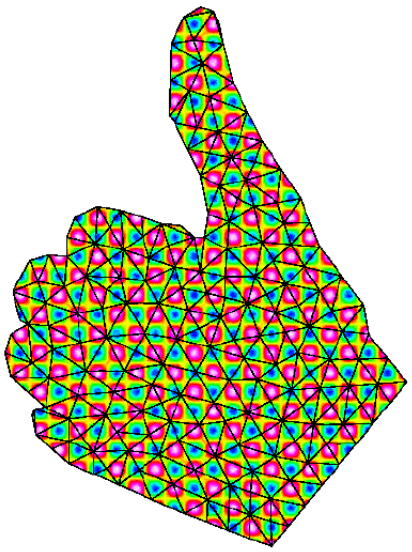
Finite Element Methods

MSc Course

Dr Arash Hamzehloo

Department of Aeronautics
Imperial College London



Contents

Aims

To provide a solid foundation of the fundamentals of Finite Element methods in fluid dynamics and structural mechanics and a basic understanding of their advantages and limitations.

Objectives

At the end of this course you should be able:

- To state appropriate model equations for different types of fluid dynamics and structural mechanics problems.
- To derive the **weak formulation** of a boundary value problem (BVP) and solve it using the Galerkin method.
- To illustrate methods of calculating shape functions for use as element displacement approximations for 1-D, 2-D, and 3-D finite elements.
- To define Dirichlet and Neumann boundary conditions and explain how to enforce these boundary conditions within the Galerkin finite element framework.
- To explain how to use an elemental decomposition with numerical integration and differentiation to construct the global matrix system arising from the Galerkin finite element technique.
- To be aware of problems discretising time dependent problems and the application of mass lumping and Petrov-Galerkin concepts.
- To be aware of the p type extension of the finite element method.

Books

- *The Finite Element Method: Linear Static and Dynamics Finite Element Analysis*, Thomas J.R. Hughes, Dover Press.
- *The Finite Element Methods*, O.C. Zienkiewics and R.K.Taylor, McGraw Hill (One of the classics, has recently been updated, three volumes).
- *Spectral/hp Element Methods for CFD*, G.Em. Karnidakis and S.J. Sherwin, 2005, Oxford University Press (For spectral element/p-type methods in fluids).
- *Finite Elements and Fast Iterative Sovlers*, G H. Elman, D. Silvester and A. Wathen, 2005, Oxford University Press.
- *Finite Element Procedures in Engineering Analysis*, Bathe, K. J., Prentice Hall.
- *Finite Element Programming*, Hinton E. & Owen R., Academic Press.
- *Concepts and Applications of Finite Element Analysis*, Cook, R. D., Malkus, D., Plesha, M., Fourth Edition, J. Wiley.
- *Finite Elements and Solution Procedures*, Crisfield, M. A., 1986, Pineridge Press.
- *An analysis of the finite element methods*, G. Strang and G.J.Fix, 1973, Prentice-Hall (More approachable mathematics book).
- *Incompressible Flow and the Finite Element Method* P.M.Gresho and F.L. Sani, 1998, Wiley (Very complete but idiosyncratic and quite expensive).
- *The Mathematical Theory of Finite Element Methods* S.C. Brenner and L. R. Scott, 1996, Springer (For a more mathematical approach).

Foreword

The finite element method was originally developed by Turner *et al.* in 1956 within the structural mechanics community. The term was originally coined by Clough in 1960. The technique is however also now applied within the fluid dynamics area. Although the foundation and application of the technique is the same within both fields the terminology can be distinctly different. The present lecture notes have been adapted from previous versions taught by Prof. Spencer Sherwin, Dr Martin Vymazal and Dr Yorgos Deskos.

1 Fundamental and Model Equations

In representing any physical process computationally the three important steps are

1. Problem definition
2. Mathematical model
3. Simulation

- We first need to define our problem of interest and the quantities which we would like to measure so that we have a well-posed problem. In general, we may not know the exact equations which govern our problem although for many fluid applications the Navier-Stokes equations are sufficient.
- We then need an appropriate mathematical model. Although the Navier-Stokes equations may govern our process there is typically a more appropriate model equation such as in potential flow.
- Finally when we can go ahead and simulate this mathematical model on a computer.

1.1 Governing Fluids Equations: Navier-Stokes

1.1.1 Compressible Conservative form

For fluid dynamics the governing equations are the Navier-Stokes equations which may be derived from first principles using Newton's laws and the conservation of mass and energy. The full equations can be written in so called *conservation form* and using vector notation as:

$$\mathbf{U}_t + \mathbf{F}_{x,x} + \mathbf{F}_{y,y} + \mathbf{F}_{z,z} = \mathbf{G}_{x,x} + \mathbf{G}_{y,y} + \mathbf{G}_{z,z} \tag{1}$$

with the vectors of unknowns and inviscid fluxes given by

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{bmatrix} \quad \mathbf{F}_x = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho uH \end{bmatrix} \quad \mathbf{F}_y = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ \rho vH \end{bmatrix} \quad \mathbf{F}_z = \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ \rho wH \end{bmatrix}$$

and the viscous fluxes are

$$\mathbf{G}_x = \begin{bmatrix} 0 \\ \sigma_{xx} \\ \sigma_{xy} \\ \sigma_{xz} \\ (\boldsymbol{\sigma}\cdot\mathbf{V})_x + kTx \end{bmatrix} \quad \mathbf{G}_y = \begin{bmatrix} 0 \\ \sigma_{yx} \\ \sigma_{yy} \\ \sigma_{yz} \\ (\boldsymbol{\sigma}\cdot\mathbf{V})_y + kTy \end{bmatrix} \quad \mathbf{G}_z = \begin{bmatrix} 0 \\ \sigma_{zx} \\ \sigma_{zy} \\ \sigma_{zz} \\ (\boldsymbol{\sigma}\cdot\mathbf{V})_z + kTz \end{bmatrix}$$

where

$$\begin{aligned} (\boldsymbol{\sigma}\cdot\mathbf{V})_x &= u\sigma_{xx} + v\sigma_{xy} + w\sigma_{xz} \\ (\boldsymbol{\sigma}\cdot\mathbf{V})_y &= u\sigma_{yx} + v\sigma_{yy} + w\sigma_{yz} \\ (\boldsymbol{\sigma}\cdot\mathbf{V})_z &= u\sigma_{zx} + v\sigma_{zy} + w\sigma_{zz} \end{aligned}$$

k is the coefficient of thermal conductivity and $\boldsymbol{V} = [u, v, w]^T$. The stress tensor $\boldsymbol{\sigma}$ for a Newtonian fluid in thermodynamic equilibrium is given by

$$\boldsymbol{\sigma}_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}(\nabla\cdot\boldsymbol{V})\delta_{ij}\right).$$

- We note that the top equation in (1) is the mass conservation law.

- This is a too complicated starting point to consider directly and so we need to consider the key mathematical ingredients of this system of equations by looking at *one* and *two* dimensional (usually scalar) equations.
- However do not lose sight of the ultimate objective of solving the Navier-Stokes equations or a related model equation.

1.1.2 Incompressible and Non-Conservative forms

For a finite volume method the conservative form is particularly attractive since the numerical scheme makes direct use of the physical properties associated with this form. However, traditionally finite elements have been applied to the incompressible non-conservative form. In the limit of steady, very low Reynolds number flow these equations are also analogous to the plane stress problem in elasticity.

Indeed, you may be more familiar with some of the equations contained in (1) in non-conservative form. For example, the mass conservation equation may be written:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) = \frac{\partial \rho}{\partial t} + \rho \nabla \cdot \boldsymbol{V} + \nabla \rho \cdot \boldsymbol{V} = 0$$

and so we have

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \boldsymbol{V} = 0.$$

- $D\rho/Dt$ is the material derivative of density (i.e. moving with the fluid).
- For an incompressible fluid $D\rho/Dt = 0$ and so this form of the equation leads to the well known condition

$$\nabla \cdot \boldsymbol{V} = 0. \tag{2}$$

- Also in incompressible flows we can decouple the energy equation and the remaining momentum equation can be written in non-conservative form as:

$$\rho \frac{\partial \boldsymbol{V}}{\partial t} + \rho(\boldsymbol{V} \cdot \nabla)\boldsymbol{V} = -\nabla p + \mu \nabla^2 \boldsymbol{V} \tag{3}$$

- Equations (2) and (3) represent a system with four variables as opposed to the full conservative form given in equation (1) which contained five variables.

- Therefore we see that choosing the correct model equation is important since it can lead to great computational savings!

1.2 Model equations arising from the Navier Stokes equations

If we consider the x-component momentum equation (non-conservative form):

$$\rho\frac{\partial u}{\partial t} + \rho(\mathbf{V}\cdot\nabla)u = -\frac{\partial p}{\partial x} + \mu\nabla^2 u \qquad \text{where } \boldsymbol{V} = [u, v, w]^T \tag{4}$$

This can be non-dimensionalised by setting:

$$\bar{x} = \frac{x}{L}, \quad \bar{u} = \frac{u}{U_\infty}, \quad \bar{v} = \frac{v}{U_\infty}, \quad \bar{t} = \frac{tU_\infty}{L} \text{ and } \bar{p} = \frac{p}{\rho U_\infty^2}$$

where L and U_∞ are the characteristic length and velocity scale. Changing variables in equation (4) we obtain the non-dimensional equation:

$$\frac{\partial \bar{u}}{\partial \bar{t}} + (\bar{\mathbf{V}}\cdot\nabla)\bar{u} = -\frac{\partial \bar{p}}{\partial \bar{x}} + \frac{\nu}{U_\infty L}\nabla^2\bar{u} \tag{5}$$

- From this non-dimensionalisation we are not surprised to see that the equation is dependent on the single parameter the Reynolds number $Re = U_\infty L/\nu$.

- If Re is small we argue that we the term premultiplied by $1/Re = \nu/(U_\infty L)$ dominates and so we can ignore the non-linear term $(\bar{\boldsymbol{V}}\cdot\nabla)u$ and we have:

$$\frac{\partial \bar{u}}{\partial \bar{t}} = -\frac{\partial \bar{p}}{\partial \bar{x}} + \frac{1}{Re}\nabla^2\bar{u}. \tag{6}$$

  This equation combined with equation (2) are the Stokes equations. If we consider p as stress and v as displacement this system is the plane stress equations.

- Now, at steady state $\partial\bar{u}/\partial\bar{t} = 0$ and so for small Re we have

$$0 = -\frac{\partial \bar{p}}{\partial \bar{x}} + \frac{1}{Re}\nabla^2\bar{u} \qquad \Rightarrow \qquad \nabla^2\bar{u} = Re\frac{\partial \bar{p}}{\partial \bar{x}}$$

  which is a Poisson equations in the variable u.

- Alternatively, we can consider the other Reynolds number limit (Re large) then far away from the boundary layers where we can ignore the viscous terms $\nabla^2\bar{u}$ and equation (5) becomes:

$$\frac{\partial \bar{u}}{\partial \bar{t}} + (\bar{\mathbf{V}} \cdot \nabla)\bar{u} = -\frac{\partial \bar{p}}{\partial \bar{x}}$$

  which in 1D reduces to

$$\frac{\partial \bar{u}}{\partial \bar{t}} + \bar{u}\frac{\partial \bar{u}}{\partial x} = -\frac{\partial \bar{p}}{\partial \bar{x}}$$

  which is the one-dimensional Burgers equation when $\frac{\partial \bar{p}}{\partial \bar{x}} = 0$ and is mathematically similar to the 1D advection equation

$$\frac{\partial \phi}{\partial t} + a\frac{\partial \phi}{\partial x} = 0$$

  which we will return to later on in the course.

- Finally, if the flow is irrotational which usually means away from boundary layers and shocks the flow can be represented as a velocity potential, ϕ such that $\boldsymbol{V} = \nabla \cdot \phi$. Then using applying equation (2) we obtain:

$$\begin{aligned} \nabla \cdot \boldsymbol{V} = \nabla \cdot \nabla \cdot \phi &= 0 \\ \nabla^2 \phi &= 0 \end{aligned}$$

  which is Laplace's equation and is used widely in Marine computations, acoustic modeling and incompressible, inviscid aerodynamics

• We will be using Laplace's and Poisson's equation as our model problem in the first part of the lecture course. We will also return to the heat and linear advection equations as model problems when we discuss fluid related applications later in the course in section 5

1.3 Mathematical Classification and Boundary Conditions

Before proceeding to a full example we must first visit the mathematical definition of the model equations we will consider and the type of boundary conditions that are associated with the classification.

1.3.1 Classification of Model Equations

The mathematical definitions of a partial differential equation can be stated as follows. Considering a general second order partial differential equation (PDE) of the form:

$$a(x,y)\frac{\partial^2\phi}{\partial x^2} + b(x,y)\frac{\partial^2\phi}{\partial x \partial y} + c(x,y)\frac{\partial^2\phi}{\partial y^2} = 0$$

Then the PDE is classified depending on the value of $(b^2 - 4ac)$:

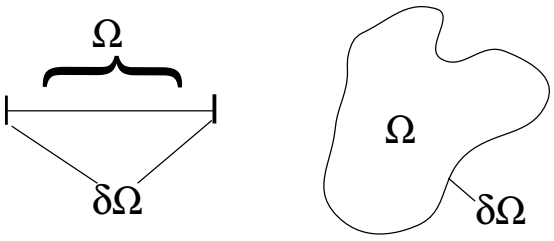
$$(b^2 - 4ac) > 0 \quad \Rightarrow \quad \textbf{Hyberbolic}$$

$$(b^2 - 4ac) = 0 \quad \Rightarrow \quad \textbf{Parabolic}$$

$$(b^2 - 4ac) < 0 \quad \Rightarrow \quad \textbf{Elliptic}$$

1.3.2 Types of boundary conditions

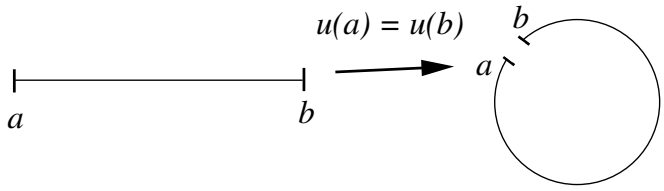
• For a differential equation to be well posed we need to have appropriate boundary conditions and the same is true for the matrix problem.



In a domain Ω with boundaries $\partial\Omega$ possible boundary conditions are:

Type	1D example	2D example
Dirichlet/Essential	$u(\partial\Omega) = c$	$u(\partial\Omega) = f(\partial\Omega)$
Neumann/Free	$\frac{\partial u}{\partial x}(\partial\Omega) = d$	$\frac{\partial u}{\partial n}(\partial\Omega) = g(\partial\Omega)$
Robin/Mixed	$u(\partial\Omega) + \frac{\partial u}{\partial x}(\partial\Omega) = e$	$u(\partial\Omega) + \frac{\partial u}{\partial n}(\partial\Omega) = h(\partial\Omega)$

- Different conditions can (and sometimes must) be attached to different parts of the boundary $\partial\Omega$ depending on the mathematical properties of the equation. (i.e. If the equation is hyperbolic we must only specify conditions on an inflow boundary.)
- Another useful type of boundary condition is a *Periodic* condition.



  - For a one-dimensional region where $a < x < b$ then a periodic boundary condition implies that $u(a) = u(b)$.
  - A stage of a compressor which is not close to inlet or outlet might also be considered as having periodic boundary conditions.

2 General Formulation

We have seen that from the point of view of the fluid dynamics applications that we normally have a governing differential equation. However in structural mechanics our starting point could also begin from the point of view of an energy or force balance at equilibrium conditions.

These different starting point can lead to quite different approaches to our discretisation which leads us to the method of weighted residuals and the principle of virtual displacement. Although these approaches may appear to be very different there are strong similarities, as they both form a variational problem. In this section we will discuss how to derive the weak formulation from the classical form as well as the method of weighted residuals.

