsymbol{u}$.

This requires a linerization. A linearisation of a function $f\left(\boldsymbol{x}\right)$ in the direction of $\Delta \boldsymbol{x}$ is

$$L_{\boldsymbol{x}}\left(f\right)\left[\Delta \boldsymbol{x}\right] = \frac{d}{d\epsilon} f\left(\boldsymbol{x} + \epsilon \Delta \boldsymbol{x}\right)\Big|_{\epsilon=0} = f\left(\boldsymbol{x}\right) + D\left(f\left(\boldsymbol{x}\right)\right)\left[\Delta \boldsymbol{x}\right] \tag{6.227}$$

The linearization of the residual equation is given by

$$\begin{aligned} D\left(\delta \Pi\left(\boldsymbol{u}, \delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] &= D\left(\delta W_{int}\left(\boldsymbol{u}, \delta \boldsymbol{u}\right) - \delta W_{ext}\left(\boldsymbol{u}, \delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] \\ &= D\left(\delta W_{int}\left(\boldsymbol{u}, \delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] - D\left(\delta W_{ext}\left(\boldsymbol{u}, \delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] \end{aligned} \tag{6.228}$$

For the directional derivative of the internal work variation it is useful to consider the internal work in terms of second Piola-Kirchoff stress and the Green-Lagrange strain *i.e.,*

$$\begin{aligned} D\left(\delta W_{int}\left(\boldsymbol{u}, \delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] &= D\left(\int_{\mathcal{B}_0} \boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right) : \delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right) \, \mathrm{d}V \right)\left[\Delta \boldsymbol{u}\right] \\ &= \int_{\mathcal{B}_0} \left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right) : D\left(\delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] + D\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta \boldsymbol{u}\right] : \delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right) \right) \mathrm{d}V \end{aligned} \tag{6.229}$$

(SPLIT T INTO DEVIATORIC PART PLUS HYDROSTATIC STRESS)

Now, $\delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right)$ can be found by pulling back $\delta \boldsymbol{\epsilon}\left(\delta \boldsymbol{u}\right)$ *i.e.,*

$$\begin{aligned} \delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right) &= \boldsymbol{F}^T \delta \boldsymbol{\epsilon}\left(\delta \boldsymbol{u}\right) \boldsymbol{F} \\ &= \boldsymbol{F}^T \frac{1}{2}\left(\left(\nabla_{\boldsymbol{x}} \delta \boldsymbol{u}\right)^T + \nabla_{\boldsymbol{x}} \delta \boldsymbol{u}\right) \boldsymbol{F} \\ &= \frac{1}{2}\left(\boldsymbol{F}^T \left(\nabla_{\boldsymbol{x}} \delta \boldsymbol{u}\right)^T \boldsymbol{F} + \boldsymbol{F}^T \nabla_{\boldsymbol{x}} \delta \boldsymbol{u} \boldsymbol{F}\right) \\ &= \frac{1}{2}\left(\left(\nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right)^T \boldsymbol{F} + \boldsymbol{F}^T \nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right) \\ &= \operatorname{sym}\left(\boldsymbol{F}^T \nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right) \end{aligned} \tag{6.230}$$

Now, recall that

$$\boldsymbol{F} = \boldsymbol{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}} = \boldsymbol{I} + \nabla_{\boldsymbol{X}} \boldsymbol{u} \tag{6.231}$$

and so

$$D\left(\boldsymbol{F}\right)\left[\Delta \boldsymbol{u}\right] = \nabla_{\boldsymbol{X}} \Delta \boldsymbol{u} \tag{6.232}$$

We thus have

$$\begin{aligned} D\left(\delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] &= D\left(\operatorname{sym}\left(\boldsymbol{F}^T \nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] \\ &= \operatorname{sym}\left(\boldsymbol{F}^T D\left(\nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right)\left[\Delta \boldsymbol{u}\right] + \left(D\left(\boldsymbol{F}\right)\left[\Delta \boldsymbol{u}\right]\right)^T \nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right) \\ &= \operatorname{sym}\left(0 + \left(\nabla_{\boldsymbol{X}} \Delta \boldsymbol{u}\right)^T \nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right) \\ &= \operatorname{sym}\left(\left(\nabla_{\boldsymbol{X}} \Delta \boldsymbol{u}\right)^T \nabla_{\boldsymbol{X}} \delta \boldsymbol{u}\right) \end{aligned} \tag{6.233}$$

as $\delta \boldsymbol{u}$ is independent of $\boldsymbol{u}$ and is this unaffected by the directional derivative (CHECK).

We also have

$$D\left(\boldsymbol{E}\left(\delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] = \operatorname{sym}\left(\left(\nabla_{\boldsymbol{X}} \Delta \boldsymbol{u}\right)^T \boldsymbol{F}\right) \tag{6.234}$$

(DERIVE).

Now

$$D\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta \boldsymbol{u}\right] = \frac{\partial \boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)}{\partial \boldsymbol{E}\left(\boldsymbol{u}\right)} : D\left(\delta \boldsymbol{E}\left(\delta \boldsymbol{u}\right)\right)\left[\Delta \boldsymbol{u}\right] \tag{6.235}$$

The derivative of the stress tensor with respect to the strain tensor is the fourth order *elas-*

ticity tensor (sometimes called the *stiffness tensor*) *i.e.,*

$$\mathsf{C} = \frac{\partial \boldsymbol{S}(\boldsymbol{E})}{\partial \boldsymbol{E}} \tag{6.236}$$

or in component form

$$C^{ABCD} = \frac{\partial S^{AB}}{\partial E_{CD}} \tag{6.237}$$

Thus we have

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u})) : D(\delta\boldsymbol{E}(\delta\boldsymbol{u}))[\Delta\boldsymbol{u}] &= \boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u})) : \operatorname{sym}\left((\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u})^T\nabla_{\boldsymbol{X}}\delta\boldsymbol{u}\right) \\ &= \boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u})) : (\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u})^T\nabla_{\boldsymbol{X}}\delta\boldsymbol{u} \\ &= \nabla_{\boldsymbol{X}}\delta\boldsymbol{u} : \nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u})) \end{aligned} \tag{6.238}$$

as $\boldsymbol{A} : \boldsymbol{B}^T\boldsymbol{C} = \boldsymbol{C} : \boldsymbol{B}\boldsymbol{A}$, and

$$\begin{aligned} D(\boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u})))[\Delta\boldsymbol{u}] : \delta\boldsymbol{E}(\delta\boldsymbol{u}) &= \frac{\partial \boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u}))}{\partial \boldsymbol{E}(\boldsymbol{u})} : D(\delta\boldsymbol{E}(\delta\boldsymbol{u}))[\Delta\boldsymbol{u}] : \delta\boldsymbol{E}(\delta\boldsymbol{u}) \\ &= \mathsf{C} : \operatorname{sym}\left(\boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\right) : \operatorname{sym}\left(\boldsymbol{F}^T\nabla_{\boldsymbol{X}}\delta\boldsymbol{u}\right) \\ &= \boldsymbol{F}^T\nabla_{\boldsymbol{X}}\delta\boldsymbol{u} : \mathsf{C} : \boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u} \end{aligned} \tag{6.239}$$

as $\boldsymbol{A} : \boldsymbol{B} = \boldsymbol{B} : \boldsymbol{A}$.

Putting this together gives

$$D(\delta W_{int}(\boldsymbol{u}, \delta\boldsymbol{u}))[\Delta\boldsymbol{u}] = \int\limits_{\mathcal{B}_0} \left(\nabla_{\boldsymbol{X}}\delta\boldsymbol{u} : \nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\boldsymbol{S}(\boldsymbol{E}(\boldsymbol{u})) + \boldsymbol{F}^T\nabla_{\boldsymbol{X}}\delta\boldsymbol{u} : \mathsf{C} : \boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\right) \mathbf{d}V \tag{6.240}$$

or, in component form,

$$\begin{aligned} D(\delta W_{int}(u_i, \delta u_i))[\Delta u_j] &= \int\limits_{\mathcal{B}_0} \left(\frac{\partial \delta u_i}{\partial X^B}\frac{\partial \Delta u_j}{\partial X^D}\delta^{ij}T^{BD} + F^i_A\frac{\partial \delta u_i}{\partial X^B}C^{ABCD}F^j_C\frac{\partial \Delta u_j}{\partial X^D}\right) \mathbf{d}V \\ &= \int\limits_{\mathcal{B}_0} \left(\frac{\partial \delta u_i}{\partial X^B}\left[\delta^{ij}T^{BD} + F^i_AF^j_CC^{ABCD}\right]\frac{\partial \Delta u_j}{\partial X^D}\right) \mathbf{d}V \end{aligned} \tag{6.241}$$

Now, consider the directional derivative of the internal work variation in the current configuration *i.e.,* with respect to cauchy stress and ??? strain

$$\begin{aligned} D\left(\delta W_{int}\left(\boldsymbol{u},\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] &= D\left(\int_{\mathcal{B}_0} \boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right) : \delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\,\mathrm{d}V\right)\left[\Delta\boldsymbol{u}\right] \\ &= \int_{\mathcal{B}_0} \left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right) : D\left(\delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] + D\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right] : \delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\right)\mathrm{d}V \end{aligned} \tag{6.242}$$

We can now use a push forward operation to find the directional derivative of the Cauchy stress in terms of the results from the directional derivative of the second Piola-Kirchoff stress *i.e.,*

$$\begin{aligned} \boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right) &= J^{-1}\chi_*\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right) \\ D\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right] &= J^{-1}\chi_*\left(D\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right]\right) \\ D\left(\delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] &= \chi_*\left(D\left(\delta\boldsymbol{E}\left(\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right]\right) \\ \delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right) &= \chi_*\left(\delta\boldsymbol{E}\left(\delta\boldsymbol{u}\right)\right) \end{aligned} \tag{6.243}$$

Now,

$$\begin{aligned} D\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right] &= J^{-1}\chi_*\left(D\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right]\right) \\ &= J^{-1}\chi_*\left(\mathbf{C} : \boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\right) \\ &= J^{-1}\boldsymbol{F}\left(\mathbf{C} : \boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\right)\boldsymbol{F}^T \end{aligned} \tag{6.244}$$

The push forward of the fourth order elasticity tensor is

$$\mathbf{c} = J^{-1}\chi_*\left(\mathbf{C}\right) \tag{6.245}$$

In component form this is

$$c^{ijkl} = J^{-1}F^i_AF^j_BF^k_CF^l_DC^{ABCD} \tag{6.246}$$

Now, consider the directional derivative of the internal work variation in the current configuration *i.e.,* with respect to cauchy stress and ??? strain

$$\begin{aligned} D\left(\delta W_{int}\left(\boldsymbol{u},\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] &= D\left(\int\limits_{\mathcal{B}_0}\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right):\delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\,\mathrm{d}V\right)\left[\Delta\boldsymbol{u}\right] \\ &= \int\limits_{\mathcal{B}_0}\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right):D\left(\delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right]+D\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right]:\delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\right)\mathrm{d}V \end{aligned} \tag{6.247}$$

We can now use a push forward operation to find the directional derivative of the Cauchy stress in terms of the results from the directional derivative of the second Piola-Kirchoff stress *i.e.,*

$$\begin{aligned} \boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right) &= J^{-1}\chi_*\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right) \\ D\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right] &= J^{-1}\chi_*\left(D\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right]\right) \\ D\left(\delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right] &= \chi_*\left(D\left(\delta\boldsymbol{E}\left(\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right]\right) \\ \delta\boldsymbol{\epsilon}\left(\delta\boldsymbol{u}\right) &= \chi_*\left(\delta\boldsymbol{E}\left(\delta\boldsymbol{u}\right)\right) \end{aligned} \tag{6.248}$$

Now,

$$\begin{aligned} D\left(\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right] &= J^{-1}\chi_*\left(D\left(\boldsymbol{S}\left(\boldsymbol{E}\left(\boldsymbol{u}\right)\right)\right)\left[\Delta\boldsymbol{u}\right]\right) \\ &= J^{-1}\chi_*\left(\mathsf{C}:\boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\right) \\ &= J^{-1}\boldsymbol{F}\left(\mathsf{C}:\boldsymbol{F}^T\nabla_{\boldsymbol{X}}\Delta\boldsymbol{u}\right)\boldsymbol{F}^T \end{aligned} \tag{6.249}$$

The push forward of the fourth order elasticity tensor is

$$\mathsf{c} = J^{-1}\chi_*\left(\mathsf{C}\right) \tag{6.250}$$

In component form this is

$$c^{ijkl} = J^{-1}F^i_AF^j_BF^k_CF^l_DC^{ABCD} \tag{6.251}$$

Putting this together gives

$$D\left(\delta W_{int}\left(\boldsymbol{u},\delta\boldsymbol{u}\right)\right)\left[\Delta\boldsymbol{u}\right]=\int\limits_{\mathcal{B}}\left(\nabla_{\boldsymbol{x}}\delta\boldsymbol{u}:\nabla_{\boldsymbol{x}}\Delta\boldsymbol{u}\boldsymbol{\sigma}\left(\boldsymbol{e}\left(\boldsymbol{u}\right)\right)+\nabla_{\boldsymbol{x}}\delta\boldsymbol{u}:\mathbf{c}:\nabla_{\boldsymbol{x}}\Delta\boldsymbol{u}\right)\mathbf{d}v \quad (6.252)$$

or, in component form,

$$\begin{aligned}D\left(\delta W_{int}\left(u_i,\delta u_i\right)\right)\left[\Delta u_j\right]&=\int\limits_{\mathcal{B}}\left(\frac{\partial\delta u_i}{\partial x^k}\frac{\partial\Delta u_j}{\partial x^l}\delta^{ij}\sigma^{kl}+\frac{\partial\delta u_i}{\partial x^k}c^{ikjl}\frac{\partial\Delta u_j}{\partial x^l}\right)\mathbf{d}v\\&=\int\limits_{\mathcal{B}}\left(\frac{\partial\delta u_i}{\partial x^k}\left[\delta^{ij}\sigma^{kl}+c^{ikjl}\right]\frac{\partial\Delta u_j}{\partial x^l}\right)\mathbf{d}v\end{aligned} \quad (6.253)$$

6.2.5 Constituative Laws

The constituative law can then be used to derive the second Piola Kirchhoff stress tensor in fibre coordinates, $\boldsymbol{S}\left(\boldsymbol{\nu}\right)$, from either the right Cauchy-Green deformation tensor or the Green-Lagrange strain tensor *i.e.,*

$$\boldsymbol{S}\left(\boldsymbol{\nu}\right)=2\frac{\partial W\left(\boldsymbol{C}\left(\boldsymbol{\nu}\right)\right)}{\partial\boldsymbol{C}\left(\boldsymbol{\nu}\right)} \quad (6.254)$$

or

$$\boldsymbol{S}\left(\boldsymbol{\nu}\right)=\frac{\partial W\left(\boldsymbol{E}\left(\boldsymbol{\nu}\right)\right)}{\partial\boldsymbol{E}\left(\boldsymbol{\nu}\right)} \quad (6.255)$$

where $W\left(\boldsymbol{C}\left(\boldsymbol{\nu}\right)\right)$ or $W\left(\boldsymbol{E}\left(\boldsymbol{\nu}\right)\right)$ is the strain energy function. In component form we have

$$S^{AB}=2\frac{\partial W}{\partial C_{AB}} \quad (6.256)$$

or

$$S^{AB}=\frac{\partial W}{\partial E_{AB}} \quad (6.257)$$

Because $\boldsymbol{C}$ is symmetric then we can deal with the invariants. The three invariants are

$$\begin{aligned}
I_1 &= \operatorname{tr} \boldsymbol{C} \\
&= C_{11} + C_{22} + C_{33} \\
I_2 &= \frac{1}{2}\left((\operatorname{tr} \boldsymbol{C})^2 - \operatorname{tr} \boldsymbol{C}^2\right) \\
&= \frac{1}{2}\left((C_{11} + C_{22} + C_{33})^2\right. \\
&\quad \left. - \left(C_{11}^2 + C_{12}C_{21} + C_{13}C_{31} + C_{21}C_{12} + C_{22}^2 + C_{23}C_{32} + C_{31}C_{13} + C_{32}C_{23} + C_{33}^2\right)\right) \\
I_3 &= \det \boldsymbol{C} \\
&= C_{11}C_{22}C_{33} + C_{12}C_{23}C_{31} + C_{13}C_{21}C_{32} \\
&\quad - C_{13}C_{22}C_{31} - C_{12}C_{21}C_{33} - C_{11}C_{23}C_{32}
\end{aligned} \tag{6.258}$$

We thus have $W\left(\boldsymbol{C}\left(\boldsymbol{\nu}\right)\right) = W\left(I_1, I_2, I_3\right)$ and thus

$$S^{AB} = 2\left(\frac{\partial W}{\partial I_1}\frac{\partial I_1}{\partial C_{AB}} + \frac{\partial W}{\partial I_2}\frac{\partial I_2}{\partial C_{AB}} + \frac{\partial W}{\partial I_3}\frac{\partial I_3}{\partial C_{AB}}\right) \tag{6.259}$$

or if we have $W\left(\boldsymbol{E}\left(\boldsymbol{\nu}\right)\right) = W\left(I_1, I_2, I_3\right)$ and thus

$$S^{AB} = \left(\frac{\partial W}{\partial I_1}\frac{\partial I_1}{\partial E_{AB}} + \frac{\partial W}{\partial I_2}\frac{\partial I_2}{\partial E_{AB}} + \frac{\partial W}{\partial I_3}\frac{\partial I_3}{\partial E_{AB}}\right) \tag{6.260}$$

Now we have

$$\frac{\partial I_1}{\partial C_{AB}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.261}$$

and

$$\frac{\partial I_2}{\partial C_{AB}} = \begin{bmatrix} C_{22} + C_{33} & -C_{21} & -C_{31} \\ -C_{12} & C_{11} + C_{33} & -C_{32} \\ -C_{13} & -C_{23} & C_{11} + C_{22} \end{bmatrix} \tag{6.262}$$

and

$$\frac{\partial I_3}{\partial C_{AB}} = \begin{bmatrix} C_{22}C_{33} - C_{23}C_{32} & C_{23}C_{31} - C_{21}C_{33} & C_{23}C_{32} - C_{22}C_{31} \\ C_{13}C_{32} - C_{12}C_{33} & C_{11}C_{33} - C_{13}C_{31} & C_{12}C_{31} - C_{11}C_{32} \\ C_{12}C_{32} - C_{22}C_{31} & C_{13}C_{23} - C_{11}C_{23} & C_{11}C_{22} - C_{12}C_{21} \end{bmatrix} \tag{6.263}$$

Mooney-Rivlin

As an example consider a Mooney-Rivlin material. The strain energy function is given by

$$W(I_1, I_2) = c_1(I_1 - 3) + c_2(I_2 - 3) \tag{6.264}$$

The second Piola Kirchhoff tensor is thus

$$S^{AB} = \begin{bmatrix} 2c_1 + 2c_2(C_{22} + C_{33}) & -2c_2C_{21} & -2c_2C_{31} \\ -2c_2C_{12} & 2c_1 + 2c_2(C_{11} + C_{33}) & -2c_2C_{32} \\ -2c_2C_{13} & -2c_2C_{23} & 2c_1 + 2c_2(C_{11} + C_{22}) \end{bmatrix} \tag{6.265}$$

or

$$S^{AB} = \begin{bmatrix} c_1 + c_2(E_{22} + E_{33}) & -c_2E_{21} & -c_2E_{31} \\ -c_2E_{12} & c_1 + c_2(E_{11} + E_{33}) & -c_2E_{32} \\ -c_2E_{13} & -c_2E_{23} & c_1 + c_2(E_{11} + E_{22}) \end{bmatrix} \tag{6.266}$$

6.3 Uniaxial extension of a unit cube

Consider a uniaxial stretch of α of a unit cube in the x direction. The resulting deformed configuration will be a cube of dimensions X, Y and Z.