2.1 Weak formulation

We consider the Poisson equation written in the classical (*strong*) form

$$-\nabla^2 u = f \quad \text{in } \Omega \tag{7}$$

were Ω is a subset of the one- two- or three-dimensional space in which the problem is considered. The Poisson equations is the simplest and the most famous elliptic partial differential equation and will be used throughout to demonstrate. The source (or load) function f is also given on the Ω domain. A solution of u satisfying equation (7) will also need to satisfy some boundary conditions on the boundary $\partial\Omega$ of Ω

$$\alpha u + \beta \frac{\partial u}{\partial n} = g \quad \text{on } \partial\Omega \tag{8}$$

where $\partial u/\partial n$ denotes the directional derivative in the direction normal to the boundary $\partial\Omega$, (conventionally pointing outwards) and α, β are constants, although variable coefficients are also possible. Equations (7) and (8), together form a *boundary value problem* (BVP). A sufficiently smooth (the highest derivatives and force terms are continuous everywhere in the region of Ω and the boundaries $\partial\Omega$) solution satisfying both (7) and (8) is known as a classical solution to the boundary value problem. In case of non-smooth domains or discontinuous source functions, the function u satisfying the BVP may not be smooth (or regular) enough to be regarded as a classical solution. For non-regular problems (non-regular solution, discontinuous sources), an alternative description of the

BVP is required. This alternative description is less restrictive in terms of the admissible data, it is called a *weak formulation*. To derive a weak formulation of a Poisson problem, we require that for an appropriate set of test functions w,

$$\int_\Omega (\nabla^2 u + f) w = 0 \tag{9}$$

This formulation exists provided that the integrals are well-defined. If u is a classical solution then it must also satisfy (7). On the other hand, if w is sufficiently smooth then, the smoothness required of u can be reduced by using the derivative of a product rule and the divergence theorem,

$$-\int_\Omega w\nabla^2 u = \int_\Omega \nabla u \cdot \nabla w - \int_\Omega \nabla \cdot (w\nabla u) \tag{10}$$

$$= \int_\Omega \nabla u \cdot \nabla w - \int_{\partial\Omega} w\frac{\partial u}{\partial n} \tag{11}$$

so that

$$\int_\Omega \nabla u \cdot \nabla w = \int_\Omega wf + \int_{\partial\Omega} w\frac{\partial u}{\partial n} \tag{12}$$

- The key point here is that the problem may have a not smooth enough solution to be a classical solution which is called *weak solution*
- If a classical solution does exist then **the weak form** (12) **is equivalent to the strong form** ((7) **and** (8)) **and the weak solution is classical.**

At this point is worth defining mathematically what we mean by the term "an appropriate set of test function w" and what guarantees the term "enough smoothness" for the weak solution u. To answer these questions we should define a space of functions in which we should search for the weak solution. To define a suitable solution we need to use a space of functions that are square integrable in the sense of Lebesgue,

$$L_2(\Omega) := \left\{ u : \Omega \to \mathbb{R} \quad \middle| \int_\Omega u^2 < \infty \right\} \tag{13}$$

and make use of the L_2 measure,

$$||u|| := \left(\int_\Omega u^2 \right)^{\frac{1}{2}} \tag{14}$$

then the left-hand side of equation (12) will be well-defined if all the first derivatives are in $L_2(\Omega)$. If Ω is for example a two-dimensional domain and $\partial u/\partial x, \partial u/\partial y \in L_2(\Omega)$ with $\partial w/\partial x, \partial w/\partial y \in L_2(\Omega)$ then by using the Cauchy-Schwarz inequality,

$$\begin{aligned} \int_\Omega \nabla u \cdot \nabla w &= \int_\Omega \left(\frac{\partial u}{\partial x}\right)\left(\frac{\partial w}{\partial x}\right) + \int_\Omega \left(\frac{\partial u}{\partial y}\right)\left(\frac{\partial w}{\partial y}\right) \\ &\leq \left|\left|\frac{\partial u}{\partial x}\right|\right| \left|\left|\frac{\partial w}{\partial x}\right|\right| + \left|\left|\frac{\partial u}{\partial y}\right|\right| \left|\left|\frac{\partial w}{\partial y}\right|\right| < \infty \end{aligned}$$

Similarly for the right-hand side of (12) $f \in L_2(\Omega)$ and $\partial u/\partial n \in L_2(\partial\Omega)$. In summary, we may define the Sobolev space $\mathcal{H}^1(\Omega)$ as

$$\mathcal{H}^1(\Omega) := \left\{ \Omega \to \mathbb{R}^N \middle| u, \frac{\partial u}{\partial x_1}, \frac{\partial u}{\partial x_2}, \ldots, \frac{\partial u}{\partial x_N} \in L_2(\Omega) \right\} \tag{15}$$

as the space of where the weak solution of (12) naturally exist, and this is also the natural home for the test functions w.

Strong form:

Find u such that

$$\begin{aligned} -\nabla^2 u &= f \quad \text{in } \Omega, \\ u = g_{\mathcal{D}} \quad \text{on} \quad \partial\Omega_{\mathcal{D}} \quad \text{and} \quad \frac{\partial u}{\partial n} &= g_{\mathcal{N}} \quad \text{on} \quad \partial\Omega_{\mathcal{N}}, \\ \text{where} \quad \partial\Omega_{\mathcal{D}} \bigcup \partial\Omega_{\mathcal{N}} &= \partial\Omega \quad \text{and} \quad \partial\Omega_{\mathcal{D}} \text{ and } \partial\Omega_{\mathcal{N}} \text{ are distinct.} \end{aligned}$$

Weak form:

Find $u \in \mathcal{H}^1_E$ such that

$$\int_\Omega \nabla u \cdot \nabla w = \int_\Omega w f + \int_{\partial\Omega} w g_{\mathcal{N}} \quad \text{for all } w \in \mathcal{H}^1_0. \tag{16}$$

A key point here is that: a classical solution of a Poisson problem has to be twice differentiable in Ω – this is a much more stringent requirement than square integrability of first derivatives. Using the weak form as the starting point enables to look for approximate solutions that only need to satisfy the smoothness requirement and the essential boundary conditions embodied in the solutions spaces.

2.2 The method of weighted residuals

In approximating numerically an exact solution we are typically replacing the exact solution which in general can only be represented by an *infinite* expansion with a *finite* representation. Such an approximation necessarily means that the differential equation cannot be satisfied everywhere in our region of interest and so we are only able to satisfy a finite number of *conditions*. It is the choice of the *conditions* which are to be satisfied that defines the type of numerical method. For example, the collocation method is an approach where the differential equation is satisfied at a few distinct positions rather than at every point in the solution region. This is the approach normally adopted in the finite difference method.

The method of weighted residuals illustrates how the choice of different weight (or test) functions in an integral or *weak form* of the equation can be used to construct many of the common numerical methods. Let us consider a general linear problem:

$$L(u) = q \tag{17}$$

where $L(u)$ is a linear (usually differential) operator for examples if $L(u) = \frac{\partial^2 u}{\partial x^2}$ and $q = \boldsymbol{f}(x)$ we have our 1-D Poisson equation:

$$u_{xx} = f(x).$$

If we use an approximate numerical solution, denoted by $u^\delta(x)$ then the L.H.S. may not exactly equal the R.H.S. We therefore introduce the *residual* $R(u)$ such that

$$R(u^\delta) = L(u^\delta) - q. \tag{18}$$

- When we have the exact answer $u^\delta(x) = u(x)$ which satisfies equation (17) then $R(u) = 0$. This is the only way of ensuring $R(u)$ is zero everywhere.
- For a given numerical approximation we don't know the exact form of $R(u^\delta)$ and so we want to eliminate this term.
- In the method of weighted residuals, as the name suggests, we multiply equation (18) by a weight (or test) function, denoted by $w(x)$, and integrated over the solution region, denoted by Ω, to obtain:

$$\int_\Omega w(x) R(u^\delta(x)) dx = \int_\Omega w(x) L(u(x)) dx - \int_\Omega w(x) q(x) dx.$$

- Finally we set the integral of the weighted residual equal to zero (i.e. $\int_\Omega w(x) R(u^\delta(x)) dx = 0$) and we are left with

$$\int_\Omega w(x) L(u^\delta(x)) dx = \int_\Omega w(x) q(x) dx \tag{19}$$

- Equation (19) is known as the *integral form* of equation (17).

Although this may appear rather abstract it leads us to an appropriate starting point for the finite element, finite volume or finite difference methods simply by a different choice of the weight function $w(x)$. Strictly speaking each choice of a weight function $w(x)$ defines a different type of projection method.

- If we represent our solution as $u(x) = \sum_{i=1}^{N} \hat{u}_i N_i(x) = \sum_{i=1}^{N} \hat{u}_i \Phi_i(x)$ where $N_i(x)$ or $\Phi_i(x)$ is the basis or trial function then the Galerkin method (which is used in the finite element approach) has a weight function $w_j(x) = N_j(x)$. The form of $N_i(x)$ for the classical linear finite elements is shown in figure 1
  - If we use a different choice of a continuous function so that $w_j(x) \neq N_j(x)$ then the projection is referred to as the Petrov-Galerkin methods. This arises when we want to introduce upwinding into the finite element method.
- If we choose a step type function (see figure 1) which has a value of 1 in a so-called "cell" and is zero outside then we have a subdomain projection which is used in the finite volume methods.
- If we choose $w_i(x) = \delta(x - x_j)$ where x_j are the mesh points then we have the collocation method which is the starting point of the the finite difference method.

A list of other types of methods with the appropriate choice of weight function is given in table (2.2).

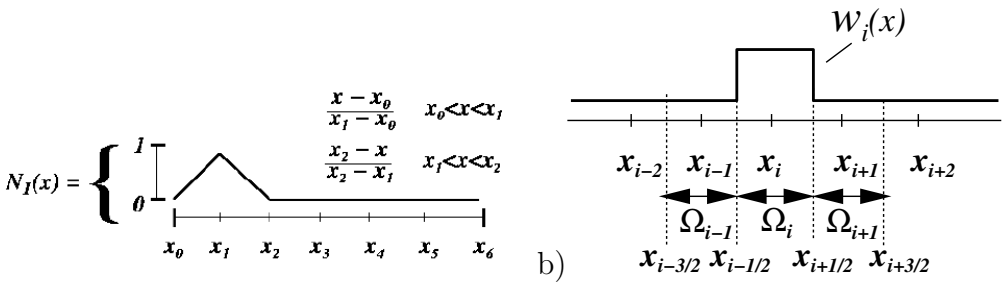


Figure 1: Weight functions for the (a) finite element and (b) finite volume methods

Table 1: Test functions used in the method of weighted residuals.

Test function	Type of method
$w_j(\boldsymbol{x}) = \delta(\boldsymbol{x} - \boldsymbol{x}_j)$	Collocation/Finite Difference
$w_j(\boldsymbol{x}) = \begin{cases} 1 & \text{inside} \quad \Omega^j \\ 0 & \text{outside} \quad \Omega^j \end{cases}$	Finite volume (subdomain)
$w_j(\boldsymbol{x}) = \dfrac{\partial R}{\partial \hat{u}_j}$	Least-squares
$w_j(x) = x^j$	Method of moments
$w_j(\boldsymbol{x}) = N_j$	Galerkin
$w_j(\boldsymbol{x}) = \Psi_i \ (\neq N_j)$	Petrov-Galerkin

3 Galerkin Formulation

As discussed in section (2.2) in the finite element methods we typically use the Galerkin formulation.

In this section, we describe how to formulate the Galerkin problem starting from a partial differential equation and illustrate the properties of this type of formulation. In section 3.1 we present an informal formulation, by considering the one-dimensional Poisson equation to introduce the basic concepts. The formulation is then illustrated by the use of a worked example using linear finite elements in section 3.2. A mathematical statement of the formulation is presented in section 3.3, and finally some important properties of the Galerkin formulation are given in section 3.3.1.

3.1 Descriptive Formulation

We consider the one-dimensional Poisson equation

$$\mathbb{L}(u) \equiv \frac{\partial^2 u}{\partial x^2} + f = 0. \tag{20}$$

3.1.1 Strong Form and Definition of Boundary Conditions

For this problem to be well posed and therefore have a unique solution we need to specify boundary conditions. If we consider the solution in a region $\Omega = \{x \mid 0 < x < 1\}$, then we could consider the following boundary conditions Dirichlet and Neumann conditions:

$$u(0) = g_{\mathcal{D}}, \qquad \frac{\partial u}{\partial x}(1) = g_{\mathcal{N}},$$

where $g_{\mathcal{D}}$ and $g_{\mathcal{N}}$ are given constants for the one-dimensional problem.

As we shall see, in the Galerkin formulation Dirichlet boundary conditions have to be specified explicitly whereas Neumann conditions are dealt with implicitly as part of the formulation. If the boundary conditions stated above are applied to equation (20) we have a two-point boundary value problem and is said to be in the *strong* or *classical* form.

3.1.2 Weak Form and Implementation of Neumann Boundary Conditions

Following the formulation of the method of weighted residuals we multiply equation (20) by a weight or test function $v(x)$ (also denoted by $w(x)$), which by definition is **zero** on all Dirichlet boundaries $\partial\Omega_{\mathcal{D}}$, and integrate over the domain Ω we obtain the inner product of $\mathbb{L}(u)$ with v,

$$(v, \mathbb{L}(u)) = \int_0^1 v \left(\frac{\partial^2 u}{\partial x^2} + f \right) dx = 0. \tag{21}$$

We can see that equation (21) is equivalent to setting the weighted residual to zero. If u^δ is an approximation to u (recalling that $\mathbb{L}(u^\delta) = R(u^\delta)$) then equation (21) is equivalent to the condition $(v, R) = 0$.

Integrating equation (21) by parts we obtain

$$\int_0^1 \frac{\partial v}{\partial x} \frac{\partial u}{\partial x} dx = \int_0^1 vf \; dx + \left[v \frac{\partial u}{\partial x} \right]_0^1. \tag{22}$$

This is a common approach in finite elements and reduces the order of the second derivative. Note that in a finite difference methods we would normally have applied an approximation to the second derivative in the strong form. As the test functions are defined to be zero on Dirichlet boundaries we know that $v(0) = 0$. Therefore, if we apply the Neumann boundary condition $\partial u(1)/\partial x = g_{\mathcal{N}}$ to the last term, equation (22) can be simplified to obtain

$$\int_0^1 \frac{\partial v}{\partial x} \frac{\partial u}{\partial x} dx = \int_0^1 vf \; dx + v(1) g_{\mathcal{N}}. \tag{23}$$

In this last step, we see how the Neumann boundary conditions are naturally included in the formulation. We recall that the integral form of the equation, as shown in equations (22) and (23), is referred to as the *weak* form of the problem.

The Galerkin finite element approximation of problem (20) is the solution to the weak form of the equation (23) when the exact solution $u(x)$ is approximated by a finite expansion denoted by $u^\delta(x)$. The weight or test function $v(x)$ in (23) is also replaced by a finite expansion, denoted by $v^\delta(x)$, and so equation (23) becomes

$$\int_0^1 \frac{\partial v^\delta}{\partial x} \frac{\partial u^\delta}{\partial x} dx = \int_0^1 v^\delta f \; dx + v^\delta(1) g_{\mathcal{N}}. \tag{24}$$

The set of functions used in the finite expansion of the solution u^δ are referred to as the *trial* functions whereas the functions contained within v^δ are referred to as the *test* functions.

3.1.3 Implementation of Dirichlet Boundary Conditions

In the Galerkin approximation the same set of functions that are used to represent v^δ are also used in the representation of the solution u^δ. As all the test functions within v^δ are defined as zero on Dirichlet boundaries it is clear that the trial solution u^δ must also contain some other functions which are non-zero at this boundary. Without this it would not be possible to satisfy the Dirichlet boundary condition of the problem. The approximate solution u^δ is therefore constructed from a known function $u^\mathcal{D}$, which satisfies the Dirichlet boundary conditions, and an unknown homogeneous function, $u^\mathcal{H}$, which is zero on the Dirichlet boundaries, that is,

$$u^\delta = u^\mathcal{H} + u^\mathcal{D}, \tag{25}$$

where

$$u^\mathcal{H}(\partial\Omega_\mathcal{D}) = 0, \qquad u^\mathcal{D}(\partial\Omega_\mathcal{D}) = g_\mathcal{D}.$$

The same set of functions are now used to represent the homogeneous solution $u^\mathcal{H}$ and the test function v^δ. Substituting equation (25) into equation (24) gives

$$\int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u^\mathcal{H}}{\partial x}\,dx = \int_0^1 v^\delta f\,dx + v^\delta(1)g_\mathcal{N} - \int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u^\mathcal{D}}{\partial x}\,dx. \tag{26}$$

As will be illustrated in section 3.2, equation (26) can be solved as a finite algebraic system as all the terms on the right-hand side are known and the homogeneous solution $u^\mathcal{H}$ and test function v^δ contain a finite number of functions. The Galerkin formulation has therefore reduced differential problem (20) to an algebraic system which we can solve on a computer.

3.2 Two-Domain Linear Finite Element Example

In this section we solve the one-dimensional Poisson equation

$$\mathbb{L}(u) \equiv \frac{\partial^2 u}{\partial x^2} + f = 0,$$

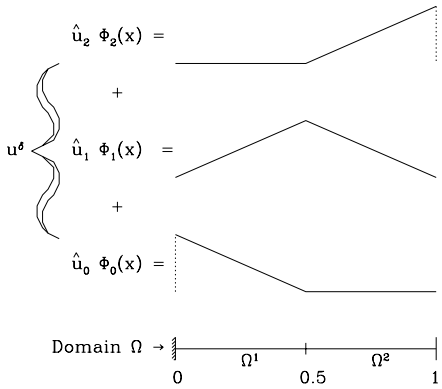


Figure 2: Linear finite element approximation $u^\delta(x) = \sum_{i=0}^{2} \hat{u}_i \Phi_i(x)$, in a domain Ω, using two elemental subdomains, Ω^1 and Ω^2.

where $f(x)$ is a known function and the boundary conditions are

$$u(0) = g_{\mathcal{D}} = 1, \qquad \frac{\partial u}{\partial x}(1) = g_{\mathcal{N}} = 1.$$

We start by considering the weak form [equation (26)], that is,

$$\int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u}{\partial x}^{\mathcal{H}} dx = \int_0^1 v^\delta f \, dx + v^\delta(1) g_{\mathcal{N}} - \int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u}{\partial x}^{\mathcal{D}} dx.$$

The solution is approximated by piecewise linear functions over two sub-domains Ω^1, Ω^2 as shown in figure 2. This type of approximation is known as an h-type approximation, where the h parameter represents the characteristic size of a sub-domain (in one dimension, its length). Convergence to the exact solution is achieved by subdividing the solution domain Ω into smaller and smaller sub-domains so that $h \to 0$. For this two sub-domain case the approximate solution is given by an expansion of the form:

$$u^\delta = \sum_{i=0}^{2} \hat{u}_i \Phi_i(x),$$

where $\Phi_i(x)$ is defined as

$$\Phi_0(x) = \begin{cases} 1-2x & 0 \le x \le \frac{1}{2} \\ 0 & \frac{1}{2} \le x \le 1 \end{cases}, \quad \Phi_1(x) = \begin{cases} 2x & 0 \le x \le \frac{1}{2} \\ 2(1-x) & \frac{1}{2} \le x \le 1 \end{cases},$$

$$\Phi_2(x) = \begin{cases} 0 & 0 \le x \le \frac{1}{2} \\ 2x-1 & \frac{1}{2} \le x \le 1 \end{cases}.$$

At this stage we have simplified the problem by considering Φ as a global expansion. However the great power of the finite element method is its geometric flexibility arises from decomposing the global expansions into local expansions as we shall see in section 4.

The only way to satisfy the Dirichlet boundary condition at $x = 0$ is to set $\hat{u}_0 = g_{\mathcal{D}}$; therefore we decompose u^δ into $u^\delta = u^{\mathcal{H}} + u^{\mathcal{D}}$, that is,

$$\begin{aligned} u^{\mathcal{H}} &= \hat{u}_1\Phi_1(x) + \hat{u}_2\Phi_2(x) \\ u^{\mathcal{D}} &= g_{\mathcal{D}}\Phi_0(x), \end{aligned}$$

where $\hat{u}_1$ and $\hat{u}_2$ are still to be determined. In the classical Galerkin approach the expansion bases used to define $u^{\mathcal{H}}$ are also used to define the test functions, and so we can define the test functions as

$$v^\delta(x) = \hat{v}_1\Phi_1(x) + \hat{v}_2\Phi_2(x).$$

where $\hat{v}_1$ and $\hat{v}_2$ are also unknown but as we shall see shortly never have to be determined. Finally we need a representation of the function $f(x)$. This function is known explicitly and therefore it is theoretically possible to evaluate exactly any operations involving $f(x)$ with other known functions such as $\Phi_i(x)$. However, in practice, in order to treat an arbitrary function in a computational implementation, the function is usually represented using the same expansion as u^δ that is,

$$f^\delta(x) = \sum_{i=0}^{2} \hat{f}_i\Phi_i(x) = \hat{f}_0\Phi_0(x) + \hat{f}_1\Phi_1(x) + \hat{f}_2\Phi_2(x).$$

Clearly, if $f(x)$ is a constant or a linear function then it will be exactly represented by $f^\delta(x)$. For more complex functions the coefficients $\hat{f}_0$, $\hat{f}_1$, and $\hat{f}_2$ need to be determined and could be chosen to satisfy an interpolation approximation

where the approximation $f^\delta(x_i) = f(x_i)$ where x_i are the mesh points. This would mean that for our problem $\hat{f}_0 = f(0)$, $\hat{f}_1 = f(0.5)$, and $\hat{f}_2 = f(1)$.

Evaluating the terms in equation (26) we find:

$$\begin{aligned}
\int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u}{\partial x}^{\mathcal{H}} dx &= \int_0^{\frac{1}{2}} (2\hat{v}_1)(2\hat{u}_1)dx + \int_{\frac{1}{2}}^1 (-2\hat{v}_1 + 2\hat{v}_2)(-2\hat{u}_1 + 2\hat{u}_2)dx \\
&= \begin{bmatrix} \hat{v}_1 & \hat{v}_2 \end{bmatrix} \begin{bmatrix} 4 & -2 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} \qquad (27a) \\
\int_0^1 v^\delta f\, dx &= \int_0^{\frac{1}{2}} (\hat{v}_1 2x)(\hat{f}_0(1-2x) + \hat{f}_1(2x))dx \\
&+ \int_{\frac{1}{2}}^1 (\hat{v}_1 2(1-x) + \hat{v}_2(2x-1))(\hat{f}_1 2(1-x) + \hat{f}_2(2x-1))dx \\
&= \begin{bmatrix} \hat{v}_1 & \hat{v}_2 \end{bmatrix} \begin{bmatrix} \frac{1}{12}\hat{f}_0 + \frac{1}{3}\hat{f}_1 + \frac{1}{12}\hat{f}_2 \\ \frac{1}{12}\hat{f}_1 + \frac{1}{6}\hat{f}_2 \end{bmatrix} \qquad (27b) \\
v^\delta(1) g_{\mathcal{N}} &= (\hat{v}_1\Phi_1(1) + \hat{v}_2\Phi_2(1))g_{\mathcal{N}} = \begin{bmatrix} \hat{v}_1 & \hat{v}_2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} g_{\mathcal{N}} \qquad (27c) \\
\int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u}{\partial x}^{\mathcal{D}} dx &= \int_0^{\frac{1}{2}} (2\hat{v}_1)(-2g_{\mathcal{D}})dx = \begin{bmatrix} \hat{v}_1 & \hat{v}_2 \end{bmatrix} \begin{bmatrix} -2g_{\mathcal{D}} \\ 0 \end{bmatrix}. \qquad (27d)
\end{aligned}$$

Therefore, equation (26) becomes

$$\begin{bmatrix} \hat{v}_1 & \hat{v}_2 \end{bmatrix} \left\{ \begin{bmatrix} 4 & -2 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} - \begin{bmatrix} \frac{1}{12}\hat{f}_0 + \frac{1}{3}\hat{f}_1 + \frac{1}{12}\hat{f}_2 \\ \\ \frac{1}{12}\hat{f}_1 + \frac{1}{6}\hat{f}_2 \end{bmatrix} - \begin{bmatrix} 0 \\ g_{\mathcal{N}} \end{bmatrix} + \begin{bmatrix} -2g_{\mathcal{D}} \\ 0 \end{bmatrix} \right\} = 0.$$

For arbitrary choices of $\hat{v}_1$ and $\hat{v}_2$ we can solve this equation by evaluating the matrix equation in the curly brackets. Recalling that $g_{\mathcal{D}} = 1$ and $g_{\mathcal{N}} = 1$ the matrix equation becomes

$$\begin{bmatrix} 4 & -2 \\ -2 & 2 \end{bmatrix} \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} = \begin{bmatrix} 2 + \frac{1}{12}\hat{f}_0 + \frac{1}{3}\hat{f}_1 + \frac{1}{12}\hat{f}_2 \\ \\ 1 + \frac{1}{12}\hat{f}_1 + \frac{1}{6}\hat{f}_2 \end{bmatrix},$$

which has a solution

$$\begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} = \begin{bmatrix} \frac{3}{2} + \frac{1}{24}\hat{f}_0 + \frac{5}{24}\hat{f}_1 + \frac{1}{8}\hat{f}_2 \\ \\ 2 + \frac{1}{24}\hat{f}_0 + \frac{1}{4}\hat{f}_1 + \frac{5}{24}\hat{f}_2 \end{bmatrix}.$$

The finite element approximation $u^\delta(x) = g_\mathcal{D}\Phi_0(x) + \hat{u}_1\Phi_1(x) + \hat{u}_2\Phi_2(x)$ is therefore:

$$u^\delta = \begin{cases} 1 + x + \dfrac{x}{12}\hat{f}_0 + \dfrac{5x}{12}\hat{f}_1 + \dfrac{x}{4}\hat{f}_2 & 0 \le x \le \frac{1}{2} \\ 1 + x + \dfrac{1}{24}\hat{f}_0 + \dfrac{2+x}{12}\hat{f}_1 + \dfrac{1+4x}{24}\hat{f}_2 & \frac{1}{2} \le x \le 1 \end{cases}.$$

3.3 Appendix: Mathematical Formulation

In general the previous formulation can be considered in a more mathematical setting. Although this is not necessary for us to continue our practical discussion of the implementation of a finite element method it may be of general interest to understand terminology of such an approach. We consider the one-dimensional Helmholtz equation

$$\mathbb{L}(u) = \frac{\partial^2 u}{\partial x^2} - \lambda u + f = 0, \tag{28}$$

where λ is a real positive constant. The equation is presumed to be supplemented with appropriate boundary conditions such as

$$u(0) = g_\mathcal{D}, \qquad \frac{\partial u}{\partial x}(l) = g_\mathcal{N}.$$

As indicated by the boundary conditions, we wish to determine the solution in the interval $0 < x < l$ which we shall denote by Ω.
Multiplying equation (28) by an arbitrary test function $v(x)$, the properties of which are to be defined, and integrating over the domain Ω we obtain:

$$\int_0^l v\frac{\partial^2 u}{\partial x^2} - \int_0^l \lambda v u \; dx + \int_0^l v f \; dx = 0.$$

Providing $u(x)$ and $v(x)$ are sufficiently smooth, we can integrate the first term by parts to arrive at:

$$\int_0^l \frac{\partial v}{\partial x}\frac{\partial u}{\partial x} + \int_0^l \lambda v u \; dx = \int_0^l v f \; dx + \left[v\frac{\partial u}{\partial x}\right]_0^l. \tag{29}$$

If we introduce the notation

$$\begin{aligned} a(v,u) &= \int_0^l \left(\frac{\partial v}{\partial x}\frac{\partial u}{\partial x} + \lambda v u \right) dx \\ f(v) &= \int_0^l v f dx + \left[v\frac{\partial u}{\partial x} \right]_0^l \end{aligned}$$

then equation (29) can be written as

$$a(v,u) = f(v). \tag{30}$$

In structural mechanics, $a(u,u)$ is referred to as the *strain energy* and the space of all functions which have a finite strain on Ω is called the *energy* space which is denoted by $E(\Omega)$:

$$E(\Omega) = \{u \mid a(u,u) < \infty\}.$$

Associated with the energy space is the energy norm $||u||_E$ defined as:

$$||u||_E = \sqrt{a(u,u)}. \tag{31}$$

Functions that belong to the energy space are called H^1 functions and satisfy the condition that the integral of the square of the function and its derivative are bounded.

We consider solutions to problem (28) where the forcing function f is *well behaved* in the sense that $f(v)$ is finite. Therefore, we only consider candidate or *trial* solutions to problem (29) which lie in the energy space and satisfy the Dirichlet boundary condition. This space is called the *trial space* and is denoted by $\mathcal{X}$. For our problem the trial space is defined by

$$\mathcal{X} = \{u \mid u \in H^1, u(0) = g_{\mathcal{D}}\}.$$

Similarly, we define the space of all test functions, denoted by $\mathcal{V}$, which are homogeneous on all Dirichlet boundaries, that is,

$$\mathcal{V} = \{v \mid v \in H^1, v(0) = 0\}.$$

The test space $\mathcal{V}$ is sometimes said to be in H_0^1 where the subscript 0 refers to the fact that it is in the homogeneous space. We can now define the generalized or weak formulation of equation (28) as

Find $u \in \mathcal{X}$, such that

$$a(v,u) = f(v), \qquad \forall\ v \in \mathcal{V}. \qquad (32)$$

The weak problem is still an infinite dimensional problem because the trial and test spaces, $\mathcal{X}$ and $\mathcal{V}$, contain an infinite number of functions. Therefore, weselect subspaces $\mathcal{X}^\delta$ ($\mathcal{X}^\delta \subset \mathcal{X}$) and $\mathcal{V}^\delta$ ($\mathcal{V}^\delta \subset \mathcal{V}$) which contain a finite number of functions. The approximate form of the weak solution can then be stated as

Find $u^\delta \in \mathcal{X}^\delta$, such that

$$a(v^\delta, u^\delta) = f(v^\delta) \qquad \forall\ v^\delta \in \mathcal{V}^\delta. \qquad (33)$$

In the Galerkin approximation the same set of functions is used for both the test and trial functions. For this to be possible we must make the solution homogeneous. We observe that the function $u^\delta \in \mathcal{X}^\delta$ can be decomposed into an known component, $u^\mathcal{D}$, which lies in the trial space ($u^\mathcal{D} \in \mathcal{X}^\delta$) and satisfies the Dirichlet boundary condition and an unknown component, $u^\mathcal{H}$, which lies in the test space ($u^\mathcal{H} \in \mathcal{V}^\delta$) and is zero on the Dirichlet boundary. In other words,

$$u^\delta = u^\mathcal{H} + u^\mathcal{D},$$

where

$$u^\mathcal{H}(0) = 0, \qquad u^\mathcal{D}(0) = g_\mathcal{D}.$$

It can be appreciated that up to the function $u^\mathcal{D}$ the test and trial spaces contain identical functions.

We are now in a position to define the Galerkin approximation but before we do so we mention that $a(v,u)$ is a *symmetric, bilinear form* which means

$$a(v,u) = a(u,v) \qquad (34a)$$

$$a(c_1 v + c_2 w, u) = c_1 a(v,u) + c_2 a(w,u), \qquad (34b)$$

where c_1 and c_2 are constants and u, v and w are functions. Furthermore, the operator $a(v,u)$ is said to be continuous (or bounded) if

$$|a(v,u)| \leq C_1 ||v||_1 ||u||_1 \qquad (34c)$$

where $C_1 < \infty$ and subscript denotes norm in H^1. It is elliptic (or coercive) if

$$a(u,u) \geq C_2 ||v||_1^2 \qquad (34d)$$

where $C_2 > 0$. The Galerkin form of the problem can now be stated as:

Find

$$u^\delta = u^\mathcal{D} + u^\mathcal{H},$$

where

$$u^\mathcal{H} \in \mathcal{V}^\delta,$$

such that

$$a(v^\delta, u^\mathcal{H}) = f(v^\delta) - a(v^\delta, u^\mathcal{D}) \quad \text{for all} \quad v^\delta \in \mathcal{V}^\delta.$$

For this linear equation another way of constructing the Galerkin solution is from a variational point of view. Equation (28) is the minimal solution to the functional

$$\mathcal{F}(v) = \int_0^l \left[\left(\frac{\partial v}{\partial x} \right)^2 + \lambda (v)^2 - 2vf \right] dx.$$

Therefore, if we minimize $\mathcal{F}(v)$ over the infinite dimensional space $\mathcal{V}$ we will find the solution to equation (28) which is the Euler equation of this functional. Replacing the variational problem by a finite dimensional subspace $\mathcal{V}^\delta$ leads to the Ritz-Galerkin method.

3.3.1 Mathematical properties of the Galerkin approximation

In this section we introduce some important properties of the Galerkin approximation. We consider the approximation u^δ to the solution u where $u^\delta \in \mathcal{X}^\delta$ and satisfies

$$a(v^\delta, u^\delta) = f(v^\delta), \qquad \forall\ v^\delta \in \mathcal{V}^\delta. \tag{35}$$

Note that this is equivalent to equation (3.3) since $a(v^\delta, u^\delta) = a(v^\delta, u^\mathcal{D}) + a(v^\delta, u^\mathcal{H})$ using the bilinearity of $a(v, u)$ [equation (34b)].

Uniqueness

To show that the solution u^δ is unique we assume that there are two distinct solutions u_1 and u_2 ($u_1, u_2 \in \mathcal{X}^\delta$) which satisfy

$$a(v^\delta, u_1) = f(v^\delta), \qquad \text{for all } v^\delta \in \mathcal{V}^\delta \tag{36a}$$

and

$$a(v^\delta, u_2) = f(v^\delta), \qquad \text{for all } v^\delta \in \mathcal{V}^\delta. \tag{36b}$$

Subtracting equation (36a) from (36b) we obtain:

$$a(v^\delta, u_1) - a(v^\delta, u_2) = a(v^\delta, u_1 - u_2) = 0 \tag{36c}$$

using the bilinearity of $a(v, u)$. Now $u_1 - u_2 \in \mathcal{V}^\delta$ and therefore we can set $v^\delta = u_1 - u_2$ so equation (36c) becomes

$$a(u_1 - u_2, u_1 - u_2) = 0.$$

However, this implies that $||u_1 - u_2||_E = 0$ which is only possible if $u_1 = u_2$, but this contradicts the assumption that they are distinct. We therefore conclude that there is only one unique solution. Strictly speaking, $||u_1 - u_2||_E = 0$ only implies that $u_1 = u_2$ if $\lambda \neq 0$. When $\lambda = 0$ the solution is only unique up to an arbitrary constant, that is, $u_1 - u_2 = C$. The constant, C, is necessarily zero if Dirichlet boundary conditions are specified, although the norm $||u_1 - u_2||_E$ cannot distinguish between functions that differ by an arbitrary constant when $\lambda = 0$.