Here the deformation gradient tensor will be

$$\boldsymbol{F} = \begin{bmatrix} 1 + \alpha & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.267}$$

everywhere, and the Jacobian of transformation will be

$$J = \det \boldsymbol{F} = 1 + \alpha \tag{6.268}$$

The right Cauchy-Green tensor will be

$$\boldsymbol{C} = \boldsymbol{F}^T \boldsymbol{F} = \begin{bmatrix} 1+\alpha & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1+\alpha & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} (1+\alpha)^2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.269}$$

And the Lagrange strain tensor will be

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{C} - \boldsymbol{I}) = \frac{1}{2}\left(\begin{bmatrix} (1+\alpha)^2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right) = \begin{bmatrix} \alpha + \frac{\alpha^2}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.270}$$

The Finger strain tensor is thus

$$\boldsymbol{f} = \boldsymbol{C}^{-1} = \begin{bmatrix} \frac{1}{(1+\alpha)^2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6.271}$$

The second Piola-Kirchoff stress tensor will be

$$\begin{aligned} \boldsymbol{S} &= \begin{bmatrix} 2c_1 + 2c_2(C_{22} + C_{33}) & -2c_2 C_{21} & -2c_2 C_{31} \\ -2c_2 C_{12} & 2c_1 + 2c_2(C_{11} + C_{33}) & -2c_2 C_{32} \\ -2c_2 C_{13} & -2c_2 C_{23} & 2c_1 + 2c_2(C_{11} + C_{22}) \end{bmatrix} \\ &= \begin{bmatrix} 2c_1 + 4c_2 & 0 & 0 \\ 0 & 2c_1 + 2c_2\left((1+\alpha)^2 + 1\right) & 0 \\ 0 & 0 & 2c_1 + 2c_2\left((1+\alpha)^2 + 1\right) \end{bmatrix} \end{aligned} \tag{6.272}$$

and the Kirchoff stress tensor will be

$$\begin{aligned}\boldsymbol{\tau} &= \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T \\ &= \begin{bmatrix}1+\alpha & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}2c_1+4c_2 & 0 & 0\\ 0 & 2c_1+2c_2\left((1+\alpha)^2+1\right) & 0\\ 0 & 0 & 2c_1+2c_2\left((1+\alpha)^2+1\right)\end{bmatrix}\begin{bmatrix}1+\alpha & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix} \\ &= \begin{bmatrix}(2c_1+4c_2)(1+\alpha)^2 & 0 & 0\\ 0 & 2c_1+2c_2\left((1+\alpha)^2+1\right) & 0\\ 0 & 0 & 2c_1+2c_2\left((1+\alpha)^2+1\right)\end{bmatrix}\end{aligned} \tag{6.273}$$

The Cauchy stress tensor is thus

$$\boldsymbol{\sigma} = \frac{\boldsymbol{\tau}}{J} = \begin{bmatrix}(2c_1+4c_2)(1+\alpha) & 0 & 0\\ 0 & \frac{2c_1+2c_2\left((1+\alpha)^2+1\right)}{1+\alpha} & 0\\ 0 & 0 & \frac{2c_1+2c_2\left((1+\alpha)^2+1\right)}{1+\alpha}\end{bmatrix} \tag{6.274}$$

The Cauchy tensor resulting from hydrostatic pressure is

$$\boldsymbol{\sigma}_p = p\boldsymbol{I} = \begin{bmatrix}p & 0 & 0\\ 0 & p & 0\\ 0 & 0 & p\end{bmatrix} \tag{6.275}$$

6.3.1 Old Stuff

Formulation of finite element equations for finite elasticity (large deformation mechanics) implemented in OpenCMISS is based on the ***principle of virtual work***. The finite element model consists of a set of non-linear algebraic equations. Non-linearity of equations stems from non-linear stress-strain relationship and quadratic terms present in the strain tensor. A typical problem in large deformation mechanics involves determination of the deformed geometry or mesh nodal parameters, from the finite element point of view, of the continuum from a known undeformed geometry, subject to boundary conditions and satisfying stress-strain (constitutive) relationship.

The boundary conditions can be either ***Dirichlet*** (displacement), ***Neumann*** (force) or a combination of them, known as the mixed boundary conditions. Displacement boundary conditions are generally nodal based. However, force boundary conditions can take any of the following forms or a combination of them - nodal-based, distributed load (e.g. pressure) or force acting at a discrete point on the boundary. In the latter two forms, the equivalent nodal forces are determined using the ***method of work equivalence*** (**?**) and the forces so obtained will then be added to the right hand side or the residual vector of the linear equation system.

There are a numerous ways of describing the mechanical characteristics of deformable materials in large deformation mechanics or finite elasticity analyses. A predominantly used form for representing constitutive properties is a strain energy density function. This model gives the energy required to deform a unit volume (hence energy density) of the deformable continuum as a function of Green-Lagrange strain tensor components or its derived variables such as invariants or principal stretches. A material that has a strain energy density function is known as a ***hyperelastic*** or ***Green-elastic material***.

The deformed equilibrium state should also give the minimum total elastic potential energy. One can therefore formulate finite element equations using the ***Variational method*** approach where an extremum of a functional (in this case total strain energy) is determined to obtain mesh nodal parameters of the deformed continuum. It is also possible to derive the finite element equations starting from the governing equilibrium equations known as Cauchy equation of motion. The weak form of the governing equations is obtained by multiplying them with suitable weighting functions and integrating over the domain (method of weighted residuals). If interpolation or shape functions are used as weighting functions, then the method is called the Galerkin finite element method. All three approaches (virtual work, variational method and Galerkin formulation) result in the same finite element equations.

In the following sections the derivation of kinematic relationships of deformation, energy conjugacy, constitutive relationships and final form the finite element equations using the virtual work approach will be discussed in detail.

Kinematics of Deformation

In order to track the deformation of an infinitesimal length at a particle of the continuum, two coordinates systems are defined. An arbitrary orthogonal spatial coordinate system, which is fixed in space and a material coordinate system which is attached to the continuum and deforms with the continuum. The material coordinate system, in general, is a curvi-linear coordinate system but must have mutually orthogonal axes at the undeformed state. However, in the deformed state, these axes are no longer orthogonal as they deform with the continuum (fig 1). In addition to these coordinate systems, there exist finite element coordinate systems (one for each element) as well. These coordinates are normalised and vary from 0.0 to 1.0. The following notations are used to represent various coordinate systems and coordinates of a particle of the continuum.

Y_1-Y_2-Y_3 - fixed spatial coordinate system axes - orthogonal
N_1-N_2-N_3 - deforming material coordinate system axes - orthogonal in the undeformed state
Ξ_1-Ξ_2-Ξ_3 - element coordinate system - non-orthogonal in general and deforms with continuum

x_1-x_2-x_3 $[\boldsymbol{x}]$ - spatial coordinates of a particle in the undeformed state wrt Y_1-Y_2-Y_3 CS
z_1-z_2-z_3 $[\boldsymbol{z}]$ - spatial coordinates of the same particle in the deformed state wrt Y_1-Y_2-Y_3 CS
ν_1-ν_2-ν_3 $[\boldsymbol{\nu}]$ - material coordinates of the particle wrt N_1-N_2-N_3 CS (these do not change)
ξ_1-ξ_2-ξ_3 $[\boldsymbol{\xi}]$ - element coordinates of the particle wrt Ξ_1-Ξ_2-Ξ_3 CS (these too do not change)

Since the directional vectors of the material coordinate system at any given point in the undeformed state is mutually orthogonal, the relationship between spatial $\boldsymbol{x}$ and material $\boldsymbol{\nu}$ coordinates is simply a rotation. The user must define the undeformed material coordinate system. Typically a nodal based interpolatable field known as fibre information (fibre, imbrication and sheet angles) is input to OpenCMISS. These angles define how much the ***reference or default material coordinate system*** must be rotated about the reference material axes. The reference material coordinate system at a given point is defined as follows. The first direction ν_1 is in the ξ_1 direction. The second direction, ν_2 is in the $\xi_1 - \xi_2$ plane but orthogonal to ν_1. Finally the third direction ν_3 is determined to be normal to both ν_1 and ν_2. Once the reference coordinate system is defined, it is then rotated about ν_3 by an angle equal to the interpolated fibre value at the point in counter-clock wise direction. This will be followed by a rotation about new ν_2 axis again in the counter-clock wise direction by an angle equal to the sheet value. The final rotation is performed about the current ν_1 by an angle defined by interpolated sheet value. Note that before a rotation is carried out about an arbitrary axis one must first align(transform) the axis of rotation with one of the spatial coordinate system axes. Once the rotation is done, the rotated coordinate system (material) must be inverse-transformed.

Having defined the undeformed orthogonal material coordinate system, the metric tensor $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}$ can be determined. As mentioned, the tensor $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}$ contains rotation required to align material coordinate system with spatial coordinate system. This tensor is therefore orthogonal. A similar metric tensor can be defined to relate the deformed coordinates $\boldsymbol{z}$ of the point to its material coordinates $\boldsymbol{\nu}$. Note that the latter coordinates do not change as the continuum deforms and more importantly this tensor is not orthogonal as well. The metric tensor, $\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}}$ is called the ***deformation gradient tensor*** and denoted as $\boldsymbol{F}$.

$$\boldsymbol{F} = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}} \tag{6.276}$$

It can be shown that the deformation gradient tensor contains rotation when an infinitesimal length $\boldsymbol{dr}_0$ in the undeformed state undergoes deformation. Since rotation does not contribute to any strain, it must be removed from the deformation gradient tensor. Any tensor can be decomposed into an orthogonal tensor and a symmetric tensor (known as polar decomposition). In other words, the same deformation can be achieved by first rotating $\boldsymbol{dr}$ and then stretching (shearing and scaling) or vice-verse. Thus, the deformation gradient tensor can be given by,

$$\boldsymbol{F} = \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}} = \boldsymbol{RU} = \boldsymbol{VR}_1 \tag{6.277}$$

The rotation present in the deformation gradient tensor can be removed either by right or left multiplication of $\boldsymbol{F}$. The resulting tensors lead to different strain measures. The right Cauchy deformation tensor $\boldsymbol{C}$ is obtained from,

$$\boldsymbol{C} = [\boldsymbol{RU}]^T[\boldsymbol{RU}] = \boldsymbol{U}^T\boldsymbol{R}^T\boldsymbol{RU} = \boldsymbol{U}^T\boldsymbol{U} \tag{6.278}$$

Similarly the left Cauchy deformation tensor or the Finger tensor $\boldsymbol{B}$ is obtained from the left multiplication of $\boldsymbol{F}$,

$$\boldsymbol{B} = [\boldsymbol{VR}_1][\boldsymbol{VR}_1]^T = \boldsymbol{VR}_1{\boldsymbol{R}_1}^T\boldsymbol{V}^T = \boldsymbol{VV}^T \tag{6.279}$$

Note that both $\boldsymbol{R}$ and $\boldsymbol{R}_1$ are orthogonal tensors and therefore satisfy the following condition,

$$\boldsymbol{R}^T\boldsymbol{R} = \boldsymbol{R}_1{\boldsymbol{R}_1}^T = \boldsymbol{I} \tag{6.280}$$

Since there is no rotation present in both $\boldsymbol{C}$ and $\boldsymbol{B}$, they can be used to define suitable strain measures as follows,

$$\boldsymbol{E} = \frac{1}{2}\left(\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}}^T\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}} - \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}^T\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}\right) = \frac{1}{2}(\boldsymbol{C} - \boldsymbol{I}) \tag{6.281}$$

and

$$\boldsymbol{e} = \frac{1}{2}\left\{\left(\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}^T\right)^{-1} - \left(\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}}\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}}^T\right)^{-1}\right\} = \frac{1}{2}\left(\boldsymbol{I} - \boldsymbol{B}^{-1}\right) \tag{6.282}$$

where $\boldsymbol{E}$ and $\boldsymbol{e}$ are called Green and Almansi strain tensors respectively. Also note that $\dfrac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}$ is an orthogonal tensor.

It is now necessary to establish a relationship between strain and displacement. Referring to figure 1,

$$\boldsymbol{z} = \boldsymbol{x} + \boldsymbol{u} \tag{6.283}$$

where $\boldsymbol{u}$ is the displacement vector.

Differentiating Equation (6.283) using the chain rule,

$$\frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\nu}} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}} = \left(\boldsymbol{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\right)\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}} \tag{6.284}$$

Substituting Equation (6.284) into Equation (6.281),

$$\boldsymbol{E} = \frac{1}{2}\left\{\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}^T\left(\boldsymbol{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\right)^T\left(\boldsymbol{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\right)\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}} - \boldsymbol{I}\right\} \tag{6.285}$$

Simplifying,

$$\boldsymbol{E} = \frac{1}{2}\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}}^T\left(\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}^T + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}^T\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}\right)\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\nu}} \tag{6.286}$$

As can be seen from Equation (6.286) the displacement gradient tensor $\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}$ is defined with respect to undeformed coordinates $\boldsymbol{x}$. This means that the strain tensor $\boldsymbol{E}$ has Lagrangian description and hence it is also also called the Green-Lagrange strain tensor.

A similar derivation can be employed to establish a relationship between the Almansi and displacement gradient tensors and the final form is given by,

$$\boldsymbol{e} = \frac{1}{2}\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{z}} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{z}}^T - \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{z}}^T \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{z}} \tag{6.287}$$

The displacement gradient tensor terms in Equation (6.287) are defined with respect to deformed coordinates $\boldsymbol{z}$ and therefore the strain tensor has Eulerian description. Thus it is also known as the Almansi-Euler strain tensor.

Energy Conjugacy

Constitutive models

Principle of Virtual Work

Elastic potential energy or simply elastic energy associated with the deformation can be given by strain and its energetically conjugate stress. Note that the Cauchy stress and Almansi-Euler strain tensors and Second Piola-Kirchhoff (2PK) and Green-Lagrange tensors are energetically conjugate. Thus, the ***total internal energy*** due to strain in the body at the deformed state (fig. 3.1) can be given by,

$$W_{int} = \int_0^v (\boldsymbol{e} : \boldsymbol{\sigma})\, \mathrm{d}v \tag{6.288}$$

where $\boldsymbol{e}$ and $\boldsymbol{\sigma}$ are Almansi strain tensor and Cauchy stress tensor respectively.

If the deformed body is further deformed by introducing virtual displacements, then the new internal elastic energy can be given by,

$$W_{int} + \delta W_{int} = \int_0^v [(\boldsymbol{e} + \boldsymbol{\delta e}) : \boldsymbol{\sigma}]\, \mathrm{d}v \tag{6.289}$$

Deducting Equation (6.288) from Equation (6.289),

$$\delta W_{int} = \int_0^v (\boldsymbol{\delta\epsilon} : \boldsymbol{\sigma})\ \mathbf{d}v \tag{6.290}$$

Using Equation (6.287) for virtual strain,

$$\boldsymbol{\delta e} = \frac{\partial \boldsymbol{\delta u}}{\partial \boldsymbol{z}} + \frac{\partial \boldsymbol{\delta u}^T}{\partial \boldsymbol{z}} + \frac{\partial \boldsymbol{\delta u}^T}{\partial \boldsymbol{z}} \frac{\partial \boldsymbol{\delta u}}{\partial \boldsymbol{z}} \tag{6.291}$$

Since virtual displacements are infinitesimally small, quadratic terms in Equation (6.291) can be neglected. The resulting strain tensor, known as small strain tensor $\boldsymbol{\epsilon}$, can be given as,

$$\boldsymbol{\delta\epsilon} = \frac{\partial \boldsymbol{\delta u}}{\partial \boldsymbol{z}} + \frac{\partial \boldsymbol{\delta u}^T}{\partial \boldsymbol{z}} \tag{6.292}$$

Since both $\boldsymbol{\sigma}$ and $\boldsymbol{\delta\epsilon}$ are symmetric, new vectors are defined by inserting tensor components as follows,

$$\boldsymbol{\delta\epsilon} = [\delta\epsilon_{11}\ \delta\epsilon_{22}\ \delta\epsilon_{33}\ 2\delta\epsilon_{12}\ 2\delta\epsilon_{23}\ 2\delta\epsilon_{13}]^T : \boldsymbol{\sigma} = [\delta\sigma_{11}\ \delta\sigma_{22}\ \delta\sigma_{33}\ 2\delta\sigma_{12}\ 2\delta\sigma_{23}\ 2\delta\sigma_{13}]^T \tag{6.293}$$

Substituting Equation (6.293) into Equation (6.290),

$$\delta W_{int} = \int_0^v \left(\boldsymbol{\delta\epsilon}^T \boldsymbol{\sigma}\right)\ \mathbf{d}v \tag{6.294}$$

The strain vector $\boldsymbol{\delta\epsilon}$ can be related to displacement vector using the following equation,

$$\boldsymbol{\delta\epsilon} = \boldsymbol{D}\boldsymbol{\delta u} \tag{6.295}$$

where $\boldsymbol{D}$ and $\boldsymbol{u}$ are linear differential operator and displacement vector respectively and given by,

$$\boldsymbol{D} = \begin{pmatrix} \frac{\partial}{\partial z_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial z_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial z_3} \\ \frac{\partial}{\partial z_2} & \frac{\partial}{\partial z_1} & 0 \\ 0 & \frac{\partial}{\partial z_3} & \frac{\partial}{\partial z_2} \\ \frac{\partial}{\partial z_3} & 0 & \frac{\partial}{\partial z_1} \end{pmatrix} \tag{6.296}$$

$$\boldsymbol{\delta u} = (\delta u_1 \ \delta u_2 \ \delta u_3)^T \tag{6.297}$$

The virtual displacement is a finite element field and hence the value at any point can be obtained by interpolating nodal virtual displacements.

$$\boldsymbol{\delta u} = \boldsymbol{\Phi \Delta} \tag{6.298}$$

6.4 Fluid Mechanics Class

6.4.1 Burgers's Equations

Generalised Burgers's Equation

Governing equations:

For a given velocity $u\left(x,t\right)$ and a kinematic viscosity of ν, Burgers's equation in one dimension is

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2} \tag{6.299}$$

The general form of this equation in OpenCMISS is7

$$a\left(\boldsymbol{x}\right)\frac{\partial \boldsymbol{u}\left(\boldsymbol{x},t\right)}{\partial t} + b\left(\boldsymbol{x}\right)\nabla^2\boldsymbol{u}\left(\boldsymbol{x},t\right) + c\left(\boldsymbol{x}\right)\boldsymbol{u}\left(\boldsymbol{x},t\right)\cdot\nabla\boldsymbol{u}\left(\boldsymbol{x},t\right) = \boldsymbol{0} \tag{6.300}$$

where $\boldsymbol{u}\left(\boldsymbol{x},t\right)$ is the velocity vector and $a\left(\boldsymbol{x}\right)$, $b\left(\boldsymbol{x}\right)$ and $c\left(\boldsymbol{x}\right)$ are material parameters. The standard form of Burgers's equation can be found with $a = 1$, $b = -\nu$ and $c = 1$.

Appropriate boundary conditions conditions for Burgers's equation are specification of Dirichlet boundary conditions on the solution, $\boldsymbol{d}$ *i.e.,*

$$\boldsymbol{u}\left(\boldsymbol{x},t\right) = \boldsymbol{d}\left(\boldsymbol{x},t\right) \quad \boldsymbol{x}\in\Gamma_D, \tag{6.301}$$

and/or Neumann conditions in terms of the solution flux in the normal direction, $\boldsymbol{e}$ *i.e.,*

$$\boldsymbol{q}\left(\boldsymbol{x},t\right) = \left(b\left(\boldsymbol{x}\right)\nabla\boldsymbol{u}\left(\boldsymbol{x},t\right)\right)\cdot\boldsymbol{n} = \boldsymbol{e}\left(\boldsymbol{x},t\right) \quad \boldsymbol{x}\in\Gamma_N, \tag{6.302}$$

where $\boldsymbol{q}\left(\boldsymbol{x},t\right)$ is the flux in the normal direction, $\boldsymbol{n}$ is the normal vector to the boudary and $\Gamma = \Gamma_D \cup \Gamma_N$.

Appropriate initial conditions for the diffusion equation are the specification of an initial value of the solution, $\boldsymbol{f}$ *i.e.,*

$$\boldsymbol{u}\left(\boldsymbol{x},0\right) = \boldsymbol{f}\left(\boldsymbol{x}\right) \quad \boldsymbol{x}\in\Omega. \tag{6.303}$$

Weak formulation:

The corresponding weak form of Equation (6.300) can be found by integrating over the domain with test functions *i.e.,*

$$\int\limits_{\Omega} \left(a\frac{\partial \boldsymbol{u}}{\partial t} + b\nabla^2 7\boldsymbol{u} + c\boldsymbol{u}\cdot\nabla\boldsymbol{u} \right) \boldsymbol{w}\,\mathbf{d}\Omega = \mathbf{0} \tag{6.304}$$

where $\boldsymbol{w}$ are suitable spatial test functions.

Applying the divergence theorem in Equation (6.398) to Equation (6.304) gives

$$\int\limits_{\Omega} \left(a\frac{\partial \boldsymbol{u}}{\partial t} \right) \boldsymbol{w}\,\mathbf{d}\Omega - \int\limits_{\Omega} b\nabla\boldsymbol{u}\cdot\nabla\boldsymbol{w}\,\mathbf{d}\Omega + \int\limits_{\Gamma} \left(b\nabla\boldsymbol{u}\cdot\boldsymbol{n}\right)\boldsymbol{w}\,\mathbf{d}\Gamma + \int\limits_{\Omega} \left(c\boldsymbol{u}\cdot\nabla\boldsymbol{u}\right)\boldsymbol{w}\,\mathbf{d}\Omega = \mathbf{0} \tag{6.305}$$

Tensor notation:

Equation (6.305) can be written in tensor notation as

$$\int\limits_{\Omega} a\dot{u}_i w_i\,\mathbf{d}\Omega - \int\limits_{\Omega} bG^{jk}u_{i;j}w_{i;k}\,\mathbf{d}\Omega + \int\limits_{\Gamma} bG^{jk}u_{i;k}n_j w_i\,\mathbf{d}\Gamma + \int\limits_{\Omega} cG^{jk}u_j u_{i;k} w_i\,\mathbf{d}\Omega = \mathbf{0} \tag{6.306}$$

or

$$\int\limits_{\Omega} a\dot{u}_i w_i\,\mathbf{d}\Omega - \int\limits_{\Omega} bG^{jk}u_{i;j}w_{i;k}\,\mathbf{d}\Omega + \int\limits_{\Gamma} q_i w_i\,\mathbf{d}\Gamma + \int\limits_{\Omega} cG^{jk}u_j u_{i;k} w_i\,\mathbf{d}\Omega = \mathbf{0} \tag{6.307}$$

or

$$\int\limits_{\Omega} a\dot{u}_i w_i\,\mathbf{d}\Omega - \int\limits_{\Omega} bG^{jk}\left(u_{i,j} - \Gamma^i_{jh}u_h\right)\left(w_{i,k} - \Gamma^i_{kl}w_l\right)\,\mathbf{d}\Omega +$$
$$\int\limits_{\Gamma} q_i w_i\,\mathbf{d}\Gamma + \int\limits_{\Omega} cG^{jk}u_j\left(u_{i,k} - \Gamma^i_{kh}u_h\right) w_i\,\mathbf{d}\Omega = \mathbf{0} \tag{6.308}$$

where G^{jk} is the contravariant metric tensor and Γ^i_{jk} is the Christoffel symbol for the spatial coordinates.

Finite element formulation:

We can now discretise the domain into finite elements *i.e.,* $\Omega = \bigcup\limits_{e=1}^{E}\Omega_e$ with $\Gamma = \bigcup\limits_{f=1}^{F}\Gamma_f$.

Equation (6.307) now becomes

$$\sum_{e=1}^{E}\int_{\Omega_e} a\dot{u}_i w_i \,\mathbf{d}\Omega - \sum_{e=1}^{E}\int_{\Omega_e} bG^{jk}\left(u_{i,j}-\Gamma^{i}_{jh}u_h\right)\left(w_{i,k}-\Gamma^{i}_{kl}w_l\right)\,\mathbf{d}\Omega + \sum_{f=1}^{F}\int_{\Gamma_f} q_i w_i\,\mathbf{d}\Gamma + \sum_{e=1}^{E}\int_{\Omega_e} cG^{jk}u_j\left(u_{i,k}-\Gamma^{i}_{kh}u_h\right)w_i\,\mathbf{d}\Omega = \mathbf{0} \quad (6.309)$$

If we assume that we are in rectangular cartesian coordinates then the Christoffel symbols are all zero and $G^{jk}=\delta^{jk}$. Equation (6.309) thus becomes

$$\sum_{e=1}^{E}\int_{\Omega_e} a\dot{u}_i w_i\,\mathbf{d}\Omega - \sum_{e=1}^{E}\int_{\Omega_e} b{u_i}^{,k}w_{i,k}\,\mathbf{d}\Omega + \sum_{f=1}^{F}\int_{\Gamma_f} q_i w_i\,\mathbf{d}\Gamma + \sum_{e=1}^{E}\int_{\Omega_e} cu^k u_{i,k}w_i\,\mathbf{d}\Omega = \mathbf{0} \quad (6.310)$$

If we now assume that the dependent variable $\boldsymbol{u}$ can be interpolated separately in space and in time we can write

$$\boldsymbol{u}(\boldsymbol{x},t) = \psi_n(\boldsymbol{x})\,\boldsymbol{u}^n(t) \quad (6.311)$$

or, in standard interpolation notation within an element,

$$u_i(\boldsymbol{\xi},t) = \psi^{\beta}_{in}(\boldsymbol{\xi})\,u^{n}_{i,\beta}(t)\,\mathrm{S}(i,n,\beta) \quad (6.312)$$

and

$$u^i(\boldsymbol{\xi},t) = G^{ij}\psi^{\beta}_{jn}(\boldsymbol{\xi})\,u^{n}_{j,\beta}(t)\,\mathrm{S}(j,n,\beta) \quad (6.313)$$

where $u^{n}_{i,\beta}(t)$ are the time varying nodal degrees-of-freedom for velocity component i, node n, global derivative β, $\psi^{\beta}_{in}(\boldsymbol{\xi})$ are the corresponding basis functions and $\mathrm{S}(i,n,\beta)$ are the scale factors.

We can also interpolate the other variables in a similar manner *i.e.,*

$$\begin{aligned} q_i(\boldsymbol{\xi},t) &= \psi^{\gamma}_{io}(\boldsymbol{\xi})\,q^{o}_{i,\gamma}(t)\,\mathrm{S}(i,o,\gamma) \\ a(\boldsymbol{\xi}) &= \psi^{\delta}_{p}(\boldsymbol{\xi})\,a^{p}_{,\delta}\mathrm{S}(p,\delta) \\ b(\boldsymbol{\xi}) &= \psi^{\delta}_{p}(\boldsymbol{\xi})\,b^{p}_{,\delta}\mathrm{S}(p,\delta) \\ c(\boldsymbol{\xi}) &= \psi^{\delta}_{p}(\boldsymbol{\xi})\,c^{p}_{,\delta}\mathrm{S}(p,\delta) \end{aligned} \quad (6.314)$$

where $q_{i,\gamma}^{o}(t)$, $a_{,\delta}^{p}$, $b_{,\delta}^{p}$ and $c_{,\delta}^{p}$ are the nodal degrees-of-freedom for the variables.

For a Galerkin finite element formulation we also choose the spatial weighting function $\boldsymbol{w}$ to be equal to the basis fucntions *i.e.,*

$$w_i(\boldsymbol{\xi}) = \psi_{im}^{\alpha}(\boldsymbol{\xi})\,\mathrm{S}(i,m,\alpha) \tag{6.315}$$

Spatial integration:

Adopting the standard integration notation we can write the spatial integration term in Equation (**??**) as

6.4.2 Poiseuille Flow Equations

Governing equations:

The Poiseuille equation describes the pressure drop that occurs due to viscous friction from laminar fluid flow over a straight pipe. Poiseuille flow can be derived from the Navier-Stokes equationsin cylindrical coordinates ($\boldsymbol{u} = (u_r, u_\theta, u_x)$) by assuming that the flow is:

1. Steady *i.e.,* $\dfrac{\partial \boldsymbol{u}}{\partial t} = 0$
2. The radial and swirl components of fluid velocity are zero *i.e.,* $u_r = u_\theta = 0$
3. The flow is axisymmetric *i.e.,* $\dfrac{\partial \boldsymbol{u}}{\partial \theta} = 0$ and fully developed *i.e.,* $\dfrac{\partial \boldsymbol{u}}{\partial x} = 0$.

The Poiseuille equation is:

$$\boxed{\Delta p = \frac{8\mu L Q}{\pi R^4}} \tag{6.316}$$

where Δp is the pressure drop along the pipe, μ is the viscosity L is the length of the pipe Q is volumetric flow and R is the radius of the pipe.

Rearranging Equation (6.316) gives

$$Q = \frac{\pi R^2 R^2 \Delta p}{8\mu L} \tag{6.317}$$

Volumetric flow is average axisymmetric flow, $\bar{u}$, multipled by the area of the pipe A *i.e.,* $Q = \bar{u}A$. Equation (6.316) now becomes

$$\bar{u}A = \frac{AR^2}{8\mu}\Delta p \tag{6.318}$$

or

$$\frac{R^2}{8\mu}\nabla p - \bar{u} = 0 \tag{6.319}$$

as

$$\frac{\Delta p}{L} = \frac{\partial p}{\partial x} = \nabla p \tag{6.320}$$

Weak formulation:

The weak form of the differential form of the Poiseuille equation system consisting of Equation (6.316) and can be written as:

$$\int\limits_{\Omega} \left(\frac{R^2}{8\mu} \nabla p - \bar{u} \right) w \, \mathbf{d}\Omega = 0 \tag{6.321}$$

Finite element formulation:

We can now discretise the domain into finite elements. Equation (6.321) becomes

$$\sum_{e=1}^{E} \int\limits_{\Omega_e} \left(\frac{R^2}{8\mu} \nabla p - \bar{u} \right) w \, \mathbf{d}\Omega = 0 \tag{6.322}$$

or

$$\sum_{e=1}^{E} \left[\int\limits_{\Omega_e} \left(\frac{R^2}{8\mu} \nabla p w \right) \Omega - \int\limits_{\Omega_e} \bar{u} \, \mathbf{d}w \, \mathbf{d}\Omega \right] = 0 \tag{6.323}$$

We can now interpolate the dependent variables using basis functions

$$\begin{aligned} p(\xi) &= \psi_n^{\beta}(\xi) \, p_{,\beta}^{n} \mathrm{S}(n, \beta) \\ \bar{u}(\xi) &= u_e \end{aligned} \tag{6.324}$$

Using a Galerkin finite element approach (where the weighting functions are chosen to be the interpolating basis functions) we have

$$w(\xi) = \psi_m^{\alpha}(\xi) \, \mathrm{S}(m, \alpha) \tag{6.325}$$

Spatial integration:

Adopting the standard integration notation we can write the spatial integration term for each element in Equation (6.327) as

$$\int_0^1 \frac{R^2}{8\mu} \frac{\partial \psi_n^\beta (\xi)\, p_{,\beta}^n \mathrm{S}(n,\beta)}{\partial \xi} \frac{\partial \xi}{\partial x} \psi_m^\alpha (\xi)\, \mathrm{S}(m,\alpha)\, |J(\xi)|\; \mathbf{d}\xi - \int_0^1 u_e \psi_m^\alpha (\xi)\, \mathrm{S}(m,\alpha)\, |J(\xi)|\; \mathbf{d}\xi = \mathbf{0} \quad (6.326)$$

or, taking the fixed degrees-of-freedom and scale factors outside the integrals, we have

$$\frac{R^2}{8\mu} \mathrm{S}(m,\alpha)\, \mathrm{S}(n,\beta)\, p_{,\beta}^n \int_0^1 \psi_m^\alpha (\xi) \frac{\partial \psi_n^\beta (\xi)}{\partial \xi} \frac{\partial \xi}{\partial x} |J(\xi)|\; \mathbf{d}\xi - \mathrm{S}(m,\alpha)\, u_e \int_0^1 \psi_m^\alpha (\xi)\, |J(\xi)|\; \mathbf{d}\xi = \mathbf{0} \quad (6.327)$$

This is an elemental equation of the form

$$K_{mn}^{\alpha\beta} p_{,\beta}^n - f_m^\alpha u_e = \mathbf{0} \quad (6.328)$$

or

$$\begin{bmatrix} K_{mn}^{\alpha\beta} & -f_m^\alpha \\ \mathbf{0}^T & 0 \end{bmatrix} \begin{bmatrix} p_{,\beta}^n \\ u_e \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \end{bmatrix} \quad (6.329)$$

The elemental stiffness matrix, $K_{mn}^{\alpha\beta}$, is given by

$$K_{mn}^{\alpha\beta} = \frac{R^2}{8\mu} \mathrm{S}(m,\alpha)\, \mathrm{S}(n,\beta) \int_0^1 \psi_m^\alpha (\xi) \frac{\partial \psi_n^\beta (\xi)}{\partial \xi} \frac{\partial \xi}{\partial x} |J(\xi)|\; \mathbf{d}\xi \quad (6.330)$$

and the elemental stiffness vector, f_m^α, is given by

$$f_m^\alpha = \mathrm{S}\left(m,\alpha\right) u_e \int\limits_0^1 \psi_m^\alpha\left(\xi\right) \left|J\left(\xi\right)\right| \,\mathbf{d}\xi \tag{6.331}$$

Note that an additional conservation of mass equations are required to solve the final matrix system.

6.4.3 Stokes Equations

Governing equations:

Stokes flow is a type of fluid flow where advective inertial forces are small compared with viscous forces. The Reynolds number is low, i.e. $\mathit{Re} \ll 1$. For this type of flow, the inertial forces are assumed to be negligible and the Navier-Stokes equations simplify to give the Stokes equations. In the common case of an incompressible Newtonian fluid, the linear Stokes equations (three-dimensional, quasi-static) can be written as:

$$\boldsymbol{f} - \nabla p + \mu \nabla^2 \boldsymbol{u} = \boldsymbol{0} \tag{6.332}$$

accompanied by the conservation of mass

$$\nabla \cdot \boldsymbol{u} = 0 \tag{6.333}$$

where $\boldsymbol{u}(\boldsymbol{x}, t) = [u_1, u_2, u_3]^T$ is the velocity vector depending on spatial coordinates $\boldsymbol{x} = [x_1, x_2, x_3]^T$ and the time t, p is the scalar pressure, $\boldsymbol{f}$ an applied body force, and the material parameter μ is fluid viscosity, respectively. Whereas Equation (6.332) has been formulated in Eulerian form, moving domain approaches often require the ALE modification taking an additional term into account, depending on the fluid density ρ and a correction velocity $\boldsymbol{u}^*$ which yields to:

$$\boldsymbol{f} - \nabla p + \mu \nabla^2 \boldsymbol{u} = -\rho \boldsymbol{u}^* \cdot \nabla \boldsymbol{u} \tag{6.334}$$

Although by definition a Stokes flow has no dependence on time (other than through changing boundary conditions), Equation (6.334) can be modified easily towards a dynamic Stokes flow:

$$\rho \frac{\partial \boldsymbol{u}}{\partial t} - \rho \boldsymbol{u}^* \cdot \nabla \boldsymbol{u} = \boldsymbol{f} - \nabla p + \mu \nabla^2 \boldsymbol{u} \tag{6.335}$$

The following section, however, describes the reordered quasi-static formulation of Equation (6.334):

$$-\nabla p + \mu \nabla^2 \boldsymbol{u} - \rho \boldsymbol{u}^* \cdot \nabla \boldsymbol{u} = \boldsymbol{f} \tag{6.336}$$

Weak formulation:

The corresponding weak form of the equation system consisting of Equation (6.332) and Equation (6.333) can be written as:

$$-\int\limits_{\Omega} \nabla p \boldsymbol{v} \,\mathrm{d}\Omega + \int\limits_{\Omega} \mu \nabla^2 \boldsymbol{u}\boldsymbol{v} \,\mathrm{d}\Omega - \int\limits_{\Omega} \rho \boldsymbol{u}^* \cdot \nabla \boldsymbol{u}\boldsymbol{v} \,\mathrm{d}\Omega + \int\limits_{\Omega} \nabla \cdot \boldsymbol{u} q \,\mathrm{d}\Omega = \int\limits_{\Omega} \boldsymbol{f}\boldsymbol{v} \,\mathrm{d}\Omega \quad (6.337)$$

Applying Green's theorm to Equation (6.337) gives

$$\begin{aligned} \int\limits_{\Omega} \nabla \cdot \boldsymbol{v} p \,\mathrm{d}\Omega - \int\limits_{\Omega} \mu \nabla \boldsymbol{v} : \nabla_{\boldsymbol{u}} \,\mathrm{d}\Omega - \int\limits_{\Omega} \rho \boldsymbol{u}^* \cdot \nabla \boldsymbol{u}\boldsymbol{v} \,\mathrm{d}\Omega + \int\limits_{\Omega} \nabla \cdot \boldsymbol{u} q \,\mathrm{d}\Omega = \\ \int\limits_{\Omega} \boldsymbol{f}\boldsymbol{v} \,\mathrm{d}\Omega + \int\limits_{\Gamma} p \cdot \boldsymbol{n}\boldsymbol{v} \,\mathrm{d}\Gamma - \int\limits_{\Gamma} \boldsymbol{v} \nabla \boldsymbol{u} \cdot \boldsymbol{n} \,\mathrm{d}\Gamma \end{aligned} \quad (6.338)$$

The left-hand side of Equation (6.338) represents the unknown velocity and pressure field on Ω whereas the right-hand side represents Neumann boundary conditions on Γ and source terms (here in form of volume or body forces) on Ω. For now we want to neglect the right-hand side and continue with the homogeneous form of Equation (6.338)

$$\int\limits_{\Omega} \nabla \cdot \boldsymbol{v} p \,\mathrm{d}\Omega - \int\limits_{\Omega} \mu \nabla \boldsymbol{v} : \nabla \boldsymbol{u} \,\mathrm{d}\Omega - \int\limits_{\Omega} \rho (\boldsymbol{u}^* \cdot \nabla) \boldsymbol{u}\boldsymbol{v} \,\mathrm{d}\Omega + \int\limits_{\Omega} \nabla \cdot \boldsymbol{u} q \,\mathrm{d}\Omega = 0 \quad (6.339)$$

Tensor notation:

If we now consider the integrand of the left hand side of Equation (6.339) in tensorial form with covariant derivatives then

6.4.4 Darcy Equation

Darcy's equation describes the flow of fluid through a porous material. Ignoring inertial and body forces, Darcy's equation is:

$$\boldsymbol{v} = -\frac{\boldsymbol{K}}{\mu}\nabla p \tag{6.340}$$

$\boldsymbol{v}$ is the relative fluid flow vector, given by $n\left(\boldsymbol{v}^f - \boldsymbol{v}^s\right)$ where n is the porosity, and $\boldsymbol{v}^f$ and $\boldsymbol{v}^s$ are the Eulerian fluid and solid component velocities. $\boldsymbol{K}$ is the permeability tensor, μ is viscosity and p is the fluid pressure.

Conservation of mass also gives:

$$\nabla \cdot v = 0 \tag{6.341}$$

The weighted residual forms of these equations over a region Ω with surface Γ are:

$$\int\limits_{\Omega} \left(\boldsymbol{v}\boldsymbol{w} + \frac{\boldsymbol{K}}{\mu}\nabla p\boldsymbol{w}\right) \mathbf{d}\Omega = 0 \tag{6.342}$$

$$\int\limits_{\Omega} q\nabla \cdot v\, \mathbf{d}\Omega = 0 \tag{6.343}$$

Where $\boldsymbol{w}$ is the weighting function for the flow and q is the weighting function for pressure. Applying Green's theorem to the pressure term to give the weak form of the equations, and multiplying through by $\mu\boldsymbol{K}^{-1}$:

$$\int\limits_{\Omega} \left(\mu\boldsymbol{K}^{-1}\boldsymbol{v}\boldsymbol{w} - p\nabla\boldsymbol{w}\right) \mathbf{d}\Omega + \int\limits_{\Gamma} p\boldsymbol{n}\boldsymbol{w}\, \mathbf{d}\Gamma = 0 \tag{6.344}$$

$$\int\limits_{\Omega} q\nabla \cdot \boldsymbol{v}\, \mathbf{d}\Omega = 0 \tag{6.345}$$

Where $\boldsymbol{n}$ is the normal vector to the surface Γ.