Orthogonality of the Error to the Test Space

The error between the exact and approximate solution, $\varepsilon = u - u^\delta$, is orthogonal to all functions in the finite dimensional test space $\mathcal{V}^\delta$ in the energy norm, that is,

$$a(v^\delta, \varepsilon) = 0, \qquad \forall\ v^\delta \in \mathcal{V}^\delta. \tag{37a}$$

To prove this property we recall that the exact solution satisfies the weak equation (32); in other words,

$$a(v, u) = f(v), \qquad \forall\ v \in \mathcal{V},$$

and the approximation satisfies equation (35). The finite dimensional test space $\mathcal{V}^\delta$ is a subspace of $\mathcal{V}$ and so the exact solution also satisfies

$$a(v^\delta, u) = f(v^\delta), \qquad \forall\ v^\delta \in \mathcal{V}^\delta. \tag{37b}$$

Subtracting equation (35) from equation (37b) with $\varepsilon = u - u^\delta$ and using the bilinearity of $a(v, u)$ gives equation (37a).

Minimal Property of Error in the Energy Norm

We can show that the finite element solution u^δ is the solution in $\mathcal{X}^\delta$ which minimizes the energy norm of the error, that is,

$$||u - u^\delta||_E = \min_{w^\delta \in \mathcal{X}^\delta} ||u - w^\delta||_E. \tag{38a}$$

To demonstrate this result we let $\varepsilon = u - u^\delta$ and observe that for any $w^\delta \in \mathcal{X}^\delta$ we can write

$$||u - w^\delta||_E^2 = ||u - u^\delta + u^\delta - w^\delta||_E^2 = ||\varepsilon + v^\delta||_E^2$$

where $v^\delta = w^\delta - u^\delta \in \mathcal{V}^\delta$. From the definition of the energy norm (31) and using the bilinearity of $a(v, u)$ [equation (34b)] we obtain

$$||u - w^\delta||_E^2 = a(\varepsilon + v^\delta, \varepsilon + v^\delta) = a(\varepsilon, \varepsilon) + 2a(v^\delta, \varepsilon) + a(v^\delta, v^\delta). \tag{38b}$$

Now, since $v^\delta \in \mathcal{V}^\delta$, we know from equation (37a) that $a(v^\delta, \varepsilon) = 0$. Therefore, if there were any choices of w^δ which gave a smaller error than $u - u^\delta$, in the energy norm, it would have to make the last term of (38b) negative. However, if $v \neq 0$ then $a(v, v) > 0$ and so the minimizing choice of w^δ is one that sets $v^\delta = 0$, thus implying that $w^\delta = v^\delta$ and proving (38a).

Equivalence of Polynomial Bases in the Energy Norm

An almost trivial observation from the uniqueness of the Galerkin approximation is that any two linearly independent expansions which span the same trial space $\mathcal{X}^\delta$ necessarily have the same approximate solution $u^\delta(x)$. So if we consider two solutions $u_1^\delta(x) = \sum_i^P \alpha_i \psi_i(x)$ and $u_2^\delta(x) = \sum_i^P \beta_i h_i(x)$, where both expansion functions are in a polynomial space of order P (i.e., $\psi_i(x), h_i(x) \in \mathcal{P}_P$) then if the solutions $u_1^\delta(x)$ and $u_2^\delta(x)$ are both determined as solutions to the Galerkin approximation (35) we know that

$$u_1^\delta(x) = u_2^\delta(x) \qquad \Rightarrow \qquad \sum_{i=0}^{P} \alpha_i \psi_i(x) = \sum_{i=0}^{P} \beta_i h_i(x).$$

The important implication of this statement is that any error estimates are independent of the type of the polynomial expansion and only depend on the polynomial space. Nevertheless, different choices of polynomial expansion bases can have an important effect on the numerical conditioning of the algebraic systems resulting from Galerkin approximation.

4 One-Dimensional Element Expansion

Having defined the finite element framework in terms of the Galerkin formulation, we can now consider how to implement a multi-element decomposition which will demonstrate the true flexibility of the finite element approach.

In the h-type method a fixed order polynomial is used in every element and convergence is achieved by reducing the size of the elements. This is the so-called h-type extension where h represents the characteristic size of an element and was illustrated for two elements in section 3.2.

In the p-type method a fixed mesh is used and convergence is achieved by increasing the order of the polynomial in every element. This is the so-called p-type extension where p represents the polynomial order in the elements.

If the whole solution domain is treated as a single element then the p-type method becomes a spectral method. The hp finite element method combines attributes from both of these methods.

4.1 Elemental Decomposition: The h-Type Extension

As will be shown, the principal use of elemental representation is to enable the treatment of operations on a local elemental basis. This not only simplifies the implementation but also allows many operations to be performed more efficiently. For the one-dimensional case, the decomposition can seem unnecessarily involved, however, the same principles are applied to the decomposition in multiple dimensions. Therefore, the one-dimensional case is explained in detail as a building block for decomposition in multiple dimensions.

4.1.1 Partitioning of Solution Domain

When using an h-type method the solution domain is subdivided or partitioned into *non-overlapping* sub-domains or elements within which a polynomial expansion is applied. In the standard case this will simply be a linear polynomial.

Considering a solution-domain Ω, we can partition it into a mesh containing N_{el} elements, denoted by Ω^e, such that the union of the non-overlapping elements equals the original domain, that is,

$$\Omega = \bigcup_{e=1}^{N_{el}} \Omega^e \quad \text{where} \quad \bigcap_{e=1}^{N_{el}} \Omega^e = \emptyset.$$

For the domain $\Omega = \{x \mid 0 < x < l\}$ a specific mesh can be denoted by the points

$$0 = x_0 < x_1 < \cdots < x_{N_{el}-1} < x_{N_{el}} = l.$$

Therefore, the eth element is defined as

$$\Omega_e = \{x \mid x_{e-1} < x < x_e\}.$$

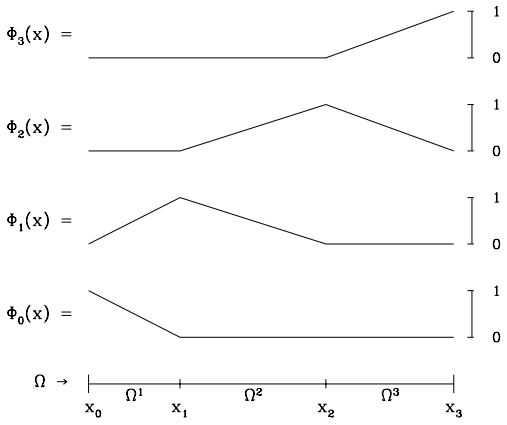


Figure 3: Elemental decomposition of the solution domain Ω into three elements $\Omega^1, \Omega^2, \Omega^3$. Also shown are the global expansion modes $\Phi_0(x), \Phi_1(x), \Phi_2(x), \Phi_3(x)$ for a linear finite element expansion over the domain Ω.

As an example we can consider the case shown in figure 3 where we consider the solution domain, $\Omega = \{x \mid 0 < x < l\}$, subdivided into $N_{el} = 3$ non-equal elements. The mesh is denoted by the $(N_{el}+1)$ points x_0, x_1, x_2, x_3 and therefore the first element is defined as

$$\Omega_1 = \{x | x_0 < x < x_1\}.$$

4.1.2 The Standard Element and the Linear Finite Element Expansion

In figure 3, the global expansion modes for the linear finite element expansion over the $N_{el} = 3$ elemental domains are also shown. As is typical in a linear

finite element expansion, each mode has a unit value at the end of one of the elemental domains and decays linearly to zero across the neighboring elements. Therefore, there are $N_{dof} = 4$ degrees of freedom in this expansion, that is, $\Phi_0(x), \Phi_1(x), \Phi_2(x)$, and $\Phi_3(x)$. The global modes are only non-zero on, at most, two elements. It would therefore be very uneconomical to consider an expansion in terms of global modes when using a large number of elements. Global modes are however attractive when there is only one element and a high order expansion is used as in a spectral or Fourier method.

We can see that at an elemental level each global mode only consists of two linearly varying functions. Therefore, if we introduce a standard element, Ω_{st} such that

$$\Omega_{st} = \{\xi \mid -1 < \xi < 1\},$$

then we can define a similar linear varying function over Ω_{st} in terms of the local coordinate ξ as

$$\phi_0(\xi) = \begin{cases} \dfrac{1-\xi}{2} & \xi \in \Omega_{st} \\ 0 & \text{otherwise} \end{cases} \qquad \phi_1(\xi) = \begin{cases} \dfrac{1+\xi}{2} & \xi \in \Omega_{st} \\ 0 & \text{otherwise} \end{cases}.$$

The standard element Ω_{st} can be mapped to any elemental domain Ω^e via the transformation, denoted as $\chi^e(\xi)$, which expresses the global coordinate x in terms of the local coordinate ξ as

$$x = \chi^e(\xi) = \frac{(1-\xi)}{2} x_{e-1} + \frac{(1+\xi)}{2} x_e, \qquad \xi \in \Omega_{st}. \tag{39}$$

The mapping (39) has an analytic inverse, $(\chi^e)^{-1}(x)$, of the form

$$\xi = [\chi^e]^{-1}(x) = 2\frac{(x - x_{e-1})}{(x_e - x_{e-1})} - 1, \qquad x \in \Omega^e.$$

The global modes $\Phi_i(x)$ can now be represented in terms of the local elemental expansion modes $\phi_p(\xi)$ by mapping the standard element Ω_{st} to each elemental domain Ω^e. For example, the first two global expansion modes $\Phi_0(x), \Phi_1(x)$ in

figure 3 can be written

$$\Phi_0(x) = \begin{cases} \dfrac{(x-x_1)}{(x_0-x_1)} & x \in \Omega^1 \\ 0 & \text{otherwise} \end{cases} = \begin{cases} \phi_0([\chi^1]^{-1}(x)) & x \in \Omega^1 \\ 0 & \text{otherwise} \end{cases}$$

$$\Phi_1(x) = \begin{cases} \dfrac{(x-x_0)}{(x_1-x_0)} & x \in \Omega^1 \\ \dfrac{(x-x_2)}{(x_1-x_2)} & x \in \Omega^2 \\ 0 & \text{otherwise} \end{cases} = \begin{cases} \phi_1([\chi^1]^{-1}(x)) & x \in \Omega^1 \\ \phi_0([\chi^2]^{-1}(x)) & x \in \Omega^2 \\ 0 & \text{otherwise} \end{cases} .$$

If a mapping for $\chi^e(\xi)$ other than the one shown in (39) has been used then the inverse mapping will not necessarily be analytic. This situation can arise in multiple dimensions where elements may be distorted or curved. In general we do not need to know the form of the inverse mapping.

4.1.3 Parametric Mapping

The mapping $\chi^e(\xi)$ from the *local* coordinate ξ to the *global* coordinate x ($x \in \Omega_e$) given in equation (39) can be considered as expanding the global coordinate as a linear finite element expansion and so it could have been written as

$$x = \chi^e(\xi) = \phi_0(\xi)\ x_{e-1} + \phi_1(\xi)\ x_e, \qquad \xi \in \Omega_{st}.$$

This technique of expressing the global coordinate, x, in terms of the local expansion function is known as a *parametric mapping*. Typically, we refer to the mapping as being *iso-parametric* if we use the same order expansion to map the coordinates as we use to represent the dependent variables, i.e. $u, v, ...$ Iso-parametric mapping is a convenient way to express curved domains. If we use a higher or lower order mapping for the coordinates as compared to the dependent variables the mapping is referred to as *super-* or *sub*-parametric, respectively. This situation arises more commonly for the p-type finite element extension.

We note that the mapping in equation (39) is linear and therefore so is its inverse. This means that the local expansion mode $\phi_p(\chi_e^{-1}(x))$ is a polynomial

in x as well as is ξ and so the global expansion modes are also polynomials in x. However, when a more complicated mapping is used, as in the curved element case, the global expansion may not remain a polynomial in x although, by definition, it is always a polynomial in ξ.

4.1.4 Global Assembly/Direct Stiffness Summation

An important operation of the elemental decomposition is the concept of *global assembly* or *direct stiffness summation* as it is sometimes known. This operation takes our local elemental description to a global setting and vice versa. Recall from section 3.2 that the finite element approximation u^δ can be written in terms of the global modes as

$$u^\delta(x) = \sum_{i=0}^{N_{dof}-1} \hat{u}_i \Phi_i(x),$$

where $\hat{u}_i$ is the global expansion coefficient and represents the solution at the mesh points in our linear finite element example. We have also seen that the global modes $\Phi_i(x)$ can be expressed in terms of the local expansion modes $\phi_p(\xi)$. So we could introduce a local expansion coefficient within each element $\hat{u}^e_p$ where e denotes the eth element and p represents an index looping over the local polynomial order of the expansion (for a linear expansion $p = 0, 1$). Therefore we can express u^δ in terms of the local expansion $\phi_p(\xi)$, in the form

$$u^\delta(x) = \sum_{i=0}^{N_{dof}-1} \hat{u}_i \Phi_i(x) = \sum_{e=1}^{N_{el}} \sum_{p=0}^{P} \hat{u}^e_p \phi^e_p(\xi),$$

where P is the polynomial order of the expansion and $\phi^e_p(\xi(x)) = \phi_p([\chi^e]^{-1}(x))$ (the superscript denotes the element in which the function is non-zero). However, as there are more local expansion coefficients $\hat{u}^e_p$ than global expansion coefficients $\hat{u}_i$, some extra constraints are required. For the linear finite element example shown in figure 3 where $P = 1$ and $N_{el} = 3$ the constraint is that the global modes are continuous everywhere, which implies

$$\begin{aligned} \hat{u}^1_1 &= \hat{u}^2_0 \\ \hat{u}^2_1 &= \hat{u}^3_0. \end{aligned} \tag{40}$$

The relationship between the local and global expansion coefficients is therefore

$$\begin{array}{lcll} \hat{u}_0^1 & = & \hat{u}_0 & \\ \hat{u}_1^1 & = & \hat{u}_0^2 & = \hat{u}_1 \\ \hat{u}_1^2 & = & \hat{u}_0^3 & = \hat{u}_2 \\ \hat{u}_1^3 & = & \hat{u}_3. & \end{array} \tag{41}$$

For this example, it can be seen that the local representation of the function has 6 elemental degrees of freedom $[(N_{eof} = N_{el} \cdot (P+1) = 6)]$ but only 4 global degrees of freedom ($N_{dof} = 4$). The two constraints shown in equation (40) ensure that $u^\delta(x)$ is C^0 continuous, which is a sufficient condition to ensure that the expansion is in H^1 space and thereby can be an admissible function for the trial space $\mathcal{X}^\delta$.

We can also consider the reverse operation of going from the local degrees of freedom to the global degrees of freedom. This is advantageous since we can perform operations locally within the elements and then assemble the global operation. In the Galerkin formulation this assembly process is typically associated with an integral operation, which implies that we have to sum the local (elemental) contributions. For example, if we consider the integral of $u^\delta(x)$ with the global mode $\Phi_1(x)$ shown in figure 3, we find

$$\begin{aligned} \int_\Omega \Phi_1(x) u^\delta(x) dx &= \int_{\Omega_1} \Phi_1(x) u^\delta(x) dx + \int_{\Omega_2} \Phi_1(x) u^\delta(x) dx, \\ &= \int_{-1}^{1} \phi_1^1(\xi) u^\delta(\chi^1) \frac{d\chi^1}{d\xi} d\xi + \int_{-1}^{1} \phi_0^2(\xi) u^\delta(\chi^2) \frac{d\chi^2}{d\xi} d\xi. \end{aligned}$$

From this we see that all integrals may also be computed in a standard region ($[-1, 1]$). The process of re-assembling the global expression from the local expression on the elemental domains is called *global assembly* or *direct stiffness summation*.

To construct a more general description of the global assembly procedure we introduce $\hat{\boldsymbol{u}}_g$ to denote a vector of all global coefficients,

$$\hat{\boldsymbol{u}}_g = [\hat{u}_0, \ldots, \hat{u}_{N_{dof}-1}]^T$$

and if $\hat{\boldsymbol{u}}^e$ is a vector of the local coefficients in element e (for example $\hat{\boldsymbol{u}}^e = [\hat{u}_0^e, \hat{u}_1^e]$

in our example), then the vector of all local coefficients, denoted by $\hat{\boldsymbol{u}}_l$, is

$$\hat{\boldsymbol{u}}_l = \begin{bmatrix} \hat{\boldsymbol{u}}^1 \\ \hat{\boldsymbol{u}}^2 \\ \cdots \\ \hat{\boldsymbol{u}}^{N_{el}} \end{bmatrix}.$$

The construction of the local degrees of freedom from the global degrees can be expressed in terms of an assembly matrix $\boldsymbol{\mathcal{A}}$ as

$$\boxed{\hat{\boldsymbol{u}}_l = \boldsymbol{\mathcal{A}}\hat{\boldsymbol{u}}_g} \tag{42}$$

where $\boldsymbol{\mathcal{A}}$ is a very sparse matrix whose entries are typically 1 (but can also be -1 in multiple dimensions). For our example in figures 3 the full form of equation (42) is

$$\hat{\boldsymbol{u}}_l = \begin{bmatrix} \hat{u}_0^1 \\ \hat{u}_1^1 \\ \hat{u}_0^2 \\ \hat{u}_1^2 \\ \hat{u}_0^3 \\ \hat{u}_1^3 \end{bmatrix} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & 1 & & \\ & & 1 & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} \hat{u}_0 \\ \hat{u}_1 \\ \hat{u}_2 \\ \hat{u}_3 \end{bmatrix}.$$

which can be directly related to the statement of connectivity given in equation (41). The global assembly operation which assembles the global degrees of freedom from the local contributions is the transpose operation $\boldsymbol{\mathcal{A}}^T$, that is,

$$\boxed{\hat{\boldsymbol{u}}_g = \boldsymbol{\mathcal{A}}^T\hat{\boldsymbol{u}}_l}. \tag{43}$$

Remark: We note that

$$\hat{\boldsymbol{u}}_g \neq \boldsymbol{\mathcal{A}}^T\boldsymbol{\mathcal{A}}\hat{\boldsymbol{u}}_g$$

which can be seen by multiply the matrix $\boldsymbol{\mathcal{A}}$ in the above example by its tranpsoe. If we do this we get a diagonal matrix whose diagonal containd the *multiplicity* of the data at the element boundaries (in this 1D case it is always 2). This occurs since the operation $\boldsymbol{\mathcal{A}}$ scatters the global degrees of freedom to the local elements. $\boldsymbol{\mathcal{A}}^T$, however, assembles the global contribution by *adding* together terms of the local degrees of freedom but not averaging the terms from

either side of the matrix. Since we could multiply every row of the Galerkin problems as given by equation (**??**) by an arbitrary constant we could rescale the $\boldsymbol{\mathcal{A}}$ operator so that its the above statement is true and this may well be the case in other text books.