The OpenCMISS implementation of Darcy's equation uses the stabilised form of the finite element equations developed by (**?**). This adds stabilising terms, so that the final form of the implemented equations is:

$$\int_{\Omega} \left(\mu \boldsymbol{K}^{-1} \boldsymbol{v} \boldsymbol{w} - p \nabla \boldsymbol{w} - \frac{1}{2} \left(\nabla p \boldsymbol{w} + \mu \boldsymbol{K}^{-1} \boldsymbol{v} \boldsymbol{w} \right) \right) \mathbf{d}\Omega + \int_{\Gamma} p \boldsymbol{n} \boldsymbol{w} \, \mathbf{d}\Gamma = 0 \tag{6.346}$$

$$\int_{\Omega} q \nabla \cdot \boldsymbol{v} + \frac{1}{2} \left(\nabla q \cdot \boldsymbol{v} + \frac{\boldsymbol{K}}{\mu} \nabla p \cdot \nabla q \right) \mathbf{d}\Omega = 0 \tag{6.347}$$

6.4.5 Navier-Stokes Equations

Governing equations:

The Navier-Stokes equations arise from applying Newton's second law to fluid motion, *i.e.,* the temporal and spatial fluid inertia is in equilibrium with internal (volume/body) and external (surface) forces. The reaction of surface forces can be described in terms of the fluid stress as the sum of a diffusing viscous term, plus a pressure term. The equations are named for the 19th century contributions of Claude-Louis Navier and George Gabriel Stokes. A solution of the Navier-Stokes equations is called a flow field, *i.e.,* velocity and pressure field, which is a description of the fluid at a given point in space and time. In the common case of an incompressible Newtonian fluid, the nonlinear Navier-Stokes equations (three-dimensional, transient) can be written using 'primitive variables' (*i.e.,* u-velocity, p-pressure) as:

$$\rho \frac{\partial \boldsymbol{u}}{\partial t} + \rho \left(\boldsymbol{u} \cdot \nabla \right) \boldsymbol{u} = \boldsymbol{f} - \nabla p + \mu \nabla^2 \boldsymbol{u} \tag{6.348}$$

accompanied by the conservation of mass (incompressibility)

$$\nabla \cdot \boldsymbol{u} = 0 \tag{6.349}$$

where $\boldsymbol{u}(\boldsymbol{x}, t) = (u_1, u_2, u_3)^T$ is the velocity vector depending on spatial coordinates $\boldsymbol{x} = (x_1, x_2, x_3)^T$ and the time t, p is the scalar pressure, $\boldsymbol{f}$ an applied body force, and the material parameters μ and ρ are the fluid viscosity and density, respectively.The first term represents unsteady accelerative inertial contributions, the second represents the nonlinear convective acceleration terms, the ∇p term the pressure contributions, and the last term represents viscous stresses in the system. As with Stokes flow, the incompressibility condition $\nabla \cdot \boldsymbol{u} = 0$ also creates restrictions on the formulation of the velocity and pressure spaces using finite elements known as the Ladyzhenkaya, Babuska, and Brezzi (LBB) or inf-sup consistency condition. Several methods have been devised to define a pressure function that is consistent with the velocity space using primitive variables. These include mixed element methods, penalty methods, generalized petrov-galerkin methods using pressure poisson correction, operator splitting, and semi-implicit pressure correction (**?**).

Using a classic Galerkin formulation, mixed methods are perhaps conceptually the most straightforward method of satisfying LBB, in which velocity is defined over a space one order higher than pressure (e.g. quadratic elements for velocity, linear for pressure), allowing incompressibility to be satisfied. It should be noted that our use of a mixed formulation to satisfy LBB will also be reflected in the shape functions that our weak formulation depends on. For example, using a 2D element with biquadratic velocity and linear pressure, we will have 9 DOFs and weight functions for each velocity component and 4 for the pressure.

Weak Formulation

The weak form of equations 6.348 and 6.349 can be found by applying standard Galerkin test functions $\boldsymbol{w}$:

$$\int_{\Omega} \left(\rho \frac{\partial \boldsymbol{u}}{\partial t} + \rho \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \boldsymbol{f} - \mu \nabla^2 \boldsymbol{u} + \nabla p \right) \boldsymbol{w} \, \mathrm{d}\Omega = 0 \tag{6.350}$$

Integrating by parts, we will get the weak form of the system of PDEs with the associated natural boundary conditions at the boundary Γ_N:

$$\int_{\Omega} \rho \frac{\partial \boldsymbol{u}}{\partial t} \boldsymbol{w} \, \mathrm{d}\Omega + \int_{\Omega} \rho \boldsymbol{u} \cdot \nabla \boldsymbol{u} \boldsymbol{w} \, \mathrm{d}\Omega + \int_{\Omega} p \nabla \boldsymbol{w} \, \mathrm{d}\Omega - \int_{\Omega} \mu \nabla \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{w} \, \mathrm{d}\Omega =$$
$$\int_{\Omega} \boldsymbol{f} \boldsymbol{w} \, \mathrm{d}\Omega + \int_{\Gamma_N} \mu \nabla \boldsymbol{u} \cdot \boldsymbol{n} \boldsymbol{w} \, \mathrm{d}\Gamma_N - \int_{\Gamma_N} p \cdot \boldsymbol{n} \boldsymbol{w} \, \mathrm{d}\Gamma_N \tag{6.351}$$

For more extensive discussion of this procedure, along with other weak forms of the PDEs, we refer to (**?**). From this weak form, we see natural (Neumann) boundary conditions arising as a direct result of the integration. Neumann boundary conditions will consist of a pressure term and viscous stress acting normal to a given boundary.

$$\boldsymbol{q}\left(\boldsymbol{x}, t\right) = \mu \nabla \boldsymbol{u}\left(\boldsymbol{x}, t\right) \cdot \boldsymbol{n} - p \cdot \boldsymbol{n} \tag{6.352}$$
$$\boldsymbol{x} \in \Gamma_N \tag{6.353}$$

Specification of Neumann boundaries will simply require the specification of the terms across element DOFs. Dirichlet boundary conditions on a boundary Γ_D for velocity will take the form:

$$\boldsymbol{u}\left(\boldsymbol{x},t\right) = \boldsymbol{d}\left(\boldsymbol{x},t\right) \quad \boldsymbol{x} \in \Gamma_D \tag{6.354}$$

$$\tag{6.355}$$

Tensor notation

Assuming no forcing terms and substituting the natural boundary as defined above, equation **??** in tensor notation becomes:

$$\int\limits_{\Omega} \rho\dot{u}_i w_i \,\mathbf{d}\Omega + \int\limits_{\Omega} \rho G^{jk} u_j u_{i;k} w_i \,\mathbf{d}\Omega + \int\limits_{\Omega} G^{jk} p_i w_{i;k} \,\mathbf{d}\Omega - \int\limits_{\Omega} \mu G^{jk} u_{i;j} w_{i;k} \,\mathbf{d}\Omega - \int\limits_{\Gamma_N} q_i w_i \,\mathbf{d}\Gamma_N = \mathbf{0} \tag{6.356}$$

or

$$\int\limits_{\Omega} \rho\dot{u}_i w_i \,\mathbf{d}\Omega + \int\limits_{\Omega} \rho G^{jk} u_j \left(u_{i,k} - \Gamma^i_{kh} u_h\right) w_i \,\mathbf{d}\Omega + \int\limits_{\Omega} G^{jk} p_i \left(w_{i,k} - \Gamma^i_{kh} w_h\right) \mathbf{d}\Omega - \int\limits_{\Omega} \mu G^{jk} \left(u_{i,j} - \Gamma^i_{jh} u_h\right)\left(w_{i,k} - \Gamma^i_{kh} w_h\right) \mathbf{d}\Omega - \int\limits_{\Gamma_N} q_i w_i \,\mathbf{d}\Gamma_N = \mathbf{0} \tag{6.357}$$

where G^{jk} is the contravariant metric tensor and Γ^i_{jk} is the Christoffel symbol of the second kind for the spatial coordinates.

Finite Element Formulation

We can now discretise the domain into finite elements *i.e.*, $\Omega = \bigcup\limits_{e=1}^{E} \Omega_e$ with $\Gamma = \bigcup\limits_{f=1}^{F} \Gamma_f$. Equation (6.357) now becomes:

$$\sum_{e=1}^{E}\int_{\Omega}\rho\dot{u}_i w_i\,\mathbf{d}\Omega+\sum_{e=1}^{E}\int_{\Omega}\rho G^{jk}u_j\left(u_{i,k}-\Gamma^i_{kh}u_h\right)w_i\,\mathbf{d}\Omega+\sum_{e=1}^{E}\int_{\Omega}G^{jk}p_i\left(w_{i,k}-\Gamma^i_{kh}w_h\right)\,\mathbf{d}\Omega$$
$$-\sum_{e=1}^{E}\int_{\Omega}\mu G^{jk}\left(u_{i,j}-\Gamma^i_{jh}u_h\right)\left(w_{i,k}-\Gamma^i_{kh}w_h\right)\,\mathbf{d}\Omega-\sum_{f=1}^{F}\int_{\Gamma_N}q_i w_i\,\mathbf{d}\Gamma_N=\mathbf{0}\quad(6.358)$$

or

$$\sum_{e=1}^{E}\int_{\Omega}\rho\dot{u}_i w_i\,\mathbf{d}\Omega+\sum_{e=1}^{E}\int_{\Omega}\rho G^{jk}u_j\left(u_{i,k}-\Gamma^i_{kh}u_h\right)w_i\,\mathbf{d}\Omega+\sum_{e=1}^{E}\int_{\Omega}G^{jk}p_i\left(w_{i,k}-\Gamma^i_{kh}w_h\right)\,\mathbf{d}\Omega$$
$$-\sum_{e=1}^{E}\int_{\Omega}\mu G^{jk}\left(u_{i,j}-\Gamma^i_{jh}u_h\right)\left(w_{i,k}-\Gamma^i_{kh}w_h\right)\,\mathbf{d}\Omega=$$
$$\sum_{f=1}^{F}\int_{\Gamma_N}\mu G^{jk}\left(u_{i,k}-\Gamma^i_{kh}u_h\right)n_i w_i\,\mathbf{d}\Gamma_N-\sum_{f=1}^{F}\int_{\Gamma_N}G^{jk}pn_i w_i\,\mathbf{d}\Gamma_N\quad(6.359)$$

We will assume that the dependent variables $\boldsymbol{u}$ and p can be interpolated separately in space and time. Here we must also be careful to note again the discrepancy between the functional spaces for velocity and pressure using a mixed formulation to satisfy the LBB consistency requirement. We will therefore define two separate weighting functions: for the velocity space on Ω, the test function will be ψ_i and for the pressure space, ϕ_i, giving:

$$\boldsymbol{u}\left(\boldsymbol{x},t\right)=\psi_n\left(\boldsymbol{x}\right)\boldsymbol{u}^n\left(t\right)\quad(6.360)$$
$$p\left(\boldsymbol{x},t\right)=\phi_n(\boldsymbol{x})p^n\left(t\right)\quad(6.361)$$

In standard interpolation notation within an element, we transform from $\boldsymbol{x}$ to $\boldsymbol{\xi}$:

$$u_i\left(\boldsymbol{\xi},t\right)=\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)u^n_{i,\beta}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\quad(6.362)$$
$$p\left(\boldsymbol{\xi},t\right)=\phi^{\beta}_{in}(\boldsymbol{\xi})p^n_{,\beta}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\quad(6.363)$$

where $u^n_{i,\beta}(t)$ are the time varying nodal degrees-of-freedom for velocity component i, node n, global derivative β, $\psi^{\beta}_{in}(\boldsymbol{\xi})$ are the corresponding velocity basis functions and $\mathrm{S}(i,n,\beta)$ are the scale factors. The scalar pressure DOFs, $p^n_{,\beta}(t)$ are interpolated similarly.

For the natural boundary, we can separate q_i into its component velocity and pressure terms as noted in 6.353 and shown in 6.359. For our current treatment, we will also assume the values of viscosity μ and density ρ are constant. These can be interpolated:

$$\begin{aligned} q_i(\boldsymbol{\xi},t) &= \psi^{\gamma}_{io}(\boldsymbol{\xi})\, q^o_{i,\gamma}(t)\,\mathrm{S}(i,o,\gamma) \\ \mu(\boldsymbol{\xi}) &= \psi^{\delta}_{r}(\boldsymbol{\xi})\, \mu^r_{,\delta}\mathrm{S}(r,\delta) \\ \rho(\boldsymbol{\xi}) &= \psi^{\delta}_{r}(\boldsymbol{\xi})\, \rho^r_{,\delta}\mathrm{S}(r,\delta) \end{aligned} \tag{6.364}$$

Using the spatial weighting functions for a Galerkin finite element formulation:

$$w_i(\boldsymbol{\xi}) = \psi^{\alpha}_{im}(\boldsymbol{\xi})\,\mathrm{S}(i,m,\alpha) \tag{6.365}$$

Spatial Integration

Using standard integration notation, we can rewrite our Galerkin FEM formulation from **??**:

$$
\begin{aligned}
&\sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \rho\left(\boldsymbol{\xi}\right) \frac{\partial \psi_{in}^{\beta}\left(\boldsymbol{\xi}\right) u_{i,\beta}^{n}\left(t\right) \mathrm{S}\left(i,n,\beta\right)}{\partial t} \psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right) \mathrm{S}\left(i,m,\alpha\right) \left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right| \, \mathbf{d}\boldsymbol{\xi} \\
&\qquad - \sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \rho\left(\boldsymbol{\xi}\right) G^{jk} \psi_{jn}^{\beta}\left(\boldsymbol{\xi}\right) u_{j,\beta}^{n}\left(t\right) \mathrm{S}\left(i,n,\beta\right) \\
&\left(\frac{\partial \psi_{in}^{\beta}\left(\boldsymbol{\xi}\right) u_{i,\beta}^{n}\left(t\right) \mathrm{S}\left(i,n,\beta\right)}{\partial x^{k}} - \Gamma_{kh}^{i} \psi_{hm}^{\alpha}\left(\boldsymbol{\xi}\right) u_{h,\beta}^{n}\left(t\right) \mathrm{S}\left(h,m,\alpha\right)\right) \psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right) \mathrm{S}\left(i,m,\alpha\right) \left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right| d\boldsymbol{\xi} \\
&\qquad + \sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} G^{jk} \phi_{in}^{\beta}(\boldsymbol{\xi}) p_{,\beta}^{n}\left(t\right) \mathrm{S}\left(i,n,\beta\right) \\
&\qquad \left(\frac{\partial \psi_{im}^{\alpha}(\boldsymbol{\xi}) \mathrm{S}\left(i,m,\alpha\right)}{\partial x^{k}} - \Gamma_{kh}^{i} \psi_{im}^{\alpha}(\boldsymbol{\xi}) \mathrm{S}\left(h,m,\alpha\right)\right) \left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right| d\boldsymbol{\xi} \\
&\quad - \sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \mu\left(\boldsymbol{\xi}\right) G^{jk} \left(\frac{\partial \psi_{in}^{\beta}\left(\boldsymbol{\xi}\right) u_{i,\beta}^{n}\left(t\right) \mathrm{S}\left(i,n,\beta\right)}{\partial x^{j}} - \Gamma_{jh}^{i} \psi_{hn}^{\beta}\left(\boldsymbol{\xi}\right) u_{h,\beta}^{n}\left(t\right) \mathrm{S}\left(h,n,\beta\right)\right) \\
&\qquad \left(\frac{\partial \psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right) \mathrm{S}\left(i,m,\alpha\right)}{\partial x^{k}} - \Gamma_{kl}^{i} \psi_{lm}^{\alpha}\left(\boldsymbol{\xi}\right) u_{h,\beta}^{n}\left(t\right) \mathrm{S}\left(l,m,\alpha\right)\right) \left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right| d\boldsymbol{\xi} \\
&\qquad = \sum_{f=1}^{F}\int\limits_{\mathbf{0}}^{\mathbf{1}} \psi_{io}^{\gamma}\left(\boldsymbol{\xi}\right) q_{i,\gamma}^{o}\left(t\right) \mathrm{S}\left(i,o,\gamma\right) \psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right) \mathrm{S}\left(i,m,\alpha\right) \left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right| \, \mathbf{d}\boldsymbol{\xi}
\end{aligned}
\tag{6.366}
$$

Where $\boldsymbol{J}\left(\boldsymbol{\xi}\right)$ represents the jacobian matrix. If we assume a rectangular cartesian coordinate system, this simplifies significantly, as the contravariant G^{jk} will become δ^{jk} and the Christoffel symbols Γ_{jk}^{i} will all be zero. This gives:

$$\sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \rho\left(\boldsymbol{\xi}\right)\frac{\partial(\psi_{in}^{\beta}\left(\boldsymbol{\xi}\right)u_{i,\beta}^{n}\left(t\right)\mathrm{S}\left(i,n,\beta\right))}{\partial t}\psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(i,m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}$$
$$-\sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \rho\left(\boldsymbol{\xi}\right)\delta^{jk}\psi_{jn}^{\beta}\left(\boldsymbol{\xi}\right)u_{j,\beta}^{n}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\left(\frac{\partial\psi_{in}^{\beta}\left(\boldsymbol{\xi}\right)u_{i,\beta}^{n}\left(t\right)\mathrm{S}\left(i,n,\beta\right)}{\partial x^{k}}\right)$$
$$\psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(i,m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}$$
$$+\sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\phi_{in}^{\beta}(\boldsymbol{\xi})p_{,\beta}^{n}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\left(\frac{\partial\psi_{im}^{\alpha}(\boldsymbol{\xi})\mathrm{S}\left(i,m,\alpha\right)}{\partial x^{k}}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}$$
$$-\sum_{e=1}^{E}\int\limits_{\mathbf{0}}^{\mathbf{1}} \mu\left(\boldsymbol{\xi}\right)\delta^{jk}\left(\frac{\partial\psi_{in}^{\beta}\left(\boldsymbol{\xi}\right)u_{i,\beta}^{n}\left(t\right)\mathrm{S}\left(i,n,\beta\right)}{\partial x^{j}}\right)\left(\frac{\partial\psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(i,m,\alpha\right)}{\partial x^{k}}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}$$
$$=\sum_{f=1}^{F}\int\limits_{\mathbf{0}}^{\mathbf{1}} \psi_{io}^{\gamma}\left(\boldsymbol{\xi}\right)q_{i,\gamma}^{o}\left(t\right)\mathrm{S}\left(i,o,\gamma\right)\psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right)\mathrm{S}\left(i,m,\alpha\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \quad (6.367)$$

or, rearranged to pull constants out of the integrals:

$$\begin{aligned}
&\sum_{e=1}^{E} \dot{u}^n_{i,\beta}(t)\mathrm{S}\left(i,n,\beta\right)\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \rho\left(\boldsymbol{\xi}\right)\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\mathbf{d}\boldsymbol{\xi} \\
&\quad-\sum_{e=1}^{E} u^n_{j,\beta}(t)\mathrm{S}\left(i,n,\beta\right)^2\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\rho\left(\boldsymbol{\xi}\right)\psi^{\beta}_{jn}\left(\boldsymbol{\xi}\right)\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)\left(\frac{\partial\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)}{\partial x^k}\right)|\boldsymbol{J}\left(\boldsymbol{\xi}\right)|d\boldsymbol{\xi} \\
&\qquad+\sum_{e=1}^{E} p^n_{,\beta}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\phi^{\beta}_{in}(\boldsymbol{\xi})\left(\frac{\partial\psi^{\alpha}_{im}(\boldsymbol{\xi})}{\partial x^k}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi} \\
&\quad-\sum_{e=1}^{E} u^n_{i,\beta}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\mu\left(\boldsymbol{\xi}\right)\left(\frac{\partial\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)}{\partial x^j}\right)\left(\frac{\partial\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)}{\partial x^k}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi} \\
&\qquad\qquad=\sum_{f=1}^{F} q^o_{i,\gamma}\left(t\right)\mathrm{S}\left(i,m,\alpha\right)\mathrm{S}\left(i,o,\gamma\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \psi^{\alpha}_{m}\left(\boldsymbol{\xi}\right)\psi^{\gamma}_{o}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi} \qquad (6.368)
\end{aligned}$$

General form

We now seek to assemble this into the corresponding general form for dynamic equations, as outlined in Section 4.2.2:

$$\boldsymbol{M}\ddot{\boldsymbol{d}}\left(t\right)+\boldsymbol{C}\dot{\boldsymbol{d}}\left(t\right)+\boldsymbol{K}\boldsymbol{d}\left(t\right)+\boldsymbol{g}\left(\boldsymbol{d}\left(t\right)\right)+\boldsymbol{f}\left(t\right)=\mathbf{0} \qquad (6.369)$$

where $\dot{\boldsymbol{d}}(t)$ and $\ddot{\boldsymbol{d}}(t)$ represent the first and second derivatives (respectively) of the degrees of freedom vector $\boldsymbol{d}(t)$, which consists of the dependent variables $\boldsymbol{u}(\boldsymbol{x},t)$ and $p(x,t)$. $\boldsymbol{M}$ is the mass matrix, which provides the shape function based weights and $\boldsymbol{C}$ is the transient damping matrix. $\boldsymbol{K}$ represents the stiffness matrix, which will contain the linear parts of the operator. $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ is the nonlinear vector function that will be used to represent the convective term and $\boldsymbol{f}\left(t\right)$ the forcing vector.

We will assume cartesian coordinates $\boldsymbol{x}=\{x,y,z\}$ and denote the corresponding velocity components $\boldsymbol{u}=\{u,v,w\}$, with N representing the number of velocity DOFs and M the number of pressure DOFs. The vector d in then becomes:

$$\boldsymbol{d} = [u_1 u_2 ... u_N v_1 v_2 ... v_N w_1 w_2 ... w_N p_1 p_2 ... p_M]^T = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \\ \boldsymbol{w} \\ \boldsymbol{p} \end{bmatrix} \tag{6.370}$$

All the components of 6.369 will similarly depend on the number of dimensions,n_{dim}, and the size of the matrices will be: $(N{\cdot}n_{dim} + M)^{n_{dim}}$.

Returning to the general case, the mass matrix $\boldsymbol{M}$ will take the form:

$$[M_{mn}^{\alpha\beta}] = \int_{\boldsymbol{0}}^{\boldsymbol{1}} \psi_{in}^{\beta}(\boldsymbol{\xi})\, \psi_{im}^{\alpha}(\boldsymbol{\xi})\, |\boldsymbol{J}(\boldsymbol{\xi})|\; \mathbf{d}\boldsymbol{\xi} \tag{6.371}$$

Without using mass-lumping, the damping matrix $\boldsymbol{C}$:

$$[C_{mn}^{\alpha\beta}] = \mathrm{S}(i,n,\beta)\mathrm{S}(i,m,\alpha) \int_{\boldsymbol{0}}^{\boldsymbol{1}} \rho(\boldsymbol{\xi})\, \psi_{in}^{\beta}(\boldsymbol{\xi})\, \psi_{im}^{\alpha}(\boldsymbol{\xi})\, |\boldsymbol{J}(\boldsymbol{\xi})|\, \mathbf{d}\boldsymbol{\xi} \tag{6.372}$$

The element stiffness matrix $\boldsymbol{K}$ will contain the linear components of the system. For the classic Galerkin formulation this includes the viscous term with the laplacian operator and the pressure gradient term. As the velocity and pressure DOFs are both included in d, $\boldsymbol{K}$ will be assembled so that the corresponding operators are applied to the DOFs discontinuously.

$$[K_{mn}^{\alpha\beta}] = \begin{cases} [A_{mn}^{\alpha\beta}] = \mathrm{S}(i,n,\beta)\,\mathrm{S}(i,m,\alpha) \displaystyle\int_{\boldsymbol{0}}^{\boldsymbol{1}} \delta^{jk} \mu(\boldsymbol{\xi}) \left(\frac{\partial \psi_{in}^{\beta}(\boldsymbol{\xi})}{\partial x^j}\right) \left(\frac{\partial \psi_{im}^{\alpha}(\boldsymbol{\xi})}{\partial x^k}\right) |\boldsymbol{J}(\boldsymbol{\xi})|\, d\boldsymbol{\xi} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } n \leq N{\cdot}N_{dim}, \\ [B_{mn}^{\alpha\beta}] = \mathrm{S}(i,n,\beta)\,\mathrm{S}(i,m,\alpha) \displaystyle\int_{\boldsymbol{0}}^{\boldsymbol{1}} \delta^{jk} \phi_{in}^{\beta}(\boldsymbol{\xi}) \left(\frac{\partial \psi_{im}^{\alpha}(\boldsymbol{\xi})}{\partial x^k}\right) |\boldsymbol{J}(\boldsymbol{\xi})|\, d\boldsymbol{\xi} \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } n > N{\cdot}N_{dim}. \end{cases} \tag{6.373}$$

To attain a square matrix of the required size, we must also apply the transpose of the gradient terms acting on the pressure DOFs. For example, in a 3 dimensional example system, $\boldsymbol{K}$ will take the form:

$$\boldsymbol{K} = \begin{bmatrix} \boldsymbol{A} & 0 & 0 & \boldsymbol{B}_x \\ 0 & \boldsymbol{A} & 0 & \boldsymbol{B}_y \\ 0 & 0 & \boldsymbol{A} & \boldsymbol{B}_z \\ -\boldsymbol{B}_x^T & -\boldsymbol{B}_y^T & -\boldsymbol{B}_z^T & 0 \end{bmatrix} \tag{6.374}$$

The nonlinear vector, $\boldsymbol{g}(t)$ will provide the convective operators and for the standard Galerkin formulation:

$$[g_{mn}^{\alpha\beta}] = \mathrm{S}\left(i, n, \beta\right)^2 \mathrm{S}\left(i, m, \alpha\right) \int_{\mathbf{0}}^{\mathbf{1}} \delta^{jk} \rho\left(\boldsymbol{\xi}\right) \psi_{jn}^{\beta}\left(\boldsymbol{\xi}\right) \psi_{im}^{\alpha}\left(\boldsymbol{\xi}\right) \left(\frac{\partial \psi_{in}^{\beta}\left(\boldsymbol{\xi}\right)}{\partial x^k}\right) |\boldsymbol{J}\left(\boldsymbol{\xi}\right)| d\boldsymbol{\xi} \tag{6.375}$$

Streamline Upwind/Petrov-Galerkin (SUPG)

For convection dominated flows, it is often useful to modify the Galerkin formulation account for numerical problems associated with large asymmetric velocity gradients. We describe augmenting the Galerkin Navier-Stokes formulation to form a Petrov-Galerkin form, which uses a different test function for the convective term to stabilize the algorithm with a balancing diffusive term. Please refer to Section **??** for more details.

For the Navier-Stokes equations (and most nonlinear problems), the use of a Petrov-Galerkin formulation becomes more difficult than for linear cases like advection-diffusion, as consistent weighting can cause instabilities when used with additional terms like body forces (**?**). These issues can be overcome with more complex Petrov-Galerkin formulations, as those derived by Hughes and colleagues in the 1980s. A useful alternative route is to apply SUPG weights to the convective operator in elements as a function of each element's Reynolds number (referred to subsequently as the cell Reynolds number). This can provide the desired stabilizing effect in the direction of large velocity gradients but can be easily implemented and is practically useful.

Assuming cartesian coordinates and applying Petrov-Galerkin weights to the convective term, the finite element formulation described in equation **??** can be written:

$$\sum_{e=1}^{E}\int_{\Omega}\rho\dot{u}_i w_i\,\mathbf{d}\Omega+\sum_{e=1}^{E}\int_{\Omega}\rho\delta^{jk}u_j\left(u_{i,k}\right)\left(w_i+\Psi_i\right)\mathbf{d}\Omega+\sum_{e=1}^{E}\int_{\Omega}\delta^{jk}p_i w_{i,k}\,\mathbf{d}\Omega$$
$$-\sum_{e=1}^{E}\int_{\Omega}\mu\delta^{jk}\left(u_{i,j}\right)\left(w_{i,k}\right)\,\mathbf{d}\Omega=\sum_{f=1}^{F}\int_{\Gamma_N}\mu\delta^{jk}\left(u_{i,k}\right)n_i w_i\,\mathbf{d}\Gamma_N-\sum_{f=1}^{F}\int_{\Gamma_N}\delta^{jk}p n_i w_i\,\mathbf{d}\Gamma_N \tag{6.376}$$

where w_i will be the classic Galerkin weight and Ψ_i will be the Petrov-Galerkin for the Navier-Stokes equations. We must also define the Peclet number, γ, for the Navier-Stokes equations to describe the ratio of the convective:viscous operators. Here, the effective Peclet number will be based on the cell Reynolds number:

$$Re=\frac{\rho\bar{U}L}{\mu}=\frac{\bar{U}L}{\nu} \tag{6.377}$$

where $\bar{U}$ is the characteristic velocity and $\nu=\frac{\mu}{\rho}$ is the kinematic viscosity. L will be the characteristic length scale for a given element. The effective Peclet number γ will be:

$$\gamma=\frac{Re}{2}=\frac{\bar{u}L}{2\nu} \tag{6.378}$$

We will retain the same value for $\alpha=\coth\frac{\gamma}{2}-\frac{2}{\gamma}$ as found for the linearized advection-diffusion equation.

$$\Psi_i=\left(\frac{\alpha L}{2}\frac{u_j}{|u_j|}\right)w_{i,k} \tag{6.379}$$

,

Equation (6.376) then becomes after integration and simplification:

$$
\begin{aligned}
&\sum_{e=1}^{E} \dot{u}^{n}_{i,\beta}(t)\mathrm{S}\left(i,n,\beta\right)\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \rho\left(\boldsymbol{\xi}\right)\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\mathbf{d}\boldsymbol{\xi}\\
&\quad-\sum_{e=1}^{E} u^{n}_{j,\beta}(t)\mathrm{S}\left(i,n,\beta\right)^{2}\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\rho\left(\boldsymbol{\xi}\right)\psi^{\beta}_{jn}\left(\boldsymbol{\xi}\right)\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)\left(\frac{\partial\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)}{\partial x^{k}}\right)|\boldsymbol{J}\left(\boldsymbol{\xi}\right)|d\boldsymbol{\xi}\\
&-\sum_{e=1}^{E} u^{n}_{j,\beta}(t)\mathrm{S}\left(i,n,\beta\right)^{2}\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\rho\left(\boldsymbol{\xi}\right)\psi^{\beta}_{jn}\left(\boldsymbol{\xi}\right)\left(\frac{(\alpha)L(\boldsymbol{\xi})}{2}\frac{u_j}{|u_j|}\right)\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)\left(\frac{\partial\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)}{\partial x^{k}}\right)|\boldsymbol{J}\left(\boldsymbol{\xi}\right)|d\boldsymbol{\xi}\\
&\qquad+\sum_{e=1}^{E} p^{n}_{,\beta}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\phi^{\beta}_{in}(\boldsymbol{\xi})\left(\frac{\partial\psi^{\alpha}_{im}(\boldsymbol{\xi})}{\partial x^{k}}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}\\
&\quad-\sum_{e=1}^{E} u^{n}_{i,\beta}\left(t\right)\mathrm{S}\left(i,n,\beta\right)\mathrm{S}\left(i,m,\alpha\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \delta^{jk}\mu\left(\boldsymbol{\xi}\right)\left(\frac{\partial\psi^{\beta}_{in}\left(\boldsymbol{\xi}\right)}{\partial x^{j}}\right)\left(\frac{\partial\psi^{\alpha}_{im}\left(\boldsymbol{\xi}\right)}{\partial x^{k}}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|d\boldsymbol{\xi}\\
&\qquad\quad=\sum_{f=1}^{F} q^{o}_{i,\gamma}\left(t\right)\mathrm{S}\left(i,m,\alpha\right)\mathrm{S}\left(i,o,\gamma\right)\int\limits_{\mathbf{0}}^{\mathbf{1}} \psi^{\alpha}_{m}\left(\boldsymbol{\xi}\right)\psi^{\gamma}_{o}\left(\boldsymbol{\xi}\right)\left|\boldsymbol{J}\left(\boldsymbol{\xi}\right)\right|\,\mathbf{d}\boldsymbol{\xi}
\end{aligned}
\tag{6.380}
$$

Notice the integration by parts is done only to the standard Galerkin weights- this is because the artificial diffusion added by the Petrov-Galerkin terms should only contribute within the elements

The use of the Petrov-Galerkin formulation in this way allows for the stabilization of moderately convective problems. However, it also results in nonsymmetric mass matrices with the additional term, making classical mass-lumping more difficult. As a result, the use of explicit methods also becomes more difficult for time-dependent problems.

Arbitrary Lagrangian-Eulerian (ALE) Formulation

Whereas Equation (**??**) has been formulated in Eulerian form, moving domain approaches often require the ALE modification taking an additional term into account, depending on the fluid density ρ and a correction velocity $\boldsymbol{u}^*$ which yields to:

$$\rho\left(\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\cdot\nabla\right)\boldsymbol{u}=\boldsymbol{f}-\nabla p+\mu\nabla^2\boldsymbol{u} \tag{6.381}$$

So far, the nonlinear term in Equation (**??**) represents the fluid spatial acceleration only. Equation (6.382) now also takes the dynamic inertia terms into account

$$\rho\frac{\partial\boldsymbol{u}}{\partial t}+\rho\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\cdot\nabla\boldsymbol{u}=\boldsymbol{f}-\nabla p+\mu\nabla^2\boldsymbol{u} \tag{6.382}$$

which gives us the complete Navier-Stokes equations in ALE formulation. The following section, however, describes the reordered quasi-static formulation of Equation (6.381):

$$\rho\left(\left(\boldsymbol{u}-\boldsymbol{u}^*\right)\cdot\nabla\right)\boldsymbol{u}-\mu\nabla^2\boldsymbol{u}+\nabla p=\boldsymbol{f} \tag{6.383}$$

Weak formulation: The corresponding weak form of the equation system consisting of Equation (**??**) and Equation (**??**) can be written in the general dynamic form (see Section **??**)

$$\boldsymbol{M}\ddot{\boldsymbol{u}}\left(t\right)+\boldsymbol{C}\dot{\boldsymbol{u}}\left(t\right)+\boldsymbol{K}\boldsymbol{u}\left(t\right)+\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)+\boldsymbol{f}\left(t\right)=\boldsymbol{0} \tag{6.384}$$

where $u\left(t\right)$ is the dependent variables vector $\boldsymbol{u}(\boldsymbol{x},t)$ and p for the degrees of freedom. $\boldsymbol{M}$ is the mass matrix, which provides the shape function based weights, $\boldsymbol{C}$ is the transient damping matrix (which we will discuss further below). $\boldsymbol{K}$ represents the stiffness matrix, which will contain the linear parts of the operator, including the viscous terms, the conservation of mass terms, and pressure terms. $\boldsymbol{g}\left(\boldsymbol{u}\left(t\right)\right)$ is the nonlinear vector function for the convective terms and $\boldsymbol{f}\left(t\right)$ the forcing vector.

The corresponding weak form of the equation system consisting of Equation (**??**) and Equation (**??**) can be written as:

$$\int\limits_{\Omega}\rho\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u}\boldsymbol{v}\,\mathrm{d}\Omega-\int\limits_{\Omega}\rho\left(\boldsymbol{u}^*\cdot\nabla\right)\boldsymbol{u}\boldsymbol{v}\,\mathrm{d}\Omega+\int\limits_{\Omega}\mu\nabla^2\boldsymbol{u}\boldsymbol{v}\,\mathrm{d}\Omega-\int\limits_{\Omega}\nabla p\boldsymbol{v}\,\mathrm{d}\Omega+\int\limits_{\Omega}\nabla\cdot\boldsymbol{u}q\,\mathrm{d}\Omega=\int\limits_{\Omega}\boldsymbol{f}\boldsymbol{v}\,\mathrm{d}\Omega \tag{6.385}$$

The general form for this kind of equation system is

$$\boldsymbol{K}\hat{\boldsymbol{u}} + \hat{\boldsymbol{g}}\left(\boldsymbol{u}\right) = \hat{\boldsymbol{f}} \tag{6.386}$$

where $\hat{\boldsymbol{u}}$ is the vector of unknown "DOFs", $\boldsymbol{K}$ is the stiffness matrix, $\hat{\boldsymbol{g}}\left(\boldsymbol{u}\right)$ a non-linear vector function and $\hat{\boldsymbol{f}}$ the forcing vector. In Equation (6.385) the only real non-linear term is represented by $\hat{\boldsymbol{g}}\left(\boldsymbol{u}\right) = \int\limits_{\Omega} \rho\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u}\boldsymbol{v}\,\mathrm{d}\Omega$. If $\hat{\boldsymbol{g}}\left(\boldsymbol{u}\right)$ is not $\equiv \boldsymbol{0}$ then we use Newton's method *i.e.*,

$$\begin{aligned} &1.\ \boldsymbol{J}\left(\boldsymbol{u}_i\right).\delta\boldsymbol{u}_i = -\boldsymbol{\psi}\left(\boldsymbol{u}_i\right) \\ &2.\ \boldsymbol{u}_{i+1} = \boldsymbol{u}_i + \delta\boldsymbol{u}_i \end{aligned} \tag{6.387}$$

where $\boldsymbol{J}\left(\boldsymbol{u}\right)$ is the Jacobian and is given by

$$\boldsymbol{J}\left(\boldsymbol{u}\right) = \boldsymbol{K} + \frac{\partial\hat{\boldsymbol{g}}\left(\boldsymbol{u}\right)}{\partial\boldsymbol{u}} \tag{6.388}$$

with the stiffness matrix $\boldsymbol{K}$ derived from Equation (6.386) by applying Green's theorem as follows:

$$\boldsymbol{K}\hat{\boldsymbol{u}} = \int\limits_{\Omega} \nabla\cdot\boldsymbol{v}p\,\mathrm{d}\Omega - \int\limits_{\Omega} \mu\nabla\boldsymbol{v} : \nabla\boldsymbol{u}\,\mathrm{d}\Omega - \int\limits_{\Omega} \rho\left(\boldsymbol{u}^*\cdot\nabla\right)\boldsymbol{u}\boldsymbol{v}\,\mathrm{d}\Omega + \int\limits_{\Omega} \nabla\cdot\boldsymbol{u}q\,\mathrm{d}\Omega \tag{6.389}$$

and $\boldsymbol{\psi}\left(\hat{\boldsymbol{u}}\right) = \boldsymbol{K}\hat{\boldsymbol{u}} + \hat{\boldsymbol{g}}\left(\boldsymbol{u}\right) + \hat{\boldsymbol{f}}$.

6.4.6 Pressure Poisson Equation

The Navier-Stokes equation can be written as

$$\rho \left(\frac{\partial \boldsymbol{u}}{\partial t} - \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) = -\nabla p + \mu \nabla \cdot \nabla \boldsymbol{u} \tag{6.390}$$

Rearranging for ∇p gives

$$\nabla p = \mu \nabla \cdot \nabla \boldsymbol{u} - \rho \left(\frac{\partial \boldsymbol{u}}{\partial t} - \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) \tag{6.391}$$

or

$$\nabla p = \boldsymbol{b} \tag{6.392}$$

where

$$\boldsymbol{b} = \mu \nabla \cdot \nabla \boldsymbol{u} - \rho \left(\frac{\partial \boldsymbol{u}}{\partial t} - \boldsymbol{u} \cdot \nabla \boldsymbol{u} \right) \tag{6.393}$$

Note that from Equation (6.392) we can write

$$\nabla p \cdot \boldsymbol{n} = \boldsymbol{b} \cdot \boldsymbol{n} \tag{6.394}$$

Taking the divergence of both sides of Equation (6.392) gives

$$\nabla \cdot \nabla p = \nabla \cdot \boldsymbol{b} \tag{6.395}$$

The corresponding weak form of Equation (6.395) is

$$\int\limits_{\Omega} \nabla \cdot \nabla p w \, \mathrm{d}\Omega = \int\limits_{\Omega} \nabla \cdot \boldsymbol{b} w \, \mathrm{d}\Omega \tag{6.396}$$

Now the divergence theorm states that

$$\int\limits_{\Omega} \nabla \cdot \boldsymbol{F} \, \mathrm{d}\Omega = \int\limits_{\Gamma} \boldsymbol{F} \cdot \boldsymbol{n} \, \mathrm{d}\Gamma \tag{6.397}$$

Applying the divergence theorm where $\boldsymbol{F} = g\boldsymbol{f}$ gives

$$\int\limits_{\Omega} \left(\boldsymbol{f} \cdot \nabla g + g \nabla \cdot \boldsymbol{f}\right) \, \mathrm{d}\Omega = \int\limits_{\Gamma} g\boldsymbol{f} \cdot \boldsymbol{n} \, \mathrm{d}\Gamma \tag{6.398}$$

Note that when $\boldsymbol{f} = \nabla f$ is used in Equation (6.398) you get Green's identity *i.e.,*

$$\int\limits_{\Omega} \left(\nabla f \cdot \nabla g + g \nabla \cdot \nabla f\right) \, \mathbf{d}\Omega = \int\limits_{\Gamma} g \nabla f \cdot \boldsymbol{n} \, \mathbf{d}\Gamma \tag{6.399}$$

Now if we apply Equation (6.399) to the left hand side of Equation (6.396) we get

$$\int\limits_{\Omega} \nabla \cdot \nabla p w \, \mathbf{d}\Omega = \int\limits_{\Gamma} \nabla p \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma - \int\limits_{\Omega} \nabla p \cdot \nabla w \, \mathbf{d}\Omega \tag{6.400}$$

or from Equation (6.394)

$$\int\limits_{\Omega} \nabla \cdot \nabla p w \, \mathbf{d}\Omega = \int\limits_{\Gamma} \boldsymbol{b} \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma - \int\limits_{\Omega} \nabla p \cdot \nabla w \, \mathbf{d}\Omega \tag{6.401}$$

Now if we apply Equation (6.398) to the right hand side of Equation (6.396) we get

$$\int\limits_{\Omega} \nabla \cdot \boldsymbol{b} w \, \mathbf{d}\Omega = \int\limits_{\Gamma} \boldsymbol{b} \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma - \int\limits_{\Omega} \boldsymbol{b} \cdot \nabla w \, \mathbf{d}\Omega \tag{6.402}$$

Substituting Equation (6.401) and Equation (6.402) into Equation (6.396) gives

$$\int\limits_{\Gamma} \boldsymbol{b} \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma - \int\limits_{\Omega} \nabla p \cdot \nabla w \, \mathbf{d}\Omega = \int\limits_{\Gamma} \boldsymbol{b} \cdot \boldsymbol{n} w \, \mathbf{d}\Gamma - \int\limits_{\Omega} \boldsymbol{b} \cdot \nabla w \, \mathbf{d}\Omega \tag{6.403}$$

or

$$\int\limits_{\Omega} \nabla p \cdot \nabla w \, \mathbf{d}\Omega = \int\limits_{\Omega} \boldsymbol{b} \cdot \nabla w \, \mathbf{d}\Omega \tag{6.404}$$

Now, using a standard Galerkin Finite Element approach Equation (6.405) can be written in the following form

$$\boldsymbol{K}_p \boldsymbol{p} = \boldsymbol{s} \tag{6.405}$$

where $\boldsymbol{K}_p$ is the pressure stiffness matrix, $\boldsymbol{p}$ the vector of unknown pressure DOFs and $\boldsymbol{s}$ the source vector.