$\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{A}}^T$ describe the principal operations required for the Galerkin finite element method. It can be appreciated that only global modes which are split into elemental contributions will have multiple entries in the columns of the $\boldsymbol{\mathcal{A}}$ matrix. For a linear expansion this normally involves any global degree of freedom which is not on a boundary.

In general, we do not construct the assembly matrix $\boldsymbol{\mathcal{A}}$ as it is very sparse and therefore numerically very inefficient to store. However, an equivalent numerical operation is to use a mapping array for each element which contains the global location of every local degree of freedom. If we denote this array by "map$[e][i]$" where e denotes the element and i is the local mode index then for our example the array would be defined as:

$$\text{map}[1][i] = \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right\} \qquad \text{map}[2][i] = \left\{ \begin{array}{c} 1 \\ 2 \end{array} \right\} \qquad \text{map}[3][i] = \left\{ \begin{array}{c} 2 \\ 3 \end{array} \right\}.$$

The scatter operation denoted by $\boldsymbol{\mathcal{A}}$ [see equation (42)] can then be evaluated as

$$\left.\begin{array}{l} \text{Do } e = 1, N_{el} \\ \quad \text{Do } i = 0, N_m^e - 1 \\ \quad\quad \hat{\boldsymbol{u}}^e[i] = \hat{\boldsymbol{u}}_g[\text{map}[e][i]] \\ \quad \text{continue} \\ \text{continue} \end{array}\right\} \Leftrightarrow \hat{\boldsymbol{u}}_l = \boldsymbol{\mathcal{A}}\hat{\boldsymbol{u}}_g,$$

where $N_m^e = P^e + 1$, and P^e is the polynomial order in element 'e' and so N_m^e is equal to 2 is our example since $P^e = 1$. Similarly, the global assembly operation, $\boldsymbol{\mathcal{A}}^T$, may be written as

$$\left.\begin{array}{l} \text{Do } e = 1, N_{el} \\ \quad \text{Do } i = 0, N_m^e - 1 \\ \quad\quad \hat{\boldsymbol{u}}_g[\text{map}[e][i]] = \hat{\boldsymbol{u}}_g[\text{map}[e][i]] + \hat{\boldsymbol{u}}^e[i] \\ \quad \text{continue} \\ \text{continue} \end{array}\right\} \Leftrightarrow \hat{\boldsymbol{u}}_g = \boldsymbol{\mathcal{A}}^T\hat{\boldsymbol{u}}_l$$

4.2 Numerical Integration

Up to this point we have assumed that all integrals are evaluated analytically. As we have seen, within each elemental domain we want to evaluate integrals of the form

$$\int_{-1}^{1} u(\xi) d\xi. \tag{44}$$

The form of $u(\xi)$ is, however, problem specific and therefore we need an automated way to evaluate such integrals. This suggests the use of numerical integration or *quadrature*. The fundamental concept is the approximation of the integral by a finite summation of the form

$$\int_{-1}^{1} u(\xi) d\xi \approx \sum_{i=0}^{Q-1} w_i u(\xi_i),$$

where w_i are specified constants or *weights* and ξ_i represent an abscissa of Q distinct points in the interval $-1 \leq \xi_i \leq 1$. Sometimes these points are also referred to as *zeros*. Although there are many different types of numerical integration we shall restrict our attention primarily to *Gaussian quadrature*. However we note that the lowest three orders of the Gauss-Lobatto-Legendre quadrature we shall demonstrate is the centrepoint rule, the trapezoidal rule and Simpson's rule.

4.2.1 Gaussian Quadrature

Gaussian quadrature is a particularly accurate method for treating integrals where the integrand, $u(\xi)$, is smooth. In this technique the integrand is represented as a Lagrange polynomial using the Q points ξ_i, which are to be specified, (a Lagrange polynomial had a value of 1 at one coordinate ξ_i and is zero at all others), that is,

$$u(\xi) = \sum_{i=0}^{Q-1} u(\xi_i) h_i(\xi) + \varepsilon(u), \tag{45}$$

where $\varepsilon(u)$ is the approximation error. If we substitute equation (45) into (44) we obtain a representation of the integral as a summation:

$$\int_{-1}^{1} u(\xi) d\xi = \sum_{i=0}^{Q-1} w_i u(\xi_i) + R(u), \tag{46}$$

where

$$w_i = \int_{-1}^{1} h_i(\xi) d\xi, \tag{47}$$

$$R(u) = \int_{-1}^{1} \varepsilon(u) d\xi. \tag{48}$$

Equation (47) defines the weights w_i in terms of the integral of the Lagrange polynomial but to perform this integration we need to know the location of the points ξ_i. Since $u(\xi)$ is represented by a polynomial of order $Q-1$ we would expect the relation above to be exact if $u(\xi)$ is a polynomial of order $Q-1$ or less (mathmatically we say when $u(\xi) \in \mathcal{P}_{Q-1}([-1,1])$ then $R(u) = 0$).

This would be true if, for example, we choose the points so that they are equispaced in the interval. There is, however, a better choice of abscissae which permits exact integration of polynomials of higher order than $Q-1$. This remarkable fact was first recognized by Gauss and is at the heart of Gaussian quadrature. Here we will only present the result of the Gauss quadrature for integrals of the type shown in equation (46) known as Legendre integration. There are three different types of Gauss quadrature known as Gauss, Gauss-Radau, and Gauss-Lobatto, respectively. The difference between the three types of quadrature lies in the choice of the points ξ_i:

- Gauss quadrature uses points which has points that are interior to the interval, $-1 < \xi_i < 1$ for $i = 0, \ldots, Q-1$, i.e. internal to the interval,
- Gauss-Radau quadrature the points include one of the end-points of the interval, usually $\xi = -1$ and
- Gauss-Lobatto quadrature the points include both end points of the interval, that is, $\xi = \pm 1$.

Introducing $\xi_{i,P}^{\alpha,\beta}$ to denote the P zeros of the Pth order Jacobi polynomial $P_P^{\alpha,\beta}$ such that

$$P_P^{\alpha,\beta}(\xi_{i,P}^{\alpha,\beta}) = 0, \qquad i = 0, 1, \ldots, P-1,$$

where

$$\xi_{0,P}^{\alpha,\beta} < \xi_{1,P}^{\alpha,\beta} < \cdots < \xi_{P-1,P}^{\alpha,\beta},$$

we can define zeros and weights which approximate the integral

$$\int_{-1}^{1} u(\xi)d\xi = \sum_{i=0}^{Q-1} w_i u(\xi_i) + R(u),$$

as:

(1) *Gauss-Legendre*

$$\begin{aligned} \xi_i &= \xi_{i,Q}^{0,0} \qquad i = 0, \ldots, Q-1 \\ w_i^{0,0} &= \frac{2}{[1-(\xi_i)^2]} \left[\frac{d}{d\xi} \left(L_Q(\xi) \right)|_{\xi=\xi_i} \right]^{-2} \qquad i = 0, \ldots, Q-1 \\ R(u) &= 0 \qquad \text{if } u(\xi) \in \mathcal{P}_{2Q-1}([-1,1]) \end{aligned}$$

(2) *Gauss-Radau-Legendre*

$$\begin{aligned} \xi_i &= \begin{cases} -1 & i = 0 \\ \xi_{i-1,Q-1}^{0,1} & i = 1, \ldots, Q-1 \end{cases} \\ w_i^{0,0} &= \frac{(1-\xi_i)}{Q^2[L_{Q-1}(\xi_i)]^2} \qquad i = 0, \ldots, Q-1 \\ R(u) &= 0 \qquad \text{if } u(\xi) \in \mathcal{P}_{2Q-2}([-1,1]) \end{aligned}$$

(3) *Gauss-Lobatto-Legendre*

$$\begin{aligned} \xi_i &= \begin{cases} -1 & i = 0 \\ \xi_{i-1,Q-2}^{1,1} & i = 1, \ldots, Q-2 \\ 1 & i = Q-1 \end{cases} \\ w_i^{0,0} &= \frac{2}{Q(Q-1)[L_{Q-1}(\xi_i)]^2} \qquad i = 0, \ldots, Q-1 \\ R(u) &= 0 \qquad \text{if } u(\xi) \in \mathcal{P}_{2Q-3}([-1.1]) \end{aligned}$$

In the above formulae $L_Q(\xi)$ is the Legendre polynomial ($L_Q(\xi) = P_Q^{0,0}(\xi)$). The zeros of the Jacobi polynomial $\xi_{i,m}^{\alpha,\beta}$ do not have an analytic form and commonly the zeros and weights are tabulated. Tabulation of data can lead to copying errors and therefore a better way to evaluate the zeros is the use of a numerical algorithm such as a newton Rhapson technique. Having determined the zeros, the weights can be evaluated from the formulae. This is done by generating the Legendre polynomial from a recursion relationship.

4.3 Differentiation

To evaluate the Poisson equation we also need to know how to differentiate our expansion bases which we recall are normally polynomials. Assuming a polynomial approximation of the form:

$$u^\delta(\xi) = \sum_{p=0}^{P} \hat{u}_p \phi_p(\xi),$$

where $\phi_p(\xi)$ could represent any basis we want for example a linear polynomial when $P = 1$, we can differentiate this expression to obtain

$$\frac{du^\delta(\xi)}{d\xi} = \sum_{p=0}^{P} \hat{u}_p \frac{d\phi_p(\xi)}{d\xi}.$$

The differentiation of $u^\delta(\xi)$ is therefore dependent on evaluating $d\phi_p(\xi)/d\xi$. In this section we shall consider the case where $\phi_p(\xi)$ is the Lagrange polynomial $h_p(\xi)$. Differentiation of this form is often referred to as differentiation in physical space or *collocation differentiation*.

Any polynomial expansion can be represented in terms of Lagrange polynomials which can then be differentiated in a similar fashion. If we assume that $u^\delta(\xi)$ is a polynomial of order equal to or less than P (i.e. $u^\delta(\xi) \in \mathcal{P}_P([-1,1])$), then it can be exactly expressed in terms of Lagrange polynomials $h_i(\xi)$ through a set of Q nodal points ξ_i $(0 \le i \le Q-1)$ as

$$u(\xi) = \sum_{i=0}^{Q-1} u(\xi_i) h_i(\xi), \qquad h_i(\xi) = \frac{\Pi_{j=0, j\neq i}^{Q-1}(\xi - \xi_j)}{\Pi_{j=0, j\neq i}^{Q-1}(\xi_i - \xi_j)}$$

where $Q \ge P + 1$. Therefore the derivative of $u(\xi)$ can be represented as

$$\frac{du(\xi)}{d\xi} = \sum_{i=0}^{Q-1} u(\xi_i) \frac{d}{d\xi} h_i(\xi).$$

Typically, we only need the derivative at the nodal points ξ_i since we will then integrate the expression using Gaussian quadrature. The discrete derivature is given by

$$\left.\frac{du(\xi)}{d\xi}\right|_{\xi=\xi_i} = \sum_{j=0}^{Q-1} d_{ij}\, u(\xi_i),$$

where

$$d_{ij} = \left.\frac{dh_j(\xi)}{d\xi}\right|_{\xi=\xi_i}.$$

So we need to calcuate an expression for d_{ij}. We note that an alternative representation of the Lagrange polynomial is

$$h_i(\xi) = \frac{g_Q(\xi)}{g'_Q(\xi_i)(\xi-\xi_i)}, \qquad g_Q(\xi) = \prod_{j=0}^{Q-1}(\xi-\xi_j).$$

Taking the derivative of $h_i(\xi)$ we obtain

$$\frac{dh_i(\xi)}{d\xi} = \frac{g'_Q(\xi)(\xi-\xi_i) - g_Q(\xi)}{g'_Q(\xi_i)(\xi-\xi_i)^2}.$$

Finally, noting that the numerator and denominator of this expression are zero as $\xi \to \xi_i$, and because $g_Q(\xi_i) = 0$ by definition then using L'hopitals rule:

$$\lim_{\xi\to\xi_i}\frac{dh_i(\xi)}{d\xi} = \lim_{\xi\to\xi_i}\frac{g''_Q(\xi)}{2g'_Q(\xi_i)} = \frac{g''_Q(\xi_i)}{2g'_Q(\xi_i)}.$$

so we can write d_{ij} as

$$d_{ij} = \begin{cases} \dfrac{g'_Q(\xi_i)}{g'_Q(\xi_j)}\dfrac{1}{(\xi_i-\xi_j)} & i \neq j, \\ \\ \dfrac{g''_Q(\xi_i)}{2g'_Q(\xi_i)} & i = j. \end{cases} \tag{49}$$

Equation (49) is the general representation of the derivative of the Lagrange polynomials evaluated at the nodal points ξ_i $(0 \le i \le Q-1)$. To proceed further we need to know specific information about the nodal points ξ_i which will allow us to deduce alternative forms of $g'_Q(\xi_i)$ and $g''_Q(\xi_i)$.

For the three Gauss-quadrature cases discussed in section 4.2.1, and recalling the definition of $\xi^{\alpha,\beta}$ and $L_Q(\xi)$ from section 4.2.1, the formulae become:
(1) *Gauss-Legendre*

$$\xi_i = \xi^{0,0}_{i,Q}$$

$$d_{ij} = \begin{cases} \dfrac{L'_Q(\xi_i)}{L'_Q(\xi_j)(\xi_i - \xi_j)} & i \neq j, 0 \leq i, j \leq Q-1 \\ \\ \dfrac{\xi_i}{(1-\xi_i^2)} & i = j \end{cases}$$

(2) *Gauss-Radau-Legendre*

$$\xi_i = \begin{cases} -1 & i = 0 \\ \xi^{0,1}_{i-1,Q-1} & i = 1, \ldots, Q-1 \end{cases}$$

$$d_{ij} = \begin{cases} \dfrac{-(Q-1)(Q+1)}{4} & i = j = 0 \\ \\ \dfrac{L_{Q-1}(\xi_i)}{L_{Q-1}(\xi_j)} \dfrac{(1-\xi_j)}{(1-\xi_i)} \dfrac{1}{(\xi_i - \xi_j)} & i \neq j, 0 \leq i, j \leq Q-1 \\ \\ \dfrac{1}{2(1-\xi_i)} & 1 \leq i = j \leq Q-1 \end{cases}$$

(3) *Gauss-Lobatto-Legendre*

$$\xi_i = \begin{cases} -1 & i = 0 \\ \xi^{1,1}_{i-1,Q-2} & i = 1, \ldots, Q-2 \\ 1 & i = Q-1 \end{cases}$$

$$d_{ij} = \begin{cases} \dfrac{-Q(Q-1)}{4} & i = j = 0 \\ \\ \dfrac{L_{Q-1}(\xi_i)}{L_{Q-1}(\xi_j)} \dfrac{1}{(\xi_i - \xi_j)} & i \neq j, 0 \leq i, j \leq Q-1 \\ \\ 0 & 1 \leq i = j \leq Q-2 \\ \\ \dfrac{Q(Q-1)}{4} & i = j = Q-1 \end{cases}$$

4.4 Example of an elemental decompositions for Helmholtz problem

The Helmholtz equation arises when discretising the heat equation with an implicit Euler backward finite difference scheme in time can be written as

$$\frac{\partial^2 u}{\partial x^2} - \lambda u = f \tag{50}$$

As in section 3.2 we consider a region $0 \leq x \leq 1$ with bounday conditions $u(0) = 1$ and $\frac{\partial u}{\partial x}(1) = 1$.