Note that you can write $\boldsymbol{s}$ in terms of $\boldsymbol{K}_u$ the velocity stiffness matrix, $\boldsymbol{M}$ the mass matrix, $\boldsymbol{u}$ the vector of know velocity DOFs and $\boldsymbol{h}(\boldsymbol{u})$ the nonlinear Navier-Stokes vector.

Note that $\boldsymbol{K}_P$ is most likely singular and therefore you will need to set a reference pressure Dirichlet condition somewhere and measure the pressures relative to this boundary condition value.

Note that this formulation removes all the nasty Neumann conditions in favour of a Dirichlet condition. It also removes the surface integrals in favour of volume integrals!

6.5 Electromechanics Class

6.5.1 Electrostatic Equations

6.5.2 Magnetostatic Equations

6.5.3 Maxwell Equations

6.6 Bioelectrics Class

6.6.1 Bidomain Equation

The bidomain model can be thought of as two co-existant intra and extrcelluar spaces. The potential in the intracellular space is denoted as ϕ_i and the potential in the extracellular space is denoted as ϕ_e. The intra and extracellular potentials are related through the transmembrane voltage, V_m, i.e.,

$$V_m = \phi_i - \phi_e \tag{6.406}$$

The bidomain equations are

$$A_m C_m \frac{\partial V_m}{\partial t} - \nabla \cdot (\boldsymbol{\sigma}_i \nabla V_m) = \nabla \cdot (\boldsymbol{\sigma}_i \nabla \phi_e) - A_m (I_{ion} - I_m) + i_i \tag{6.407}$$

$$\nabla \cdot ((\boldsymbol{\sigma}_e + \boldsymbol{\sigma}_i) \nabla \phi_e) = -\nabla \cdot (\boldsymbol{\sigma}_i \nabla V_m) - i_i + i_e \tag{6.408}$$

where A_m is the surface to volume ratio for the cell, C_m is the specific membrance capacitance $\boldsymbol{\sigma}_i$ is the intracellular conductivity tensor, $\boldsymbol{\sigma}_e$ is the extracellular conductivity tensor, I_{ion} is the ionic current, I_m is the transmembrane current

Operator splitting

Consider the initial value problem

$$\frac{du}{dt} = (L_1 + L_2) u \tag{6.409}$$

$$u(0) = u_0 \tag{6.410}$$

where L_1 and L_2 are some operators.

For Gudunov splitting we first solve

$$\frac{dv}{dt} = L_1 v \tag{6.411}$$

$$v(0) = u_0 \tag{6.412}$$

for $t \in [0, \Delta t]$ to give $v(\Delta t)$. Next we solve

$$\frac{dw}{dt} = L_2 w \tag{6.413}$$

$$w(0) = v(\Delta t) \tag{6.414}$$

for $t \in [0, \Delta t]$ to give $w(\Delta t)$. We now set

$$\tilde{u}(\Delta t) = w(\Delta t) \tag{6.415}$$

where $\tilde{u}$ is a approximate solution to the orginal initial value problem.

For Strang splitting we first solve

$$\frac{dv}{dt} = L_1 v \tag{6.416}$$

$$v(0) = u_0 \tag{6.417}$$

for $t \in \left[0, \frac{\Delta t}{2}\right]$ to give $v\left(\frac{\Delta t}{2}\right)$. Next we solve

$$\frac{dw}{dt} = L_2 w \tag{6.418}$$

$$w(0) = v\left(\frac{\Delta t}{2}\right) \tag{6.419}$$

for $t \in [0, \Delta t]$ to give $w(\Delta t)$. Next we solve

$$\frac{dv}{dt} = L_1 v \tag{6.420}$$

$$v\left(\frac{\Delta t}{2}\right) = w(\Delta t) \tag{6.421}$$

for $t \in \left[\frac{\Delta t}{2}, \Delta t\right]$ to give $v(\Delta t)$.We now set

$$\tilde{u}(\Delta t) = v(\Delta t) \tag{6.422}$$

where $\tilde{u}$ is a approximate solution to the orginal initial value problem.

6.6.2 Monodomain Equation

Under certain conditions the Biodomain equations given in Equation (6.407) and Equation (6.408) can be simplified to

$$A_m C_m \frac{\partial V_m}{\partial t} - \nabla \cdot (\boldsymbol{\sigma}_m \nabla V_m) = -A_m I_{ion} + i_i \tag{6.423}$$

where A_m is the surface to volume ratio for the cell, C_m is the specific membrance capacitance, I_{ion} is the ionic current, i_i is the intracellular stimulus current and $\boldsymbol{\sigma}_m$ is the conductivity tensor given by

$$\boldsymbol{\sigma}_m = \frac{\boldsymbol{\sigma}_i \boldsymbol{\sigma}_e}{\boldsymbol{\sigma}_i + \boldsymbol{\sigma}_e} \tag{6.424}$$

Rearrangign Equation (6.423) to give

$$A_m C_m \frac{\partial V_m}{\partial t} = \frac{(\nabla \cdot (\boldsymbol{\sigma}_m \nabla V_m) - I_{ion} + i_i)}{} \tag{6.425}$$

6.7 Multiphysics Class

6.7.1 Poroelasticity

Coupled Darcy Equation and Elasticity with Sub-iterations

CMISSProblemFiniteElasticityDarcyType couples the finite elasticity equations and Darcy's equation by solving each equation in term, iterating between the two equations until convergence is reached. The full Darcy equations with velocity and pressure are solved.

Darcy's equation is solved on the deformed geometry and the finite elasticity constitutive relation has a dependence on the Darcy fluid pressure.

Coupled Darcy Fluid Pressure and Elasticity

The fully dynamic governing equations for poromechanics are described in (**?**). CMISSProblemfiniteElasticityFluidPressureType implements a static form of these equations and strongly couples the equations for finite elasticity to an equation describing the Darcy fluid pressure.

Darcy's equation describing the fluid flow is substituted into the mass conservation equation to give:

$$\nabla \cdot \left(\frac{\boldsymbol{K}}{\mu} \nabla p \right) = 0 \tag{6.426}$$

A static form of the constitutive law developed by (**?**) is implemented in the equations set subtype CMISSEquationsSetElasticityFluidPressureStaticInriaSubtype.

6.7.2 Fluid-Structure Interaction

Fluid Model

The physics in the fluid domain is described by the ALE form of the Navier-Stokes equations *i.e.*,

$$\rho^f \left(\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v} \cdot \nabla)(\boldsymbol{v} - \boldsymbol{w}) \right) = \nabla \cdot \boldsymbol{\sigma}^f + \rho^f \boldsymbol{g} \tag{6.427}$$

$$\nabla \cdot \boldsymbol{v} = 0 \tag{6.428}$$

where ρ^f is the fluid density, $\boldsymbol{v}$ is the fluid velocity, $\boldsymbol{w}$ is the fluid mesh velocity, $\boldsymbol{\sigma}^f$ is the fluid stress and $\boldsymbol{g}$ is the gravity acceleration vector.

For Newtonian fluids, we have

$$\boldsymbol{\sigma}^f = -p^f \boldsymbol{I} + \mu \left(\nabla \boldsymbol{v} + (\nabla \boldsymbol{v})^T \right) \tag{6.429}$$

where p^f is the fluid pressure and μ is the fluid viscosity.

Solid Model

The physics in the solid domain is described by a momentum balance

$$\rho^s \frac{\partial^2 \boldsymbol{u}}{\partial t^2} = \nabla \cdot \boldsymbol{\sigma}^s + \rho^s \boldsymbol{g} \tag{6.430}$$

where ρ^s is the solid density, $\boldsymbol{u}$ is the solid displacement, $\boldsymbol{\sigma}^s$ is the solid stress and $\boldsymbol{g}$ is the gravity acceleration vector.

6.8 Modal Class

6.9 Fitting Class

6.10 Optimisation Class

Chapter 7

Analytic Solutions

7.1 Classical Field Class

7.1.1 Generalised Laplace Equation

7.1.2 Two Dimensional Anisotropic Solution

Consider the problem shown in Figure **??**.

For this test problem the conductivity tensor in fibre coordinates is

$$\boldsymbol{\sigma}^* = \begin{bmatrix} \sigma_t & 0 \\ 0 & \sigma_n \end{bmatrix} \tag{7.1}$$

The conductivity tensor in reference coordinates can be calculated from the conductivity tensor in fibre coordinates by the tensor transformation rule $\boldsymbol{\sigma} = \boldsymbol{Q}^T \boldsymbol{\sigma}^* \boldsymbol{Q}$ where $\boldsymbol{Q}$ is given by

$$\boldsymbol{Q} = \begin{bmatrix} \cos(-\theta) & -\sin(-\theta) \\ \sin(-\theta) & \cos(-\theta) \end{bmatrix} = \begin{bmatrix} \cos\theta & +\sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{7.2}$$

Hence

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_t \cos^2\theta + \sigma_n \sin^2\theta & (\sigma_t - \sigma_n)\sin\theta\cos\theta \\ (\sigma_t - \sigma_n)\sin\theta\cos\theta & \sigma_t \sin^2\theta + \sigma_n\cos^2\theta \end{bmatrix} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{bmatrix} \tag{7.3}$$

The generalised Laplace equation for this problem in reference coordinates is hence given by

$$\nabla \cdot (\boldsymbol{\sigma}\nabla\Phi) = \sigma_{11}\frac{\partial^2\Phi}{\partial x^2} + 2\sigma_{12}\frac{\partial^2\Phi}{\partial x\partial y} + \sigma_{22}\frac{\partial^2\Phi}{\partial y^2} = 0 \tag{7.4}$$

Equation (7.4) cannot be solved using traditional eigenfunction expansions as it is non-separable. In fact, for an arbitrary domain and arbitrary boundary conditions, Equation (7.4) is extremely difficult to solve. It can be solved, however, by realising that from the definition of σ_{ij} in Equation (7.3) the equation is elliptic everywhere in the solution domain. From **?**, a system of second order elliptic partial differential equations is of the form

$$a_{ijkl}\frac{\partial^2\Phi_k}{\partial x_j \partial x_l} = 0 \tag{7.5}$$

where Φ_k are complex-valued functions of the dependent variables x_1 and x_2 and $i, k = 1, \ldots, N$. The a_{ijkl} are real constants which satisfy the symmetry condition $a_{ijkl} = a_{klij}$ and, since the system is elliptic, satisfy $a_{ijkl}\xi_{ij}\xi_{kl} > 0$ for every non-zero $N \times 2$ real matrix ξ_{ij}.

Real solutions to Equation (7.5) may be written in the form

$$\Phi_k = \sum_{\alpha} A_{k\alpha} f_\alpha (z_\alpha) + \sum_{\alpha} \overline{A_{k\alpha} f_\alpha (z_\alpha)} \tag{7.6}$$

where $\alpha = 1, \ldots, N$, $f_\alpha (z_\alpha)$ are arbitrary analytic functions of a complex variable z_α, $\overline{f_\alpha (z_\alpha)}$ the complex conjugate of the function $f_\alpha (z_\alpha)$, $A_{k\alpha}$ are complex constants (to be determined), $z_\alpha = x_1 + \tau_\alpha x_2$ and τ_α are complex constants given by

$$\det \left[a_{i1k1} + a_{i1k2}\tau + a_{i2k1}\tau + a_{i2k2}\tau^2 \right] = 0 \tag{7.7}$$

Comparing Equation (7.4) with Equation (7.5) it can be seen that $N = 1$ and $a_{1i1j} = \sigma_{ij}$ and thus τ is given by

$$\tau = \frac{-\sigma_{12} + \imath\sqrt{\sigma_{11}\sigma_{22} - \sigma_{12}^2}}{\sigma_{22}} = \frac{-\sigma_{12}}{\sigma_{22}} + \frac{\imath\sqrt{\sigma_{11}\sigma_{22} - \sigma_{12}^2}}{\sigma_{22}} = -\lambda_1 + \imath\lambda_2 \tag{7.8}$$

where $\imath$ is the square root of -1. With this definition it can be seen that

$$z = x + \tau y = (x - \lambda_1 y) + \imath\lambda_2 y \tag{7.9}$$

From **?** $A_{11} = 1$ for this problem and hence a solution to Equation (7.4) can be found by specifying an arbitrary analytic function $f(z)$.

For the test problem in this chapter the analytic function is chosen to be $f(z) = e^z$. Thus the complete analytic solution is given by

$$\begin{aligned}
\Phi(x, y) &= f(z) + \overline{f(z)} \\
&= e^{(x-\lambda_1 y)+\imath\lambda_2 y} + \overline{e^{(x-\lambda_1 y)+\imath\lambda_2 y}} \\
&= e^{(x-\lambda_1 y)} \left(\cos(\lambda_2 y) + \imath \sin(\lambda_2 y)\right) + e^{(x-\lambda_1 y)} \left(\cos(\lambda_2 y) - \imath \sin(\lambda_2 y)\right) \\
&= 2e^x e^{-\lambda_1 y} \cos(\lambda_2 y)
\end{aligned} \tag{7.10}$$

7.1.3 Diffusion Equation

One Dimensional Analytic Function 1

From http://eqworld.ipmnet.ru/en/solutions/lpde/lpde101.pdf we have for the one-dimension diffusion equation

$$\frac{\partial u}{\partial t} = a\frac{\partial^2 u}{\partial x^2} \tag{7.11}$$

the analytic solution

$$u\left(x,t\right) = Ae^{-a\mu^2 t}\cos(\mu x + B) + C \tag{7.12}$$

Now, OpenCMISS solves diffusion equations of the form

$$\frac{\partial u}{\partial t} + k\frac{\partial^2 u}{\partial x^2} = 0 \tag{7.13}$$

and therefore $a = -k$. Choosing $\mu = \dfrac{2\pi}{L}$ gives the OpenCMISS diffusion equation one dimensional analytic function 1, namely

$$u\left(x,t\right) = Ae^{\frac{4\pi^2 kt}{L^2}}\cos(\frac{2\pi x}{L} + B) + C \tag{7.14}$$

To prove the analytic solution we differentiate Equation (7.14) to give

$$\frac{\partial u}{\partial t} = \frac{4\pi^2 k}{L^2}Ae^{\frac{4\pi^2 kt}{L^2}}\cos(\frac{2\pi x}{L} + B) \tag{7.15}$$

$$\frac{\partial u}{\partial x} = \frac{-2\pi}{L}Ae^{\frac{4\pi^2 kt}{L^2}}\sin(\frac{2\pi x}{L} + B) \tag{7.16}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{2\pi}{L}\frac{-2\pi}{L}Ae^{\frac{4\pi^2 kt}{L^2}}\cos(\frac{2\pi x}{L} + B) \tag{7.17}$$

$$= \frac{-4\pi^2}{L^2}Ae^{\frac{4\pi^2 kt}{L^2}}\cos(\frac{2\pi x}{L} + B) \tag{7.18}$$

Substiting Equation (7.33) into Equation (7.13) gives

$$\frac{\partial u}{\partial t} + k\frac{\partial^2 u}{\partial x^2} = \frac{4\pi^2 k}{L^2}Ae^{\frac{4\pi^2 kt}{L^2}}\cos(\frac{2\pi x}{L} + B) + k\frac{-4\pi^2}{L^2}Ae^{\frac{4\pi^2 kt}{L^2}}\cos(\frac{2\pi x}{L} + B) \tag{7.19}$$

$$= 0 \tag{7.20}$$

The analytic field component definitions in OpenCMISS are shown in Table 7.1.

Analytic constant	*Analytic field component*
A	1
B	2
C	3
L	4

TABLE 7.1: OpenCMISS analytic field components for the one-dimension diffusion equation analytic function 1.

One Dimensional Quadratic Source Analytic Function 1

From http://eqworld.ipmnet.ru/en/solutions/npde/npde1104.pdf we have for the one-dimension polynomial source diffusion equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + qw + rw^m \tag{7.21}$$

the analytic solution

$$u\left(x,t\right) = \left[\pm\beta + Ae^{\left(\lambda t \pm \mu x\right)}\right]^{\frac{2}{1-m}} \tag{7.22}$$

where

$$\beta = \sqrt{\frac{-r}{q}} \tag{7.23}$$

and

$$\lambda = \frac{q\left(1-m\right)\left(m+3\right)}{2\left(m+1\right)} \tag{7.24}$$

and

$$\mu = \sqrt{\frac{q\left(1-m\right)^2}{2\left(m+1\right)}} \tag{7.25}$$

Now, OpenCMISS solves quadratic source diffusion equations of the form

$$\frac{\partial u}{\partial t} + k\frac{\partial^2 u}{\partial x^2} + a + bu + cu^2 = 0 \tag{7.26}$$

and therefore $k = -1$, $m = 2$, $a = 0$, $q = -b$ and $r = -c$. Choosing the positive wave gives the

OpenCMISS quadratic source diffusion equation one dimensional analytic function 1, namely

$$u\left(x,t\right) = \frac{1}{\left[\beta + Ae^{(\lambda t + \mu x)}\right]^2} \tag{7.27}$$

where

$$\beta = \sqrt{\frac{-c}{b}} \tag{7.28}$$

and

$$\lambda = \frac{5b}{6} \tag{7.29}$$

and

$$\mu = \sqrt{\frac{-b}{6}} \tag{7.30}$$

To prove the analytic solution we differentiate Equation (7.27) to give

$$\frac{\partial u}{\partial t} = \frac{2\lambda A\left(\lambda t + \mu x\right)e^{(\lambda t + \mu x)}}{\left[\beta + Ae^{(\lambda t + \mu x)}\right]^3} \tag{7.31}$$

$$\frac{\partial u}{\partial x} = \frac{2\mu A\left(\lambda t + \mu x\right)e^{(\lambda t + \mu x)}}{\left[\beta + Ae^{(\lambda t + \mu x)}\right]^3} \tag{7.32}$$

$$\frac{\partial^2 u}{\partial x^2} = \tag{7.33}$$

The analytic field component definitions in OpenCMISS are shown in Table **??**.

Analytic constant	*Analytic field component*
A	1

TABLE 7.2: OpenCMISS analytic field components for the one-dimension quadratic source diffusion equation analytic function 1.

One Dimensional Exponential Source Analytic Function 1

From http://eqworld.ipmnet.ru/en/solutions/npde/npde1105.pdf we have for the one-dimension exponential source diffusion equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + q + re^{\lambda u} \tag{7.34}$$

the analytic solution

$$u\left(x,t\right)=\frac{-2}{\lambda}\ln\left[\pm\beta+Ae^{\left(\pm\mu x-\frac{1}{2}q\lambda t\right)}\right] \tag{7.35}$$

where

$$\beta=\sqrt{\frac{-r}{q}} \tag{7.36}$$

and

$$\mu=\sqrt{\frac{q\lambda}{2}} \tag{7.37}$$

Now, OpenCMISS solves diffusion equations of the form

$$\frac{\partial u}{\partial t}+k\frac{\partial^2 u}{\partial x^2}+a+be^{cx}=0 \tag{7.38}$$

and therefore $k=-1$, $q=-a$ and $r=-b$. Choosing the positive wave gives the OpenCMISS exponential source diffusion equation one dimensional analytic function 1, namely

$$u\left(x,t\right)=\frac{-2}{c}\ln\left[\beta+Ae^{\mu(x-\mu t)}\right] \tag{7.39}$$

where

$$\beta=\sqrt{\frac{-b}{a}} \tag{7.40}$$

and

$$\mu=\sqrt{\frac{-ac}{2}} \tag{7.41}$$

To prove the analytic solution we differentiate Equation (7.39) to give .....

The analytic field component definitions in OpenCMISS are shown in Table **??**.

Analytic constant	*Analytic field component*
A	1

TABLE 7.3: OpenCMISS analytic field components for the one-dimension exponential source diffusion equation analytic function 1.