We construct the weak Galerkin form by testing equation (50) against a numerical test function v^δ and then integrating the second derivative by parts to arrive at:

$$\int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u^\delta}{\partial x}dx + \lambda \int_0^1 v^\delta u^\delta dx = -\int_0^1 v^\delta f dx + \left[v^\delta \frac{\partial u^\delta}{\partial x}\right]_0^1. \tag{51}$$

We recall that we can homogonise the problem by making the decomposition:

$$u(x) = u^{\mathcal{H}}(x) + u^{\mathcal{D}}(x) \text{ where } \begin{cases} u^{\mathcal{D}}(0) = 1 \\ u^{\mathcal{H}}(0) = 0 \end{cases}$$

Since $u^{\mathcal{D}}$ is known equation (51) can be recast as

$$\int_0^1 \left[\frac{\partial v^\delta}{\partial x}\frac{\partial u^{\mathcal{H}}}{\partial x} + \lambda v^\delta u^{\mathcal{H}}\right] dx = -\int_0^1 \left[v^\delta f + \frac{\partial v^\delta}{\partial x}\frac{\partial u^{\mathcal{D}}}{\partial x} + \lambda v^\delta u^{\mathcal{D}}\right] dx + \left[v^\delta \frac{\partial u^\delta}{\partial x}\right]^1. \tag{52}$$

We note that the term $\int_0^1 \frac{\partial v^\delta}{\partial x}\frac{\partial u^{\mathcal{H}}}{\partial x}dx$ is the weak Laplacian and will lead to the *Laplacian matrix*. The term $\int_0^1 v^\delta u^{\mathcal{H}} dx$ will lead to the *mass matrix*.

We once again consider the two element computational region shown in figure 4 and therefore the solution can be represented as

$$u^\delta = \sum_{i=0}^{2} \hat{u}_i \Phi_i(x) = \sum_{e=1}^{2}\sum_{i=0}^{P=1} \hat{u}_i^e \phi_i(x)$$

where the second part of the expression represents the elemental representation and requires some connectivity to be applied to generate the global expansion. We note that we can define the Dirichlet and Homogeneous solutions as:

$$u^{\mathcal{D}} = \hat{u}_0 \Phi_0$$
$$u^{\mathcal{H}} = \hat{u}_1 \Phi_1 + \hat{u}_2 \Phi_2$$

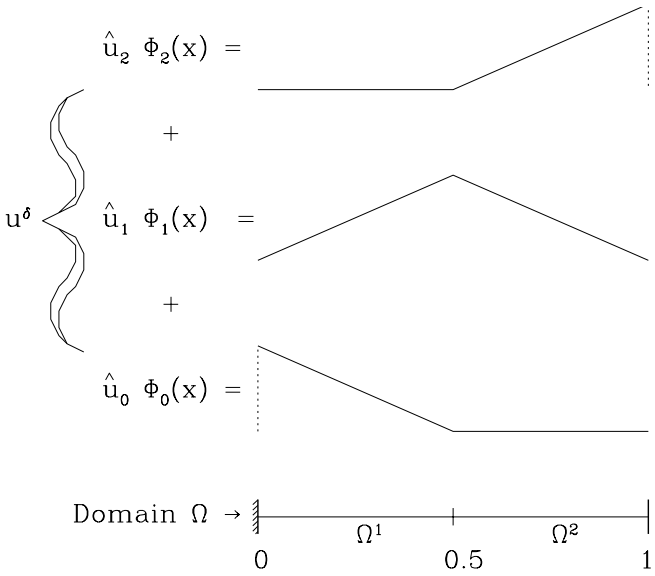


Figure 4: Linear finite element approximation $u^\delta(x) = \sum_{i=0}^{2} \hat{u}_i \Phi_i(x)$, in a domain Ω, using two elemental subdomains, Ω^1 and Ω^2.

where $\hat{u}_0 = 1$. This choice for $u^{\mathcal{D}}$ is equivalent to manipulating the matrix system as we will shown shortly. In general we are free to choose any function for $u^{\mathcal{D}}$ which satisifies the boundary condition.

4.4.1 Gloabl matrix construction from elemental components

We will now discuss how to generate the global mass matrix, $\boldsymbol{M}$, from its elemental components. An analagous technique can be used for the Laplacian matrix, $\boldsymbol{L}$, after the derivative terms have been evaluated. We recall that following a Galerkin construction we choose the test space to be the same as the trial spaces and so $v^\delta = \sum_i^{N_{dof}} \hat{v}\Phi_i$, $u^\delta = \sum_j^{N_{dof}} \hat{u}_j \Phi_j$. The second LHS term in equation (52) was equivalent to

$$\int_\Omega v^\delta u^\delta dx = \sum_i \hat{v}_i \int_\Omega \Phi_i \left\{ \sum_j \Phi_j \hat{u}_j \right\} dx \quad \Rightarrow \quad \hat{\mathbf{v}}^T \boldsymbol{M} \hat{\mathbf{u}}$$

where

$$\boldsymbol{M}[i,j] = \int_\Omega \Phi_i \Phi_j dx, \quad \hat{\mathbf{v}} = [\hat{v}_0, \ldots, \hat{v}_{N_{dof}}]^T, \quad \hat{\mathbf{u}} = [\hat{u}_0, \ldots, \hat{u}_{N_{dof}}]^T$$

Now the mass matrix, $\boldsymbol{M}$, can be generated from the elemental version of the mass matrix which we will denote as, $\boldsymbol{M}^e$, using the global assembly process discussed in section 4.1.4, i.e.

$$\boldsymbol{M} = \boldsymbol{\mathcal{A}}^T \underline{\boldsymbol{M}}^e \boldsymbol{\mathcal{A}}$$

where the Underlined matrix denotes a block diagonal matrix of components, $\boldsymbol{M}^1, \cdots, \boldsymbol{M}^{N_{el}}$.

To proceed we need to define how to construct $\boldsymbol{M}^e$,

$$\boldsymbol{M}^e[i,j] = \int_{\Omega_e} \phi_i \phi_j dx = \int_{\Omega_{st}} \phi_i \phi_j J^e d\xi, \qquad J^e = \frac{dx^e}{d\xi}$$

where ϕ_i is the local bases and J^e is the jacobian mapping the elemental region Ω^e to the standard element Ω_{st}. For the e^{th} element we need to define a mapping between the upper x_u^e and lower x_l^e of the element 'e' and the standard element $-1 \leq \xi \leq 1$. In 1D the most straight forards mapping is a linear affine mapping of the form

$$x^e(\xi) = x_l^e \frac{1-\xi}{2} + x_u^e \frac{1+\xi}{2} = \frac{1+\xi}{2}(x_u^e - x_l^e) + x_l^e. \tag{53}$$

The middle expression illustrates the isoparametric nature of this mapping for a linear finite element expansion. In multiple dimensions we may need to use more complicated mappings particularly if the element is curved. From equation (53) we can define the Jacobian as

$$J^e = \frac{dx^e}{d\xi} = \frac{x_u^e - x_l^e}{2}.$$

Finally we can evaulate the local gaussian quadrature as:

$$\boldsymbol{M}^e[i,j] = \int_{-1}^{1} \phi_i(\xi)\phi_j(\xi) J^e d\xi = \sum_{k=0}^{Q-1} \phi_i(\xi_k)\phi_j(\xi_k) J^e w_k.$$

If ϕ_i, ϕ_j are linear (i.e. $P = 1$) then the integrand, $\phi_i \phi_j J^e$ is a quadratic polynomial and so we should use three points ($Q = 3$) to evaluate the integral exactly. For the mesh defined in figure 4 the global connectivity and assembly matrix $\boldsymbol{\mathcal{A}}$ are defined as:

$$\left.\begin{array}{rcl} \hat{u}_0^1 &=& \hat{u}_0 \\ \hat{u}_1^1 &=& \hat{u}_0^2 \quad = \quad \hat{u}_1 \\ \hat{u}_1^2 &=& \hat{u}_2. \end{array}\right\} \quad \hat{\boldsymbol{u}}_l = \begin{bmatrix} \hat{u}_0^1 \\ \hat{u}_1^1 \\ \hat{u}_0^2 \\ \hat{u}_1^2 \end{bmatrix} = \begin{bmatrix} 1 & & \\ & 1 & \\ & 1 & \\ & & 1 \end{bmatrix} \begin{bmatrix} \hat{u}_0 \\ \hat{u}_1 \\ \hat{u}_2 \end{bmatrix} = \boldsymbol{\mathcal{A}} \hat{\boldsymbol{u}}_g.$$

Therefore for our example the full mass matrix is:

$$\boldsymbol{M} = \boldsymbol{\mathcal{A}}^T \underline{\boldsymbol{M}}^e \boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{A}}^T \begin{bmatrix} \boldsymbol{M}^1 & 0 \\ 0 & \boldsymbol{M}^2 \end{bmatrix} \boldsymbol{\mathcal{A}}.$$

In practice we recall that we should not in general generate a matrix $\boldsymbol{\mathcal{A}}$ but use a mapping as discussed in section 4.1.4. We note that $\boldsymbol{M}$ is the global mass matrix system including all degrees of freedom independent of whether they maybe Dirichlet or Neumann boundary conditions. When we have Dirichlet conditions we have seen previously that we can decompose the solution in to an homogeneous ($u^{\mathcal{H}}$) and Dirichlet components ($u^{\mathcal{D}}$). To handle this decomposition we have two choices

1. Only assemble the homogeneous components by using a mapping which does not include the Dirichlet components (or puts them at the end of the global matrix).

2. Generate the full matrix and then extract the submatrix which to the corresponding homogenous degrees of freedom. This is quite straight forward in 1D but for higher dimentions the first method is more efficient. However as an example if we listed all the unknown degrees of freeom in $\hat{\boldsymbol{u}}_g$ first and denote them as $\hat{\boldsymbol{u}}_g^{\mathcal{H}}$ followed by all the known Dirichlet degrees of freedom, $\hat{\boldsymbol{u}}_g^{\mathcal{D}}$, then the mass matrix systems could be written as

$$\boldsymbol{M} \hat{\boldsymbol{u}}_g = \begin{bmatrix} \boldsymbol{M}^{\mathcal{H}\mathcal{H}} & \boldsymbol{M}^{\mathcal{H}\mathcal{D}} \\ \boldsymbol{M}^{\mathcal{D}\mathcal{H}} & \boldsymbol{M}^{\mathcal{D}\mathcal{D}} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{u}}_g^{\mathcal{H}} \\ \hat{\boldsymbol{u}}_g^{\mathcal{D}} \end{bmatrix}$$

   - Submatrix $\boldsymbol{M}^{\mathcal{H}\mathcal{H}}$ is the matrix which need assembling and inverting

- $\boldsymbol{M}^{\mathcal{HD}}\hat{\boldsymbol{u}}_g^{\mathcal{D}}$ is equivalent to the RHS component $\int_0^1 v^\delta u^{\mathcal{D}} dx$ in equation (52).
- $\boldsymbol{M}^{\mathcal{DH}}, \boldsymbol{M}^{\mathcal{DD}}$ are never needed.

The other LHS term in equation (52) is the weak Laplacian matrix

$$\boldsymbol{L}[i,j] = \int_\Omega \frac{\partial \Phi_i}{\partial x}\frac{\partial \Phi_j}{\partial x} dx = \boldsymbol{\mathcal{A}}^T \underline{\boldsymbol{L}}^e \boldsymbol{\mathcal{A}}$$

where

$$\boldsymbol{L}^e[i,j] = \int_{\Omega^e} \frac{\partial \phi_i}{\partial x}\frac{\partial \phi_j}{\partial x} dx = \int_{\Omega^{st}} \frac{\partial \phi_i}{\partial x}\frac{\partial \phi_j}{\partial x} J^e d\xi.$$

The only difference between this matrix and the mass matrix from the implementation point of view is that we need to calcuate $\frac{\partial \phi_i}{\partial x}$ which can be evaluated by applying the chain rule, i.e. $\frac{\partial \phi_i}{\partial x} = \frac{\partial \phi_i}{\partial \xi}\frac{\partial \xi}{\partial x}$. So the elemental Laplacian matrix can be evaluated as

$$\boldsymbol{L}^e[i,j] = \int_{\Omega^{st}} \frac{\partial \phi_i}{\partial \xi}\frac{\partial \phi_j}{\partial \xi}\left(\frac{\partial \xi}{\partial x}\right)^2 J^e d\xi.$$

To evaluate the integration we once again use the Guassian quadrature discused in section 4.2.1. This means that we need to know the derivatives at the quadrature points. Since the derivatives are now in terms of the local coordinates ξ we can evaluate them at the quadrature points using the techniques discussed in section 4.3, i.e. $\frac{\partial \phi}{\partial x}(\xi_i) = \sum_j D_{ij}\phi(\xi_j)$.

4.4.2 Evaluation of RHS terms

The RHS terms in equation (52) are treated very analagously to the previous section however since we have explicit knowledge of these functions we only need to do one $\boldsymbol{\mathcal{A}}^T$ assembly process. For example the first RHS term of equation (52), using the mesh in figure 4 can be treated as

$$\int_\Omega v^\delta f(x) dx = \sum_i \hat{v}_i \int_\Omega \Phi_i f(x) dx \Rightarrow \hat{\boldsymbol{f}}_g = \boldsymbol{\mathcal{A}}^T \begin{bmatrix} \boldsymbol{f}^1 \\ \boldsymbol{f}^2 \end{bmatrix},$$

where $\boldsymbol{f}^e$ is an elemental matrix containing the inner product of $f(x)$ with the local basis, i.e.

$$\boldsymbol{f}^e[j] = \int_{\Omega^e} \phi_j f(x) dx.$$

This integral expression can be evaluated using Gaussian quadrature by either evaluating the local function at the quadrature points ($f(\xi_k)$) or representing the function in a finite element expansion $f(x) \approx \sum_{e=1}^{N_{el}} \sum_{i=0}^{i=P} \hat{f}_i^e \phi_i$.

4.4.3 Summary of methodology

In summary the discrete Helmholtz matrix problem

$$\left[\boldsymbol{L}^{\mathcal{H}\mathcal{H}} + \lambda \boldsymbol{M}^{\mathcal{H}\mathcal{H}}\right] \hat{\boldsymbol{u}}_g^{\mathcal{H}} = \hat{\boldsymbol{f}}_g^*$$

can be solved as:

1. Construct elemental matrices $\boldsymbol{M}^e, \boldsymbol{L}^e$
2. Assemble global system for homogeneous degrees of freedom $\boldsymbol{M}^{\mathcal{H}\mathcal{H}}, \boldsymbol{L}^{\mathcal{H}\mathcal{H}}$
3. Construct modified RHS vector which includes the Dirichlet and Neumann boundary contribution, $\hat{\boldsymbol{f}}_g^*$.
4. Invert $\boldsymbol{L}^{\mathcal{H}\mathcal{H}} + \lambda \boldsymbol{M}^{\mathcal{H}\mathcal{H}}$ (either directly or iteratively) to get $\hat{\boldsymbol{u}}_g^{\mathcal{H}}$.
5. Recover the solution $u^\delta = u^{\mathcal{H}} + u^{\mathcal{D}}$.

4.4.4 Properties of the Mass and Laplacian Matrices

To complete our example it is worth noting some properties of the Mass and Laplcian matrices which help provide useful debugging checks as well as providing information about the appropriate type of solvers to be used for matrix inversion. The key property of the mass matrix is that it is symmetric and positive definite. Postitive definiteness means that all its eigenvalues are strictly positive which is useful since it means that iterative solver such as the conjugate gradient method can be applied to invert this system.

To demonstrate this property we note that an alternative but equivalent definition of positive definiteness of the matrix $\boldsymbol{M}$ is that

$$\hat{\boldsymbol{u}}^T \boldsymbol{M} \hat{\boldsymbol{u}} > 0 \quad \text{for all} \quad \hat{\boldsymbol{u}}.$$

Therefore for any non-zero vector $\hat{\boldsymbol{u}}$ pre- and post-multiplication will lead to a value which is always positive. Recalling that the defintion of the mass matrix

is:

$$\boldsymbol{M}[i,j] = \int_\Omega \Phi_i \Phi_j dx$$

then the i^{th} component of $\boldsymbol{M}\hat{\boldsymbol{u}}$ is

$$\left(\boldsymbol{M}\hat{\boldsymbol{u}}\right)[i] = \int_\Omega \Phi_i \sum_j \Phi_j \hat{u}_j dx = \int_\Omega \Phi_i u^\delta dx,$$

and the inner product of $\hat{\boldsymbol{u}}$ with $\boldsymbol{M}\hat{\boldsymbol{u}}$ is

$$\hat{\boldsymbol{u}}^T \boldsymbol{M}\hat{\boldsymbol{u}} = \int_\Omega \left\{ \sum_i \hat{u}_i \Phi_i \right\} u^\delta dx = \int_\Omega u^\delta u^\delta dx$$

So the condition of positive definiteness simply boils down to the integral of $(u^\delta)^2$ over the region Ω which has to be positive unless $u^\delta = 0$.

This property also provides a useful debugging check. If the vector u^δ represents a constant of unit value which for a linear finite element expansion means the entries of the vectors are all 1 then summing all entries of the mass matrix will should give us the area of the region Ω, i.e.

$$\text{if } \hat{\boldsymbol{u}} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \quad \text{then} \quad \hat{\boldsymbol{u}}^T \boldsymbol{M}\hat{\boldsymbol{u}} = \int_\Omega u^\delta u^\delta dx = \int_\Omega dx = \Omega.$$

Similarly summing the i^{th} row of the matrix (or column) is the same as integrating the global bases Φ_i over the region Ω.