Three Dimensional with linear source Analytic Function 1

The three dimensional diffusions equation with a linear source is given by

$$\frac{\partial u}{\partial t} = a\nabla^2 u + bu + c \tag{7.42}$$

The general form of the OpenCMISS diffusion equation with linear source is given by

$$a\frac{\partial u}{\partial t} + k\nabla^2 u + bu + c = 0 \tag{7.43}$$

Consider substituting an function of the form

$$u\left(\boldsymbol{x},t\right) = e^{-bt}w\left(\boldsymbol{x},t\right) \tag{7.44}$$

This gives

$$-be^{bt}w\left(\boldsymbol{x},t\right) + e^{-bt}\frac{\partial w\left(\boldsymbol{x},t\right)}{\partial t} + ke^{-bt}\nabla^2 w\left(\boldsymbol{x},t\right) + be^{-bt}w\left(\boldsymbol{x},t\right) + c = 0 \tag{7.45}$$

or

$$e^{-bt}\frac{\partial w\left(\boldsymbol{x},t\right)}{\partial t} + ke^{-bt}\nabla^2 w\left(\boldsymbol{x},t\right) + c = 0 \tag{7.46}$$

7.2 Elasticity Class

7.3 Fluid Mechanics Class

7.3.1 Burgers' Equation

One Dimensional Analytic Function 1

The one-dimension Burgers' equation in OpenCMISS form can be written as

$$a\frac{\partial u}{\partial t}+b\frac{\partial^2 u}{\partial x^2}+cu\frac{\partial u}{\partial x}=0 \tag{7.47}$$

Adapted from http://eqworld.ipmnet.ru/en/solutions/npde/npde1301.pdf we have for the one-dimension Burgers' equation the analytic solution

$$u\left(x,t\right)=\frac{A+ax}{B+ct} \tag{7.48}$$

To prove the analytic solution we differentiate Equation (7.48) to give

$$\frac{\partial u}{\partial t}=\frac{-c\left(A+ax\right)}{\left(B+ct\right)^2} \tag{7.49}$$

$$\frac{\partial u}{\partial x}=\frac{a}{\left(B+ct\right)} \tag{7.50}$$

$$\frac{\partial^2 u}{\partial x^2}=0 \tag{7.51}$$

$$u\frac{\partial u}{\partial x}=\frac{a\left(A+ax\right)}{\left(B+ct\right)^2} \tag{7.52}$$

$$\tag{7.53}$$

Substiting Equation (7.53) into Equation (7.47) gives

$$a\frac{\partial u}{\partial t}+b\frac{\partial^2 u}{\partial x^2}+cu\frac{\partial u}{\partial x}=\frac{-ac\left(A+ax\right)}{\left(B+ct\right)^2}+b.0+\frac{ac\left(A+ax\right)}{\left(B+ct\right)^2} \tag{7.54}$$

$$=0 \tag{7.55}$$

The analytic field component definitions in OpenCMISS are shown in Table 7.4.

Analytic constant	*Analytic field component*
A	1
B	2

TABLE 7.4: OpenCMISS analytic field components for the one-dimension diffusion equation analytic function 1.

One Dimensional Analytic Function 2

Adapted from http://eqworld.ipmnet.ru/en/solutions/npde/npde1301.pdf we have for the one-dimension Burgers's equation the analytic solution

$$u\left(x,t\right) = aA + \frac{2b}{c\left(x - cAt + B\right)} \tag{7.56}$$

To prove the analytic solution we differentiate Equation (7.48) to give

$$\frac{\partial u}{\partial t} = \frac{2bcA}{c\left(x - cAt + B\right)^2} \tag{7.57}$$

$$\frac{\partial u}{\partial x} = \frac{-2b}{c\left(x - cAt + B\right)^2} \tag{7.58}$$

$$\frac{\partial^2 u}{\partial x^2} = \frac{4b}{c\left(x - cAt + B\right)^3} \tag{7.59}$$

$$u\frac{\partial u}{\partial x} = \frac{-2abA}{c\left(x - cAt + B\right)^2} - \frac{4b^2}{c^2\left(x - cAt + B\right)^3} \tag{7.60}$$

$$\tag{7.61}$$

Substiting Equation (7.61) into Equation (7.47) gives

$$a\frac{\partial u}{\partial t} + b\frac{\partial^2 u}{\partial x^2} + cu\frac{\partial u}{\partial x} = \frac{2abcA}{c\left(x - cAt + B\right)^2} + \frac{4b^2}{c\left(x - cAt + B\right)^3} - \tag{7.62}$$

$$\frac{-2abcA}{c\left(x - cAt + B\right)^2} - \frac{4b^2c}{c^2\left(x - cAt + B\right)^3} \tag{7.63}$$

$$= \frac{2abcA - 2abcA}{c\left(x - cAt + B\right)^2} + \frac{4b^2 - 4b^2}{c\left(x - cAt + B\right)^3} \tag{7.64}$$

$$= 0 \tag{7.65}$$

The analytic field component definitions in OpenCMISS are shown in Table 7.5.

Analytic constant	*Analytic field component*
A	1
B	2

TABLE 7.5: OpenCMISS analytic field components for the one-dimension Burgers' equation analytic function 2.

7.4 Electromechanics Class

7.5 Bioelectrics Class

7.6 Modal Class

7.7 Fitting Class

7.8 Optimisation Class

Chapter 8

Solvers

8.1 Linear Solvers

8.1.1 Linear Direct Solvers

8.1.2 Linear Iterative Solvers

8.2 Nonlinear Solvers

8.2.1 Newton Solvers

Newton Line Search Solvers

Newton Trust Region Solvers

8.2.2 Quasi-Newton Solvers

Broyden-Fletcher-Goldfarb-Shanno (BFGS) Solvers

8.2.3 Sequential Quadratic Program (SQP) Solvers

8.3 Dynamic Solvers

8.4 Differential-Algebraic Equation (DAE) Solvers

8.5 Eigenproblem Solvers

8.6 Optimisation Solvers

Chapter 9

Coupling

There are two general forms of coupling that concern us, which we will term volume-coupling and interface-coupling. A volume-coupled problem is one where several equations are defined over the whole of a region, and communicate over the entirety of that region (although the coupling terms may be zero in certain parts of the domain). An interface-coupled problem concerns those cases where two, or more, separate regions are interacting at a common interace. In 3-d this interface is a surface, and in 2-d it is a line. A typical example of this type of problem is fluid-structure interaction. An example of a volume-coupled problem are double porosity networks, often used in porous flow modelling for geophysics (citation). Clearly, it is conceivable that both types of coupling could exist in a single problem, for example, if one wished to model cardiac electrophysiology, finite deformation solid mechanics and fluid mechanics within the ventricle.

9.1 Volume coupling

Two solution strategies are available to us within OpenCMISS, a partitioned approach and a monolithic approach. A partitioned solution strategy involves solving each separate equation in turn, passing the relevant information from one equation to the next. In certain cases, sub-iteration may be required to ensure that a converged solution is reached. For coupling together only two equations this can be an attractive option, however, it can become unwieldy if many equations are to be coupled together. In contrast, the aim of a monolithic strategy is to solve all the equations together, in one solution process, by computing the matrix form of each equation, and then assembling all of these forms into one large matrix system, which is then passed to the solver e.g. PETSc. Again, it is possible to use both strategies within the same problem, if required. For example, if the time scale of one process is much slower than the others, it may be better to solve this separately, and for a larger time step, than the other equations.

9.1.1 Common aspects

Problem & equation class/type/subtype

For a new coupled problem it is necessary to define appropriate new classes, types & subtypes for the problem and the equations. For a single physics case the problem hierarchy takes the following form (as appropriate):

1. PROBLEM_CLASSICAL_FIELD_CLASS
2. PROBLEM_POISSON_EQUATION_TYPE
3. PROBLEM_LINEAR_SOURCE_POISSON_SUBTYPE

For a coupled system, with equations, PHYSICS_ONE and PHYSICS_TWO the hierarchy should approximately follow:

1. PROBLEM_MULTI_PHYSICS_CLASS
2. PROBLEM_PHYS_ONE_PHYS_TWO_EQUATION_TYPE
3. PROBLEM_PHYS_ONE_PHYS_TWO_EQUATION_STANDARD_SUBTYPE

However, for the equations type hierarchy, it is usually not necessary to use the multi-physics equations class. For example, the types could be declared in the following way for the first equation:

1. EQUATIONS_SET_CLASSICAL_FIELD_CLASS
2. EQUATIONS_SET_PHYS_ONE_EQUATION_TYPE
3. EQUATIONS_SET_PHYS_ONE_COUPLED_PHYS_TWO_EQUATION_SUBTYPE

and for the second equation:

1. EQUATIONS_SET_CLASSICAL_FIELD_CLASS
2. EQUATIONS_SET_PHYS_TWO_EQUATION_TYPE
3. EQUATIONS_SET_PHYS_TWO_COUPLED_PHYS_ONE_EQUATION_SUBTYPE

The obvious implication of the above is that a new PHYS_ONE_PHYS_TWO_EQUATION_ROUTINES.f90 must be created, which in principle will include all the same subroutines as any other equations_routines.f90 file e.g. finite element calculate, equations set setup, problem setup. However, usually only the problem related information needs to be setup in the new file, with the equations specific setup included within the separate equations routines of the component physics (using CASE statements on the subtype to select the appropriate sections of code). More details on this will be given in Sections 9.1.3 & 9.1.4.

Dependent field

For a single physics problem a dependent field will usually possess two field variable types, the *U* variable type (for the left hand side) and the *delUdelN* type (for the right hand side). For a multi-physics problem we must define additional field variable types, typically two for every additional equations set. The list of possible field variable types is given in field_routines.f90 . Whilst, it is sensible to assign additional field variables in ascending order, this is not necessary, excepting that a dependent field must have at least a *U* variable type. Attempting to setup a dependent field with only *V* and *delVdelN* variable types will result in an error.

As there is only a single dependent field, its setup need only occur in the first of the equations set's setup routines, with the remaining equations set merely checking that the correct field setup has occurred. The key subroutine calls that must be made (either within the library for an auto-created field, or within the example file for a user-specified field) are:

```
CALL FIELD_NUMBER_OF_VARIABLES_CHECK(EQUATIONS_SET_SETUP%FIELD,4,&
  & ERR,ERROR,*999)
CALL FIELD_VARIABLE_TYPES_CHECK(EQUATIONS_SET_SETUP\%FIELD, &
  & (/FIELD_U_VARIABLE_TYPE,FIELD_DELUDELN_VARIABLE_TYPE, &
  & FIELD_V_VARIABLE_TYPE,FIELD_DELVDELN_VARIABLE_TYPE/), &
  & ERR,ERROR,*999)
```

Then the field dimension, field data type and field number of components must be set/checked for each variable type that has been defined on the field.

Equations set field

The equations set field is a mandatory data structure that must be defined for all problems. However, if only a single equations set is being used for that subtype, it does not need to be initialised. However, a variable still needs to be defined in the example file, and passed in to the equations set create start:

```
CALL CMISSEquationsSetCreateStart(EquationsSetUserNumberDiffusion,Re
    & CMISSEquationsSetClassicalFieldClass, &
    & CMISSEquationsSetDiffusionEquationType,CMISSEquationsSetMultiC
    & EquationsSetFieldUserNumberDiffusion,EquationsSetFieldDiffusio
    & Err)
```

Need to describe the setup of the equations set field. Number of components, data type etc.

Other fields

Although there is only a single dependent field, we still define separate fields of other types for each equations set, that is, separate materials, source and independent fields, as appropriate for the particular problem under investigation.

9.1.2 Boundary conditions

For each equations set there must be a separate boundary condition object defined as standard. However, it is important to ensure the the correct field variable type is passed into the call to CMISSBoundaryConditionsSetNode, in agreement with the field variable associated with the particular equations set.

Equations mapping

9.1.3 Partitioned scheme specifics

The setup of the partitioned solution scheme is placed in the new PHYS_ONE_PHYS_TWO_EQUATION_ROU file. Specifically, the following subroutines will need to be completed:

1. PHYS_ONE_PHYS_TWO_PRE_SOLVE
2. PHYS_ONE_PHYS_TWO_POST_SOLVE
3. PHYS_ONE_PHYS_TWO_PROBLEM_SETUP
4. PHYS_ONE_PHYS_TWO_PROBLEM_SUBTYPE_SET

The pre and post solve routines will call various specific pre/post-solve routines, such as output data, update boundary conditions etc. These pre/post-solve routines can be those specific to PHYS_ONE and PHYS_TWO, and contained within PHYS_ONE_EQUATION_ROUTINES.f90 and PHYS_TWO_EQUATION_ROUTINES.f90 to eliminate excessive code duplication.

PHYS_ONE_PHYS_TWO_PROBLEM_SETUP defines the various solvers required for each equation. Some key subroutine calls required are:

```
CALL SOLVERS_CREATE_START(CONTROL_LOOP,SOLVERS,ERR,ERROR,*999)
CALL SOLVERS_NUMBER_SET(SOLVERS,2,ERR,ERROR,*999)
CALL SOLVERS_SOLVER_GET(SOLVERS,1,SOLVER_PHYSICS_ONE,ERR,ERROR,*999)
!
!Set properties of solver one...
!
CALL SOLVERS_SOLVER_GET(SOLVERS,2,SOLVER_PHYSICS_TWO,ERR,ERROR,*999)
!
!Set properties of solver two...
!
```

9.1.4 Monolithic scheme specifics

Coupling together equations of differing type/subtype

Not yet attempted.

Coupling together several equations of the same type/subtype

For certain physical problems it is required that several equations of the same type must be coupled in the same domain. This means that the equations_set_subtype is no longer sufficient for distinguishing between the equations, and the specific behaviour one desires for each. It is neceessary in these cases to make use of the equations_set_field structure, which is defined for each equations_set, and contains two integers. The first integer is the id number of that particular equations_set, with the second integer defining the total number of equations_sets of that type. Obtaining the first integer from the equations_set_field within the finite_element_calculate subroutine allows the correct field_variable_type to be identified for interpolation purposes, as well as enabling an automated setup of the dependent field, equations_mappings etc.

The diffusion equation once converted into a finite-element formulation will have a matrix formulation similar to:

$$\mathbf{M}_1 \frac{d\mathbf{c}_1}{dt} + \mathbf{K}_1 \mathbf{c}_1 = \mathbf{f}_1 \tag{9.1}$$

To solve three such diffusion equations monolithically, would result in the following matrix system (before discretisation in time):

$$\begin{pmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_3 \end{pmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix} + \begin{pmatrix} \mathbf{K}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{K}_3 \end{pmatrix} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix} \tag{9.2}$$

Considering a general case of *N* diffusion equations, there will be *N* equations sets, with *N* respective materials fields, to store the diffusion coefficient tensor for each equation. If we now require the equations to be coupled, there will also be ceofficients that determine the strength of this coupling. For each equation set this will be a further N coefficients. Rather than creating a further *N* materials fields, an extra variable type is defined for the existing fields, for simplicity, the *V* variable type. The *V* variable type is set to be a vector dimension type, with N components. With coupling terms the matrix system is:

$$\begin{pmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}_3 \end{pmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix} + \begin{pmatrix} \mathbf{K}_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{K}_3 \end{pmatrix} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix} \tag{9.3}$$

$$+ \begin{pmatrix} -\boldsymbol{\lambda}_{12} - \boldsymbol{\lambda}_{13} & \boldsymbol{\lambda}_{12} & \boldsymbol{\lambda}_{13} \\ \boldsymbol{\lambda}_{21} & -\boldsymbol{\lambda}_{21} - \boldsymbol{\lambda}_{23} & \boldsymbol{\lambda}_{23} \\ \boldsymbol{\lambda}_{31} & \boldsymbol{\lambda}_{32} & -\boldsymbol{\lambda}_{31} - \boldsymbol{\lambda}_{32} \end{pmatrix} \begin{bmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix} \tag{9.4}$$

For a standard diffusion equation there are two dynamic matrices defined, stiffness and damping, which are mapped to the U variable type. To store the coupling contributions from the additional equations, extra matrices must be defined. An array of $N - 1$ linear matrices are defined, which map to the $N - 1$ other field variable types (V, U1, U2, etc). The terms on the diagonal of the global coupling matrix must be included within the stiffness matrix, as it is not possible to have both a dynamic matrix (stiffness) and an additional linear matrix mapping to the same field variable type (the library will be modified in future to allow this, see tracker item 2741 to check progress on this feature).

For a simple implementation of this refer to the example, /examples/MultiPhysics/Coupled-DiffusionDiffusion/MonolithicSchemeTest .

Setting up the solver

One solver, but add multiple solver equations.

Show some example lines for the mappings that need to be defined in the set setup.

In the set_linear_setup, for the setup case(EQUATIONS_SET_SETUP_FINISH_ACTION), the appropriate field variable must be mapped to the particular equations set that is being setup. This requires a certain degree of hard coding and *a priori* knowledge about the field variable types avaiable, and the order in which they will be assigned to equations sets.

First, retrieve the equations set field information and store in a local array.

```
EQUATIONS_SET_FIELD_FIELD=>EQUATIONS_SET%EQUATIONS_SET_FIELD% &
  & EQUATIONS_SET_FIELD_FIELD
```

```
CALL FIELD_PARAMETER_SET_DATA_GET(EQUATIONS_SET_FIELD_FIELD,&
  & FIELD_U_VARIABLE_TYPE,FIELD_VALUES_SET_TYPE, &
  & EQUATIONS_SET_FIELD_DATA,ERR,ERROR,*999)
imy_matrix = EQUATIONS_SET_FIELD_DATA(1)
Ncompartments = EQUATIONS_SET_FIELD_DATA(2)
```

Here arrays the store the mapping between the equations set id and the field variable type are allocated and initialised, for use when setting the mappings.

```
CALL EQUATIONS_MAPPING_LINEAR_MATRICES_NUMBER_SET &
   & (EQUATIONS_MAPPING, Ncompartments-1,ERR,ERROR,*999)

ALLOCATE(VARIABLE_TYPES(2*Ncompartments))
ALLOCATE(VARIABLE_U_TYPES(Ncompartments-1))
DO num_var=1,Ncompartments
  VARIABLE_TYPES(2*num_var-1)=FIELD_U_VARIABLE_TYPE+ &
   & (FIELD_NUMBER_OF_VARIABLE_SUBTYPES*(num_var-1))
  VARIABLE_TYPES(2*num_var)=FIELD_DELUDELN_VARIABLE_TYPE+ &
   & (FIELD_NUMBER_OF_VARIABLE_SUBTYPES*(num_var-1))
ENDDO
num_var_count=0
DO num_var=1,Ncompartments
  IF(num_var/=imy_matrix)THEN
    num_var_count=num_var_count+1
    VARIABLE_U_TYPES(num_var_count)=VARIABLE_TYPES(2*num_var-1)
  ENDIF
ENDDO
```

Finally, set the mapping for the dynamic variable (as standard for the diffusion equation), the linear variable types (for the coupling matrices), the right hand side variable (i.e. delUdelN types) and the source variable. Note that the source variable that is mapped is always the FIELD_U_VARIABLE because we use separate source fields (containing only a U variable) for each equations set.

```
CALL EQUATIONS_MAPPING_DYNAMIC_VARIABLE_TYPE_SET &
```

```
 & (EQUATIONS_MAPPING, VARIABLE_TYPES(2*imy_matrix -1),ERR,ERROR,*999)
CALL EQUATIONS_MAPPING_LINEAR_MATRICES_VARIABLE_TYPES_SET &
 & (EQUATIONS_MAPPING, VARIABLE_U_TYPES ,ERR,ERROR,*999)
CALL EQUATIONS_MAPPING_RHS_VARIABLE_TYPE_SET
 & (EQUATIONS_MAPPING, VARIABLE_TYPES(2*imy_matrix ),ERR,ERROR,*999)
CALL EQUATIONS_MAPPING_SOURCE_VARIABLE_TYPE_SET &
 & (EQUATIONS_MAPPING, FIELD_U_VARIABLE_TYPE ,ERR,ERROR,*999)
```

For a monolithic solution step there is only a single solver equation to setup, but several equations set equations to map into this. In the example file this is done in the following way:

```
DO icompartment=1,Ncompartments
  CALL CMISSSolverEquationsEquationsSetAdd(SolverEquationsDiffusion,&
     & EquationsSetDiffusion(icompartment),&
     & EquationsSetIndex,Err)
ENDDO
```

9.1.5 IO

Currently field_IO_routines does not support the export of fields containing field variable types other than U and delUdelN. Therefore, it is necessary to use the FieldML output routines, and output separate files, one for each variable type, using calls of the following form:

```
CALL FieldmlOutput_AddField(fieldmlInfo,baseName//".dependent_U",&
  & region, mesh, DependentField,CMISSFieldUVariableType, err )

CALL FieldmlOutput_AddField(fieldmlInfo,baseName//".dependent_V",&
  & region, mesh, DependentField,CMISSFieldVVariableType, err )
```

9.2 Interface coupling

9.3 Theory

9.3.1 Dirichlet Boundary Condition

Introduction to Dirichlet boundary condition

Dirichlet boundary condition is a type of essential boundary condition where the value of the variable in an equation does not change at a particular computational node - for e.g. pressure or temperature. In contrast, a natural boundary condition can be flux in diffusion problem or stress in linear elasticity. Dirichlet boundary condition is enforced as a fixed specification of values at a set of boundary nodes. This type of boundary condition is implemented by modifying the set of equations as opposed to the right hand side of the equation. In a finite element framework, by enforcing the Dirichlet boundary condition an equation matrix which has been formed by the test function w is replaced, to give a new set of equations.