The Laplacian matrix is positive semi-definite which means that all eigenvalues are positive or zero but not negative. An eigenvalue can be zero since if we have a vector $\hat{\boldsymbol{u}}$ which represents a constant, i.e. $u^\delta = 1$ then the derivative of this function will be zero. The positive semi-definite property can be shown in a similar manner to the Mass matrix example above. It also leads to the good debugging property that for a linear finite element Laplacian matrix all rows (or columns) should sum to zero.

4.5 Appendix: h-Convergence of Linear Finite Elements

In this section, we consider the convergence of the linear finite element expansion on a regular mesh in the energy norm. In this case convergence is achieved using the h-type extension, that is, increasing the number of elements so that $h \to 0$.

We consider the one-dimensional Helmholtz problem described in section 3.3, which is stated in weak form as
Find $u \in \mathcal{X}$ such that

$$\int_0^l \left[\frac{\partial v}{\partial x}\frac{\partial u}{\partial x} + \lambda v u\right] dx = \int_0^l v f dx \qquad \text{for all } v \in \mathcal{V}. \tag{54}$$

We assume that $f(x)$ and λ are defined so that $u'' = d^2u/dx^2$ is bounded and continuous such that $|u''| \leq C$ in the interval $0 \leq x \leq l$. The energy norm is defined as $||u||_E = \sqrt{a(u,u)}$ where $a(v,u)$ is the left-hand side of equation (54), i.e.

$$(||u||_E)^2 = a(u,u) = \int_0^l \left[\frac{\partial u}{\partial x}\frac{\partial u}{\partial x} + \lambda u u\right] dx.$$

To determine the error between the finite element solution u^δ and the exact solution u we consider the error between linear interpolant $\mathbb{I}u$ and u. Since u^δ is the minimal solution to u for all functions in the trial space $\mathcal{X}^\delta$ [see equation (38a)] the interpolation error will bound the error of the finite element solution. Let us consider a uniform mesh as shown in figure 5 where the domain consists of N_{el} elements and each element is of equal length $h = l/N_{el}$. The linear interpolant $\mathbb{I}u$ is a piecewise linear approximation to u such that

$$\mathbb{I}u(jh) = u(jh), \qquad j = 0, \ldots, N_{el}.$$

Therefore, the interpolation error, $\bar{\varepsilon}(x) = u(x) - \mathbb{I}u(x)$, is zero at the end of each element Ω_e. We shall denote the interpolation error in each element by $\bar{\varepsilon}_e$:

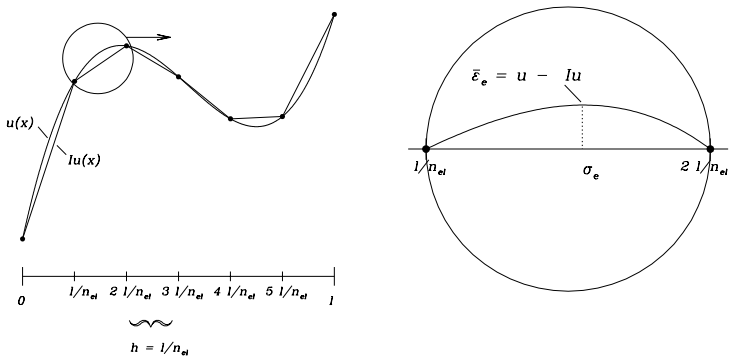
$$\bar{\varepsilon}_e(x) = u(x) - \mathbb{I}u(x), \qquad (e-1)h \leq x \leq eh.$$

Since $\bar{\varepsilon}_e(x)$ vanishes at the ends of each element there is a point in every element, σ_e, where $|\bar{\varepsilon}_e|$ is a maximum (see figure 5). At this point $\bar{\varepsilon}_e'(\sigma_e) = 0$ and so we can write

$$\bar{\varepsilon}_e'(x) = \int_{\sigma_e}^x \bar{\varepsilon}_e''(s) ds = \int_{\sigma_e}^x u''(s) ds, \qquad (e-1)h \leq x \leq eh.$$

The last part of this relation uses the fact that the interpolant, $\mathbb{I}u(x)$ is linear and therefore $[\mathbb{I}u(x)]'' = 0$. As $|u''| \leq C$ we find

$$\max |\bar{\varepsilon}_e'(x)| \leq Ch, \qquad (e-1)h \leq x \leq eh. \tag{55}$$



(a) (b)

Figure 5: (a) Linear interpolation, $\mathbb{I}u(x)$, of the function $u(x)$ in the region $0 \leq x \leq l$ using a uniform mesh with $N_{el} = 6$ elemental domains. Each element is of size $h = l/N_{el}$. (b) The linear interpolant is exact at the endpoints of every element and so there is a point σ_e where the error $\bar{\varepsilon} = u - \mathbb{I}u$ is a maximum.

Assuming that σ_e lies nearer to eh than $(e-1)h$ so that $eh - \sigma_e \leq h/2$ we can, by applying the Taylor expansion to the error $\bar{\varepsilon}_e$ about the point σ_e, represent the error $\bar{\varepsilon}_e$ at eh as

$$\bar{\varepsilon}_e(eh) = \bar{\varepsilon}_e(\sigma_e) + (eh - \sigma_e)\bar{\varepsilon}'_e(\sigma_e) + \frac{(eh - \sigma_e)^2}{2}\bar{\varepsilon}''_e(s), \tag{56}$$

where s is a point between σ_e and eh. [If σ_e had been closer to $(e-1)h$ than eh we could have written the Taylor expansion for the point $(e-1)h$ instead of that for eh and obtained the same result.]

Now, by definition, $\bar{\varepsilon}_e(eh) = 0$ and $\bar{\varepsilon}'_e(\sigma_e) = 0$. Accordingly, on substitution into (56) we deduce that

$$\max|\bar{\varepsilon}_e(\sigma_e)| \leq C\frac{h^2}{8}. \tag{57}$$

We recall that $\bar{\varepsilon}_e(\sigma_e)$ was defined as the point of maximum absolute error in the eth element and so

$$\max|\bar{\varepsilon}_e(x)| \leq |\bar{\varepsilon}_e(\sigma_e)|.$$

Finally, using equation (55) and (57) and noting the square of the error in the energy norm can be written as:

$$\begin{aligned}
\left(||\bar{\varepsilon}||_E\right)^2 \;=\; a(\bar{\varepsilon},\bar{\varepsilon}) \;&=\; \int_0^l \left(\bar{\varepsilon}'\right)^2 + \lambda\left(\bar{\varepsilon}\right)^2 dx \\
&=\; \sum_{e=1}^{N_{el}} \int_{(e-1)h}^{eh} \left(\bar{\varepsilon}_e'\right)^2 + \lambda\left(\bar{\varepsilon}_e\right)^2 dx \\
&\leq\; h\; N_{el} \left((Ch)^2 + \lambda \left(C\tfrac{h^2}{8} \right)^2 \right).
\end{aligned}$$

Noting that that $l = h \cdot N_{el}$, there is a constant K such that

$$\left(||\bar{\varepsilon}||_E\right)^2 \leq lKC^2h^2.$$

Due to the minimal property of the finite element solution, the error in the finite element approximation $\varepsilon = u - u^\delta$ is bounded by $\bar{\varepsilon}$ and so

$$||\varepsilon||_E \leq ||\bar{\varepsilon}||_E \leq K_1 Ch,$$

where C depends on f and λ but is independent of h.

5 Unsteady Problems

5.1 Problem statement

Consider the advection-diffusion equation

$$u_t + au_x = \alpha u_{xx} \tag{58}$$

which as shown in figure 6 has a solution that propagates towards the right at a speed a and diffuses at a rate which is dependent upon α.

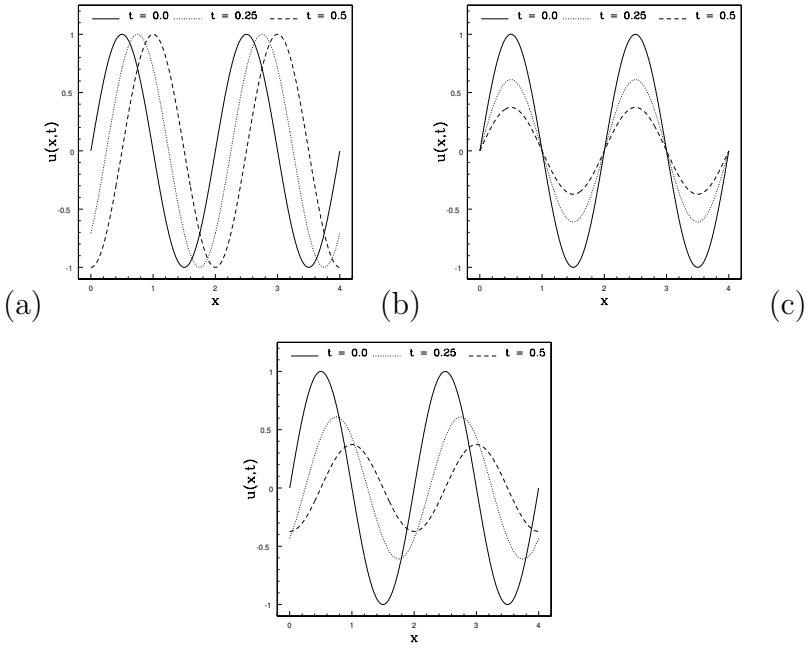


Figure 6: Solution to advection diffusion equation at t = 0.1, 0.25 and 0.5 from a sin wave initial condition. (a) $a = 1, \alpha = 0.0$, (b) $a = 0, \alpha = 0.1$, (c) $a = 1, \alpha = 0.1$

The finite element approximation to the equation is found by multiplying equation (58) by a weight function $w(x)$ and integrating over the solution domain as well as integrating the second derivative term by parts to arrive at:

$$\int_\Omega w u_t dx + \int_\Omega w a u_x dx = -\int_\Omega w_x \alpha u_x dx$$

where we have not considered the boundary terms for simplicity. If we consider

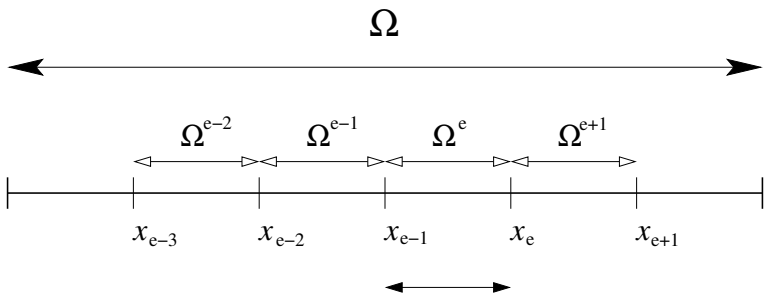


Figure 7: Equispaced mesh

an equispaced discretisation as shown in figure 7 then the weak form can be represented in matrix form as

$$\boldsymbol{M}\hat{\boldsymbol{u}}_t + a\boldsymbol{D}\hat{\boldsymbol{u}} = -\alpha\boldsymbol{L}\hat{\boldsymbol{u}} \tag{59}$$

where $\boldsymbol{M}$ is the global mass matrix, $\boldsymbol{D}$ is the derivative matrix and $\boldsymbol{L}$ is all weak Laplacian matrix. Each of these matrices can be constructed from their elemental contributions, i.e.

$$\boldsymbol{M} = \boldsymbol{\mathcal{A}}^T \underline{\boldsymbol{M}}^e \boldsymbol{\mathcal{A}}.$$

For a $P = 1$ local basis the elemental matrices are $(i, j = 0, 1)$:

$$\begin{aligned}
\boldsymbol{M}^e[i][j] &= \int_{\Omega^e} \phi_i \phi_j dx = \int_{-1}^{1} \phi_i \phi_j J^e d\xi \\
\boldsymbol{D}^e[i][j] &= \int_{\Omega^e} \phi_i \frac{d\phi_j}{dx} dx = \int_{-1}^{1} \phi_i \frac{d\phi_j}{d\xi}\frac{d\xi}{dx} J^e d\xi \\
\boldsymbol{L}^e[i][j] &= \int_{\Omega^e} \frac{d\phi_i}{dx}\frac{d\phi_j}{dx} dx = \int_{-1}^{1} \frac{d\phi_i}{d\xi}\frac{d\phi_j}{d\xi}\left(\frac{d\xi}{dx}\right)^2 J^e d\xi
\end{aligned}$$

where

$$\phi_0 = \frac{(1-\xi)}{2} \qquad \phi_1 = \frac{(1+\xi)}{2}$$

and

$$\begin{aligned} x^e &= x_{e-1}\phi_0 + x_e\phi_1 = (x_e - x_{e-1})\frac{(1+\xi)}{2} + x_{e-1} = \frac{h(1+\xi)}{2} + x_{e-1} \\ J^e &= \frac{dx^e}{d\xi} = \frac{h}{2} \qquad \frac{d\xi}{dx} = \frac{2}{h} \quad \text{and} \quad h = x_e - x_{e-1}. \end{aligned}$$

Therefore evaluating the elemental matrices and noting $\frac{\phi_0}{d\xi} = -\frac{1}{2}, \frac{\phi_1}{d\xi} = \frac{1}{2}$ we find:

$$\left.\begin{aligned} \boldsymbol{M}^e[0][0] &= \int_{-1}^{1}\left(\tfrac{(1-\xi)}{2}\right)^2 \tfrac{h}{2}d\xi = \tfrac{h}{3} \\ \boldsymbol{M}^e[0][1] &= \boldsymbol{M}^e[1][0] = \int_{-1}^{1}\tfrac{(1-\xi)}{2}\tfrac{(1+\xi)}{2}\tfrac{h}{2}d\xi = \tfrac{h}{6} \\ \boldsymbol{M}^e[1][1] &= \int_{-1}^{1}\left(\tfrac{(1+\xi)}{2}\right)^2 \tfrac{h}{2}d\xi = \tfrac{h}{3} \end{aligned}\right\} \Rightarrow \boldsymbol{M}^e = \begin{bmatrix} \frac{h}{3} & \frac{h}{6} \\ \frac{h}{6} & \frac{h}{3} \end{bmatrix}$$

$$\left.\begin{aligned} \boldsymbol{D}^e[0][0] &= \int_{-1}^{1}\tfrac{(1-\xi)}{2}\left(-\tfrac{1}{2}\right)\tfrac{2}{h}\tfrac{h}{2}d\xi = -\tfrac{1}{2} \\ \boldsymbol{D}^e[0][1] &= \int_{-1}^{1}\tfrac{(1+\xi)}{2}\left(\tfrac{1}{2}\right)\tfrac{2}{h}\tfrac{h}{2}d\xi = \tfrac{1}{2} \\ \boldsymbol{D}^e[1][0] &= \int_{-1}^{1}\tfrac{(1-\xi)}{2}\left(\tfrac{1}{2}\right)\tfrac{2}{h}\tfrac{h}{2}d\xi = -\tfrac{1}{2} \\ \boldsymbol{D}^e[1][1] &= \int_{-1}^{1}\tfrac{(1+\xi)}{2}\left(\tfrac{1}{2}\right)\tfrac{2}{h}\tfrac{h}{2}d\xi = \tfrac{1}{2} \end{aligned}\right\} \Rightarrow \boldsymbol{D}^e = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$\left.\begin{aligned} \boldsymbol{L}^e[0][0] &= \int_{-1}^{1}\left(-\tfrac{1}{2}\right)^2\left(\tfrac{2}{h}\right)^2\tfrac{h}{2}d\xi = \tfrac{1}{h} \\ \boldsymbol{L}^e[0][1] &= \boldsymbol{L}^e[1][0] = \int_{-1}^{1}\left(-\tfrac{1}{2}\right)\left(\tfrac{1}{2}\right)\left(\tfrac{2}{h}\right)^2\tfrac{h}{2}d\xi = -\tfrac{1}{h} \\ \boldsymbol{L}^e[1][1] &= \int_{-1}^{1}\left(\tfrac{1}{2}\right)^2\left(\tfrac{2}{h}\right)^2\tfrac{h}{2}d\xi = \tfrac{1}{h} \end{aligned}\right\} \Rightarrow \boldsymbol{L}^e = \begin{bmatrix} \frac{1}{h} & -\frac{1}{h} \\ -\frac{1}{h} & \frac{1}{h} \end{bmatrix}$$

These elemental matrices are then assembled into global matrices using the assembly process $\boldsymbol{\mathcal{A}}$ such that

$$\boldsymbol{M} = \boldsymbol{\mathcal{A}}^T \begin{bmatrix} \boldsymbol{M}^1 & 0 & & 0 \\ 0 & \boldsymbol{M}^2 & 0 & \\ & 0 & \ddots & 0 \\ 0 & & 0 & \boldsymbol{M}^{N_{el}} \end{bmatrix} \boldsymbol{\mathcal{A}} = \begin{bmatrix} \ddots & \ddots & 0 & 0 & 0 & 0 \\ \ddots & \ddots & \ddots & 0 & 0 & 0 \\ 0 & \frac{h}{6} & \frac{2h}{3} & \frac{h}{6} & 0 & 0 \\ 0 & 0 & \frac{h}{6} & \frac{2h}{3} & \frac{h}{6} & 0 \\ 0 & 0 & 0 & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & 0 & \ddots & \ddots \end{bmatrix}.$$

Similarly the differential matrix $\boldsymbol{D}$ produces a tridiagonal matrix which has central component of the form

$$D \rightarrow \begin{array}{ccc} -\frac{1}{2} & 0 & \frac{1}{2} \end{array}$$

and the weak Laplacian matrix has tridiagonal components of the form

$$L \rightarrow \begin{array}{ccc} -\frac{1}{h} & \frac{2}{h} & -\frac{1}{h} \end{array}$$

We shall see in section 5.3 that these diagonal components are analogous to the finite difference approximation of the same operators.