The Dirichlet boundary condition being imposed on a particular nodal value implies that the specific row of the equation matrix corresponding to that particular node is independent of the rest of the nodes and hence can be eliminated and the matrix simplified in the process. For example, if the weak form of any of the problems is,

$$\begin{pmatrix} M_{11} & M_{12} & \dots & M_{1N} \\ M_{21} & M_{22} & \dots & M_{2N} \\ M_{31} & M_{32} & \dots & M_{3N} \\ \vdots & \vdots & \ddots & \vdots \\ M_{n1} & M_{n2} & \dots & M_{NN} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

Now, if you would like to impose a Dirichlet boundary condition on the first node only i.e., on ϕ_1 as C, for examples as in the Laplace equation where Dirichlet boundary condition is implemented on the first and last node only,

$$\begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_n \end{pmatrix} = \begin{pmatrix} C \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

which modifies the equation matrix as,

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ M_{21} & M_{22} & \dots & M_{2N} \\ M_{31} & M_{32} & \dots & M_{3N} \\ \vdots & \vdots & \ddots & \vdots \\ M_{n1} & M_{n2} & \dots & M_{NN} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_n \end{pmatrix} = \begin{pmatrix} C \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

We can perform another elimination and rewrite the equation as,

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & M_{22} & \dots & M_{2N} \\ 0 & M_{32} & \dots & M_{3N} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & M_{n2} & \dots & M_{NN} \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_n \end{pmatrix} = \begin{pmatrix} C \\ b_2 - CM_{21} \\ b_3 - CM_{31} \\ \vdots \\ b_n - CM_{N1} \end{pmatrix}$$

Now we can store a reduced matrix as,

$$\begin{pmatrix} M_{22} & M_{23} & \dots & M_{2N} \\ M_{32} & M_{33} & \dots & M_{3N} \\ \vdots & \vdots & \ddots & \vdots \\ M_{n2} & M_{n3} & \dots & M_{NN} \end{pmatrix}$$

and the right hand side is now,

$$\begin{pmatrix} C \\ (b_2 - CM_{21}) \\ (b_3 - CM_{31}) \\ \vdots \\ (b_n - CM_{N1}) \end{pmatrix}$$

Therefore to calculate the right hand side of the equation numerically it is required to store the degrees of freedom associated with the node on which Dirichlet condition is enforced. The implementation of this step is done in OpenCMISS in the *boundary conditions routines* and is explained in detail in the following section.

Dirichlet boundary condition in OpenCMISS

In this section we give a short description of some of the routines and OpenCMISS coding details required to modify a part of the code related to boundary conditions.

In an example file, for example Laplace, the boundary condition is implemented in the following steps,

- Initializing an object *CmissBoundaryconditiontype*.
- `EQUATIONS_SET_BOUNDARY_CONDITIONS_CREATE_START`

  - As an overview this call stack passes through the *EQUATIONS SET SETUP* (for e.g. Classical field → Laplace → Standard Laplace setup) to use the information of the pointer to the particular 'equation set' and return a pointer to the created boundary condition.
- In general in the next calls we evaluated on which nodes the boundary condition is to be set. The routine
  - `DECOMPOSITION_NODE_DOMAIN_GET`
    - implements a tree search to get a nodal value
  - `BOUNDARY_CONDITIONS_SET_NODE`
    - specifies the type of boundary condition on the particular node.
- `EQUATIONS_SET_BOUNDARY_CONDITIONS_CREATE_FINISH`
  - `EQUATIONS_SET_SETUP`

    - sets up the specific details of the equation set by using the

    `EQUATIONS_SET_SETUP_INFO`
  - `EQUATIONS_SET_BOUNDARY_CONDITIONS_GET`

- Obtains boundary condition information for a particular type of equation set (Classical field → Laplace)

– `EQUATIONS_SET_BOUNDARY_CONDITIONS_CREATE_FINISH`

In *boundary conditions routine* the Dirichlet boundary condition is imposed by first calculating the Dirichlet degrees of freedom. Then the row-column information of the equations matrix is obtained by obtaining information about the distributed matrix, more specifically the:

∗ *DISTRIBUTED MATRIX STORAGE TYPE*

∗ *DISTRIBUTED MATRIX STORAGE LOCATIONS*

In OpenCMISS the sparse distributed matrix is stored in a format called Compressed Row Storage (CRS) format (more details given in the next section). There are several other possibilities such as a column based approach: a Compressed Column Storage (CCS) format or a Block Storage type which have not been implemented yet.

In our current structure the equation matrix is stored in the row based format i.e., CRS. To implement a Dirichlet boundary condition however, the information of the degrees of freedom are required which are stored in the columns of the matrix. Therefore it is useful to store the matrix in a column based approach in order to avoid redundant looping over all rows. Currently a linked list based approach has been proposed which stores the matrix in a CRS format as well as a linked list format when a Dirichlet boundary condition is imposed on the problem. In order reduce the memory usage of storing the equation matrix in two different data formats we store only the degrees of freedom of the matrix that correspond to the nodes in the boundary. The boundary nodes are calculated using a parameter mapping function of the local columns obtained using the *global to local mapping* of columns of the equation matrix. In this routine the column-row information of the matrix are obtained from a linked list and the degree of freedom are calculated and stored to be used in the calculation of the right hand side of the equation.

More on matrix storage

The Compressed row storage (CRS) is a storage format which stores the nonzero elements of a matrix sequentially. The storage algorithm is outlined by the following example. For a matrix of the form,

$$\begin{pmatrix} 7 & 0 & -3 & 0 & -1 & 0 \\ 2 & 8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ -3 & 0 & 0 & 5 & 0 & 0 \\ 0 & -1 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & -2 & 0 & 6 \end{pmatrix}$$

The compressed row storage or CRS is given by a pointer to the first element of row i of A and a one dimensional array of the column indices,

$$\begin{pmatrix} row\,indices \rightarrow & 1 & 4 & 7 & 9 & 11 & & & & & & & \\ column\,indices \rightarrow & 1 & 3 & 5 & 1 & 2 & 3 & 1 & 4 & 2 & 5 & 4 & 6 \\ values \rightarrow & 7 & -3 & -1 & 2 & 8 & 1 & -3 & 5 & -1 & 4 & -2 & 6 \end{pmatrix}$$

For a row i, the number of non-zero elements is easily obtained from $row\,indices(i+1)$-$row\,indices(i)$. For more example on the implementation refer to the *matrix vector routines* in OpenCMISS repository.

9.3.2 Neuman Boundary Condition

9.3.3 Robin Boundary Condition

Chapter 10

Developers' Document

10.1 Introduction

This chapter is intended for new and existing developers of OpenCMISS. It contains tips from the developers who previously encountered the learning curve, and are now trying to reduce it for those who are new to OpenCMISS. New developers are encouraged to use this chapter written in an informal narrative style as an independent study guide to get up to speed with the codebase. The sections are ordered in an increasing level of difficulty, which introduce the basic concepts first and then progress through to more advanced features of the code. Care was taken not to introduce too much information at once – as such, it may at times appear to lack rigor, but after reading this chapter developers will be empowered to answer their own questions. In addition, existing developers of OpenCMISS will find that this chapter may serve as a reference to assist day-to-day development work, and to keep up-to-date with extensions that are made in the core library functionalities.

10.2 The Anatomy of an Example File

Let's assume for the sake of discussion that this is your first encounter with OpenCMISS code. If not, simply skip to the next section. As you may already know, OpenCMISS source code is split up into two major parts, one which provides the core library functionality, like evaluating a basis function or solving a pde with finite element method, and one which solves a particular problem by using these library routines. The source files (fortran .f90 and some .c files) for the library routines are entirely contained within `/cm/src`, whereas the example files can be found under `/cm/examples`.

As a new developer, a good place to attack the 300,000+ lines of source code is to start at an example file because it gives a good bird's eye view. for historical reasons (it was the first to be set up and the most "proper") the Laplace example is often used as a showcase. So let's go ahead and fire up the following file in your favourite editor (In Linux Kate, emacs or gedit work well. In Windows, maybe emacs, Kate or Notepad++ are okay):
`/cm/examples/ClassicalField/Laplace/Laplace/src/LaplaceExample.f90`.

Later in this chapter we will address the finer details of this example file, however, for now we'll look at the general outline and flow. Scroll down and have a brief look - after all the variable declarations, there should be a call to

```
CALL CMISSInitialise(...)
```

All OpenCMISS routines calls are made after this line, since this routine tells the library that we are ready to start using it. Similarly, near the end of the file there is a call to

```
CALL CMISSFinalise(...)
```

which initiates the finalising of objects which had been created by OpenCMISS throughout the execution.

Of course, what comes in between these calls does all the interesting stuff. It's about 200 lines of solid blocks of code, but there is an easier way to read this - most tasks in an example file are arranged in blocks, which looks like

```
CALL CMISS****TypeInitialise(...)
CALL CMISS****CreateStart(...)
...
CALL CMISS****CreateFinish(...)
```

where **** denotes the type of the object such as coordinate system, region, basis etc. The initialise call usually creates a space in memory for the object and perhaps assign the default values for some of its members. If you forget to issue this call, the executable may or may not throw up an error that says "**** is already associated." depending on whether the developer has written code to check it. Using an object without initialising it may lead to some subtle memory problems. You have been warned. Between CreateStart and CreateFinish, we basically call routines that assign properties to shape and mould this specific instance of the object. It's only when the CreateFinish call is issued, that OpenCMISS oils up its gears and actually gets to work. Thus it is possible that you may have entered conflicting arguments, but the error may not occur until the CreateFinish is called. Because a lot happens in CreateFinish, it's usually the place that you might want to put a stop flag in your debugger (which will be discussed later).

So, armed with the above knowledge, most of the example file can be broken down into these blocks:

```
CoordinateSystem
Region
Basis
Mesh/GeneratedMesh
Decomposition
Field
EquationsSet
BoundaryConditions
Problem
Solver
```

From these, you probably won't touch CoordinateSystem (except for changing dimensions), Region and Decomposition because, well, there's nothing much to change unless you're doing something quite advanced. This leaves for an average Joe developer/user the following bits to tinker with: Basis, Mesh, Field, EquationsSet, BoundaryConditions, Problem and Solver.

Basis objects are required for all finite element problems, which currently is the only solution method type implemented. A mesh object holds the geometry and mapping information. Any kind of numerical data that you might want to hold in a vector or matrix, such as the dependent (unknown) variables, material parameters or the nodal coordinates themselves are neatly packaged into the Field type object, which has several variants. These objects are described in further detail in section 10.11.4, but for now we will crack on with this introduction. EquationsSets, Solver and BoundaryConditions objects are so big and important that they have their own designated sections elsewhere in this document (10.11.5, 10.4 and 10.7). This might leave you wondering what the role of the Problem object is - this one manages the control loops, which is a general way to handle linear/nonlinear/steady/time-dependent problems. The Problem objects also holds meta information regarding what equations are being solved, including coupled physics problems that have been introduced recently.

If you're a keen developer and you have peeked ahead at any of the library source files, you might have noticed that they look quite a bit different from the example source file. Every OpenCMISS routine called from the example file begin with `CMISS...` The reason for this is because the example file may only use the library through OpenCMISS *bindings*, or *application programming interface* rather than directly calling the routines from the core library itself. This layer of separation is a pretty standard thing and it protects the user from working directly with the object pointers which may be dangerous. All binding routines are implemented in the file `/src/opencmiss.f90` which is the only module we include via the call

```
USE OPENCMISS
```

at the top of the example file. When you start developing user-callable library routines, you will have to also write (and maintain!) the bindings if they're to be used in the example file.

Once you start to modify the code yourself, there will invariably be errors. That's okay. What you should know though is how OpenCMISS handles errors. When there is an error in the library routine, in most cases OpenCMISS won't exit straight away with an all-too familiar message like "segmentation fault" but takes a more graceful approach. This is great for users of the library, but as a developer it can take a little while to pinpoint exactly at which line the error has occurred. The default error handling behaviour is to output the error and continue execution, which, for a scientific code like this usually leads to more errors. This behaviour can be changed via

```
CALL CMISSErrorHandlingModeSet(CMISSTrapError,Err)
```

which forces the program to stop after the error message has been printed.

While we're on the subject of bug-hunting, let's address the issue of viewing variables. There are a couple of different approaches one can use to check on the value of variables mid-execution. The first is to use a debugger like TotalView, which isn't free but is worth every penny. The other way is to go old-school and put `WRITE(*,*)` all over the source file (Don't forget to remove this before committing). This approach involves you having to re-compile the entire library which is quite time-consuming. Also, because most data you will be interested in are encapsulated under extensive object structures, it may require some time to figure out exactly what to print out.

Having TotalView installed also helps with looking at routines. At this point we will break through the surface of the example file and go under. Let's take the error handling mode setting routine from above. To follow it down, open up `/src/opencmiss.f90` and Ctrl-F for the routine. Between the `ENTERS` and `EXITS` routines (which will be described later) you will see that the binding routine simply makes a call to the actual routine which does the work:

```
CALL CMISS_ERROR_HANDLING_MODE_SET ( . . . )
```

This subroutine is not defined in `opencmiss.f90`, as it only contains the bindings. The convention in OpenCMISS code is to prefix every routine name with the module name, which, in this case is `CMISS`. You can now open up `/src/cmiss.f90` and search again for the routine. If you hate having to connect the dots every step of the way in this fashion, you can fire up the example in TotalView and simply double click on the routine names to dive into them.

You should now have a good background to start modifying or setting up your own example files. A large part of doing this involves copy & pasting an existing example and modifying them to fit your own problem (be sure to use the `svn cp` command to avoid nagging emails). In this case, you might end up spending a lot of time figuring out what arguments a certain function should be called with. For example, you might want to change the type of matrix storage from Full to Sparse. The binding routine that sets this is called

```
CMISSEquationsSparsityTypeSet(Equations , SparsityType , Err)
```

The second argument, which is what you want to change, is meant to be selected from a set of named constants. How do I know this? It's obvious after a while, but you can Ctrl-F for this routine in `opencmiss.f90`. There, you will see in the comment next to `SparsityType` it will say `see OPENCMISS_EquationsSparsityTypes`. These constants are also defined within `opencmiss.f90` so Ctrl-F it again and it will take you to the near the top of the file where

```
CMISSEquationsSparseMatrices
CMISSEquationsFullMatrices
```

are defined. The whole thing can also be done by looking at the developer's page:
`http://cmiss.bioeng.auckland.ac.nz/OpenCMISS/doc/doxygen/html/` if

you have a lot of patience that is.

Lastly, if you don't know already, learn how to search within full directory of files for a keyword in your editor. Because Fortran doesn't have a great IDE (integrated development environment), this is unfortunately the fastest way to find information you're after.

10.3 Data Model in OpenCMISS

As outlined in the previous section, the example file manages two major tasks. The first is all about the **data**, that is, to do with defining and assigning various bits of information like mesh coordinates, material parameters and so on. The second is all about **doing**, like setting up and calling the solver, for example.

By *data model*, I am simply referring here as to how these different bits of numbers are stored, passed around, and accessed. If I asked you right now to find, say, 'the y-coordinate of node 7', you'll probably notice that it's not very straightforward since all numbers have been locked away inside the OpenCMISS data structures. Indeed this is a source of frustration and makes the learning curve steep. Why is this so complicated? The answer is simple - it's because you haven't read this section yet. Oh and also having the formal structures to support data handling makes the parallel coding much easier, even if it does cause some overhead in the code writing.

The key to understanding the OpenCMISS data model is to get to grips with the hierarchy of data objects, which we'll go through now. We'll do this pretty quickly now and come back later to look at the details. Put very simply,

TOP	FIELD
MIDDLE	VARIABLE, MESH
BOTTOM	BASIS

That's it. Easy eh? Now the details: The thing to remember is, OpenCMISS has its own shelves it packs away data onto, and at times it will seem inflexible. In some cases you might be right, but do you want to learn this or not?! Moreover, these shelves (objects) have names which you may already associate in your mind with something else – don't let it throw you off. Be strong. In the below list, pay attention to the indentation for the hierarchical order.

1. In terms of data, FIELD is the top structure. Think of this as a continuous spatial distri-

bution of numbers, to be discretised by the structures below in the hierarchy. Note there are different types of fields, like geometric or materials field. This list applies mainly to the dependent field, which is another name for the unknown variables of the problem.

- VARIABLES The term variable here relates closer to a mathematical variable, not a code variable. In the dependent field, it makes sense to group different variables under one field object. This is like keeping displacement and velocity under a common field object.
  - COMPONENTS In the above example, each displacement and velocity could each have three different components in three dimensions. The data structure allows independent management of the components. It's quite general - you can use different order basis functions for X,Y or Z component, for example.
    - PARAM_TO_DOF_MAP Here the word 'PARAM' refers to a node or grid point or gauss point etc, depending on whichever way you decided to interpolate the variable. If you chose a node-based interpolation, 'node_param2dof_map' will tell you exactly where in the dof (degrees of freedom) vector the unknown of this component of this variable of this field maps to.
  - PARAMETER_SETS Even though for a given variable, components can be handled separately, all numbers of all components for a variable are actually dumped into one long vector. This vector is called the parameter set, because it contains the parameters for the chosen interpolation scheme. Underneath the bonnet, this vector is powered by the distributed_vector_class which helps with parallel communication of data.
    - PARAMETER_SET_TYPE You might want to store different type of data, even for a given field variable. For example, you might want to store the displacement at the last time step as well the displacement. Obviously the two data sets belong to the same variable and components, so what you have to do is to allocate another vector of the same size to put your data into. This is easily achieved by creating an additional parameter set and assigning it an appropriate (and different) parameter set type, so that OpenCMISS will handle its storage. This way, all your data are close by and you don't have to worry about creating and passing various different vectors around the

code.

2. The field is continuously varying, but we characterise it by a discrete vector of numbers. This should hint to you that there is a MESH involved in this whole business – you're right. And you figured it out all on your own.

   •

Did you notice how parameter_sets contain the data, while param_to_dof_map contains its mapping into arrays? The structure above separates data from bookkeeping.

10.4 Solver Object

10.5 PETSc and OpenCMISS

10.6 Overview of Finite Element Routines

10.7 Boundary conditions

10.8 Time Integrations

10.9 Parallel Execution

10.10 HECToR

10.11 Description of OpenCMISS Objects

10.11.1 Basis Object

10.11.2 Mesh Object

10.11.3 Domain Object

10.11.4 Field Object

10.11.5 EquationsSet Object

10.11.6 Decomposition Object

10.12 CMISS Conventions, Bits and Bobs

References

[CGSMVC10] D Chapelle, J-F Gerbeau, J Sainte-Marie, and I E Vignon-Clementel. A poroelastic model valid in large strains with applications to perfusion in cardiac modeling. *Computational Mechanics*, 46(1):91–101, 2010.

[Chu10] T. J Chung. *Computational fluid dynamics*. Cambridge University Press, Cambridge; New York, 2010.

[Cle81] David L Clements. *Boundary Value Problems Governed by Second Order Elliptic Systems*. Pitman Publishing Limited, London, 1981.

[Cou04] Olivier Coussy. *Poromechanics*. John Wiley & Sons, West Sussex, 2004.

[GSE00] P. M Gresho, Robert L Sani, and M. S Engelman. *Incompressible flow and the finite element method. Vol. 1, Advection-diffusion*. Wiley, Chichester, 2000.

[HP99] Juan C Heinrich and D. W Pepper. *Intermediate finite element method : fluid flow and heat transfer applications*. Taylor & Francis, Philadelphia, PA, 1999.

[Hut04] D. V. Hutton. *Fundamentals of Finite Element Analysis*. McGraw-Hill, 2004.

[MH02] A Masud and T J R Hughes. A stabilized mixed finite element method for Darcy flow. *Computer Methods in Applied Mechanics and Engineering*, 191(39-40):4341–4370, 2002.

[NLGSH91] P M F Nielsen, I J Le Grice, B H Smaill, and P J Hunter. Mathematical model of geometry and fibrous structure of the heart. *Am. J. Physiol.*, 260(Heart Circ. Physiol. 29):H1365–H1378, 1991.

[PP94] J Petera and J F T Pittman. Isoparameteric Hermite elements. *Int. J. Numer. Methods Eng.*, 37:3489–3519, 1994.

[ZT06] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method Set, Sixth Edition*. Butterworth-Heinemann, 6 edition, January 2006.