5.2 Mass lumping

If we use an explicit approximation to the time derivative in equation (59), i.e. $\frac{\partial \hat{u}}{\partial t} = \frac{\hat{u}^{n+1} - \hat{u}^n}{\Delta t}$ then we have the following matrix problems to solve

$$\boldsymbol{M}\frac{\hat{\boldsymbol{u}}^{n+1} - \hat{\boldsymbol{u}}^n}{\Delta t} + a\boldsymbol{D}\hat{\boldsymbol{u}}^n = -\alpha \boldsymbol{L}\hat{\boldsymbol{u}}^n$$

which can be written as

$$\boldsymbol{M}\frac{\hat{\boldsymbol{u}}^{n+1}}{\Delta t} = \boldsymbol{M}\frac{\hat{\boldsymbol{u}}^n}{\Delta t} - a\boldsymbol{D}\hat{\boldsymbol{u}}^n - \alpha \boldsymbol{L}\hat{\boldsymbol{u}}^n.$$

Despite the explicit nature of the temporal approximation we still end up with the full mass matrix on the left hand side. A common practice in linear finite element methods is to circumvent this problem by "lumping" all the components of the mass matrix onto the diagonal thereby producing the so-called *lumped mass matrix*. This naturally makes the mass matrix diagonal and trivial to invert thereby maintaining the explicit nature of the scheme in a finite difference sense. For the example discussed in the previous section the mass matrix would have diagonal terms of the form:

$$\boldsymbol{M}^{lumped}[i][i] = \sum_j \boldsymbol{M}[j][i] = \frac{h}{6} + \frac{2h}{3} + \frac{h}{6} = h.$$

The full mass matrix is often referred to as the *Consistent mass matrix*

5.3 Analogy with Finite Differences

If we consider the i^{th} row of equation (59) using a consistent mass matrix the equation would read:

$$\frac{h}{6}\left[\frac{\partial \hat{u}_{i-1}}{\partial t} + 4\frac{\partial \hat{u}_i}{\partial t} + \frac{\partial \hat{u}_{i+1}}{\partial t}\right] + a\frac{-\hat{u}_{i-1} + \hat{u}_{i+1}}{2} = -\alpha\frac{-\hat{u}_{i-1} + 2\hat{u}_i - \hat{u}_{i+1}}{h} \tag{60}$$

However if we consider the i^{th} row of equation (59) using a lumped mass matrix the equation would read:

$$h\frac{\partial \hat{u}_i}{\partial t} + a\frac{-\hat{u}_{i-1} + \hat{u}_{i+1}}{2} = -\alpha\frac{-\hat{u}_{i-1} + 2\hat{u}_i - \hat{u}_{i+1}}{h} \tag{61}$$

or equivalently (by dividing by h)

$$\frac{\partial \hat{u}_i}{\partial t} + a\frac{\hat{u}_{i+1} - \hat{u}_{i-1}}{2h} = \alpha\frac{\hat{u}_{i-1} - 2\hat{u}_i + \hat{u}_{i+1}}{h^2}.$$

The second term is the standard centred finite difference approximation to the first derivative on an equispaced mesh, i.e.

$$\frac{du}{dx} = \frac{u_{i+1} - u_{i-1}}{2h} + O(h^2)$$

and the third term is the centred finite difference approximation to the second derivative, i.e.

$$\frac{d^2u}{dx^2} = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + O(h^2).$$

So we see that mass lumping produces an analogous approximation to the centred finite difference mesh which is also equivalent to the standard finite volume approximation on this mesh. However if the mesh had not been equispaced then the finite element and finite difference formulations would have provided an automated way of developing a second order approximation to the first derivative. The second order derivative in the above equation is only second order accurate because of mesh symmetries and so would only be first order accurate using any of the scheme on a non-equispaced mesh.

5.4 Phase properties

The lumped mass matrix would appear to be much more advantageous in comparison to the consistent mass matrix from a implementation point of view since there is no-longer a global matrix inverse for an explicit scheme. This is especially true when considering the spatial approximation which is of a similar accuracy. However another type of error which arises in unsteady problems involved propagation or advection is the dispersion error. It is in the dispersion error that we see a significant different between the lumped and consistent mass matrix. We should also note that this approach is not so readily extended to higher order expansions.

We can derive a mathematical solution to the advection diffusion problem in a periodic region by seeking a solution of the form

$$u(x,t) = \tilde{u}(t)e^{Ikx}$$

where $I = \sqrt{-1}$. On substitution of this form in to equation (58) lead to

$$\begin{aligned} \frac{\partial \tilde{u}}{\partial t}e^{Ikx} &+ aIk\tilde{u}e^{Ikx} = -\alpha k^2 \tilde{u}e^{Ikx} \\ \frac{\partial \tilde{u}}{\partial t} &= -aIk\tilde{u} - \alpha k^2 \tilde{u} \\ \tilde{u}(t) &= Ae^{-(aIk+\alpha k^2)t} \end{aligned}$$

and so the general solution is

$$u(x,t) = Ae^{Ik(x-at)}e^{-\alpha k^2 t}.$$

Let us consider the case of pure advection where $\alpha = 0$ and so the general solution in a periodic region is

$$u(x,t) = Ae^{I(kx-\omega t)}$$

where $\omega = ka$ is the temporal wave number and is also known as the analytical dispersion relationship i.e. for a given spatial wavenumber k it tells us how long it takes for the wave to repeat in time. Now numerically we can derive a similar form for the equispaced mesh since there is a periodicity about the mesh. For a more detailed discussion see the book by Gresho and Sani.

Let us assume we can represent our numerical solution $\hat{u}_i(t)$ at node point i as

$$\hat{u}_i(t) = Ae^{I(kih-\omega^\delta t)}.$$

The difference between the numerical temporal wavenumber $w^\delta(k)$ and the analytical wavenumber $\omega(k)$ is called the dispersion error and tells us how well the numerical solution tracks different spatial frequencies k.

To calculate ω^δ we first note that

$$\hat{u}_{i+1}(t) = Ae^{I(k(i+1)h-\omega^\delta t)} = Ae^{I(kih-\omega^\delta t)}e^{Ikh} = \hat{u}_i(t)e^{Ikh}$$

and similarly

$$\hat{u}_{i-1}(t) = Ae^{I(k(i-1)h-\omega^\delta t)} = Ae^{I(kih-\omega^\delta t)}e^{-Ikh} = \hat{u}_i(t)e^{-Ikh}$$

Using these results in equation (60) when $\alpha = 0$ therefore leads to the equation:

$$\begin{aligned} \frac{h(e^{-Ikh}+4+e^{Ikh})}{6}\frac{\partial \hat{u}_i}{\partial t} + a\frac{-e^{-Ikh}+e^{Ikh}}{2}\hat{u}_i &= 0 \\ -\frac{h(e^{-Ikh}+4+e^{Ikh})}{6}I\omega^\delta \hat{u}_i + a\frac{-e^{-Ikh}+e^{Ikh}}{2}\hat{u}_i &= 0 \end{aligned}$$

which can be simplified and re-arranged to give the discrete dispersion relationship

$$\omega^\delta = \frac{a}{h}\frac{3\sin(kh)}{(2+\cos(kh))}$$

Since k is the spatial wavenumber for all possible frequencies that can be represented on all our mesh we can write $k_n = 2\pi n/L$ where L is the length of the domain. Further we note that $L = hN_{el}$ then we can define $\theta = k_n h = 2\pi n/N_{el}$ so the dispersion relation can be re-written as

$$\omega^\delta = \frac{ak}{\theta}\frac{3\sin(\theta)}{(2+\cos(\theta))}$$

For a lumped mass matrix the term $\frac{3}{(2+\cos(\theta))}$ is replaced by 1 and so the dispersion relationship becomes:

$$\omega^\delta_{lumped} = ak\frac{\sin(\theta)}{\theta}$$

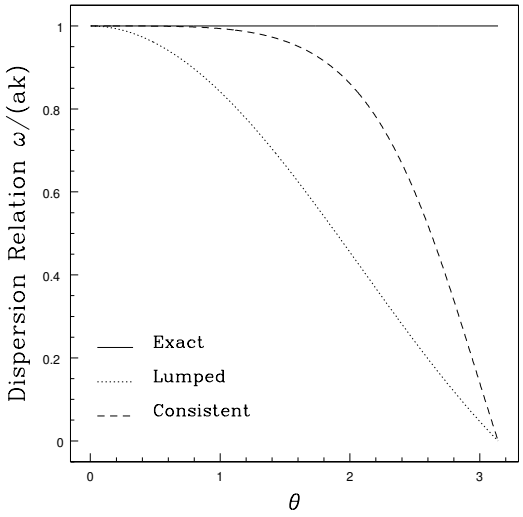


Figure 8: Phase relationship for a consistent and lumped mass matrix approximation to the linear advection equation.

So what does it all mean? Figure 8 shows the dispersion relationship of $\omega/(ak)$ versus θ. Small values of θ correspond to long spatial wavelengths and so we see that these waves propagate at the correct speed. As $\theta \to \pi$ we are considering the smallest wave which fits onto a mesh and so these waves do not propagate at the correct speed. Finally we see that the consistent mass matrix has a much better approximation to the analytic dispersion relationship than the lumped mass matrix.

5.5 Streamline Upwinded Petrov Galerkin

As a final point we shall briefly introduce Streamline Upwinded Petrov Galerkin (SUPG) which is the finite element equivalent to upwinding. If we discretise the time derivative in equation (60) or (61) using a first order explicit scheme (sometimes called Euler Forward) then the scheme is not stable. To show this requires further analysis similar to the phase analysis above however we note that a higher order integration in time would be stable. Nevertheless to overcome this instability the first order derivative is commonly *upwinded* in finite difference schemes.

An equivalent scheme in finite elements is to use a Petrov-Galerkin formulation where we use a different test solution to the trial function. The motivation here is to use a test function which is biased in the upwind direction and therefore introduce some upwinding/stabilisation into the scheme. An alternative interpretation is that the upwinding is adding artificial diffusion.

Following the book of Zienkiewicz, figure 9, the test function $w(x)$ is modified to be of the form:

$$w_i(x) = \phi_i(x) + \alpha\tilde{w}_i.$$

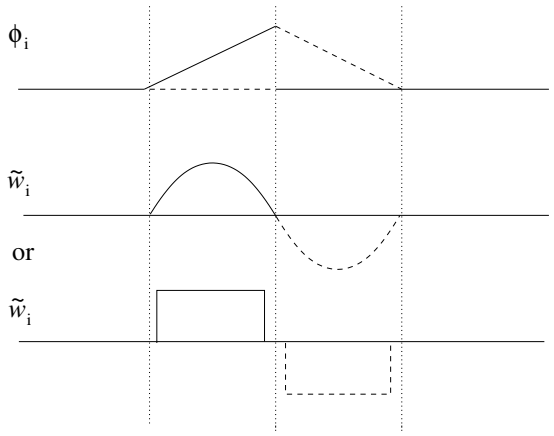


Figure 9: Linear basis function ϕ_i and two possible upwinding test functions $\tilde{w}_i$

A simple choice of the $\tilde{w}$ is

$$\tilde{w} = \frac{h}{2}\frac{d\phi_i}{dx}\ \text{sign}(a)$$

where a is the propagation velocity. We see here that the modification is dependent upon the sign of the wave propagation which introduces the upwinding. This test function is discontinuous for a linear basis $\phi_i(x)$ (see Zienkiewicz) for further discussion. However the modification can be generalised to multiple dimensions by using

$$\tilde{w} = \frac{h}{2}\frac{A_j}{|A|}\frac{d\phi_i}{dx_j}$$

where A_j is the j^{th} component of the wave velocity.

$$\tilde{w} = \frac{h}{2}\frac{A_j}{|A|}\frac{d\phi_i}{dx_j}$$

6 Spectral element/p-type expansion

Up to now we have primarily focused on the classical linear finite element or h-type methods where convergence of the discrete solution to the exact solution is achieved by reducing the size of the element, i.e. let $h \to 0$. However we do not have to restrict our attention to simply using a linear expansion within the each element. There is a class of finite element methods known as the *spectral element* or *p-type methods* which choose to use higher order polynomial expansions within each elemental region. In this approach convergence can be achieved by letting the order of the polynomial expansion p tend to infinity whilst keeping the mesh size fixed, i.e. $p \to \infty$. A combination of these concepts is therefore the *hp method* where both the mesh size and polynomial order are change to achieve convergence. In this section we will outline some of the details of this approach.

6.1 Modal and Nodal Expansions

The key historical distinction between a spectral element and a p-type finite element is whether the expansion is nodal or modal. As has been mentioned previously the Galerkin approximation is the minimising solution independent of the polynomial approach so if there are no integration errors then the methods are mathematically equivalent. However each approach does have different numerical properties in terms of efficiency of implementation, ability to vary the polynomial order and the conditioning of the global matrix systems. Traditionally p-type finite elements have been adopted in structural mechanics where as spectral elements have been adopted in fluid dynamics. Although this is not a hard and fast distinction.

To start our discussion we can consider three different expansion bases as shown in figure 10 (for even order of p):

- The first expansion set, $\Phi_p^A(x)$, simply increases the order of x and we shall refer to it as the *moment* expansion (each order contributing an extra moment to the expansion). This basis is called a *hierarchical modal* expansion because the expansion set of order P is contained within the expansion set of order $P+1$.

- The second polynomial $\Phi_p^B(x)$ is a Lagrange polynomial which is based on a series of $P+1$ nodal points x_q (see section 6.4). The Lagrange polynomial

is a non-hierarchical basis because it consists of $P+1$ polynomials of order P. This can be contrasted with the hierarchical expansion $\Phi_p^A(x)$ which consists of polynomials of increasing order. The Lagrange basis has the notable property that $\Phi_p^B(x_q) = \delta_{pq}$, where δ_{pq} represents the Kronecker delta. This property implies that for an expansion of the form

$$u^\delta(x) = \sum_{p=0}^{P} \hat{u}_p \Phi_p^B(x),$$

the expansion coefficient $\hat{u}_p$ can be defined in terms of approximate solution at the point x_p since

$$u^\delta(x_q) = \sum_{p=0}^{P} \hat{u}_p \Phi_p^B(x_q) = \sum_{p=o}^{P} \hat{u}_p \delta_{pq} = \hat{u}_q.$$

The coefficients therefore have a physical interpretation in that they represent the approximate solution at the points x_q. The points x_q are referred to as *nodes* and the Lagrange expansion basis is referred to as a *nodal* expansion. Linear finite elements are an example of a nodal expansion where the nodal points are at the ends of the domain.

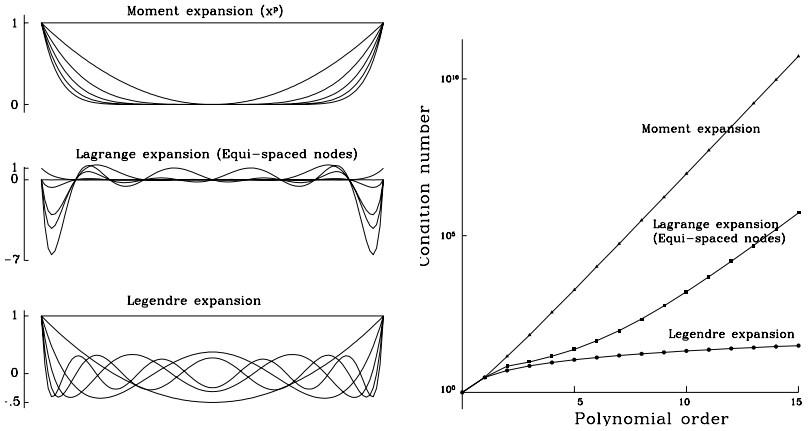
- The final expansion, $\Phi_p^c(x)$, is also a hierarchical modal expansion. However, in this case the expansion is the Legendre polynomial $L_p(x)$. By definition, this polynomial is orthogonal in the Legendre inner product

$$(L_p(x), L_q(x)) = \int_{-1}^{1} L_p(x) L_q(x) dx = \left(\frac{2}{2p+1}\right) \delta_{pq}.$$

Orthogonality has important numerical implications for the conditioning of the Galerkin method.

The choice of an expansion set is influenced by its numerical efficiency, conditioning, and the linear independence of the basis as well as its approximation properties.The conditioning of the matrix $\boldsymbol{M}$ is related to the linear independence of the expansion and is very important in the numerical inversion of matrix systems.

The condition number in the L^2 norm for the three types of expansion bases $\Phi_p^A(x)$, $\Phi_p^B(x)$, and $\Phi_p^C(x)$ is shown in figure 10b as a function of polynomial order.



(a) (b)

Figure 10: (a) Expansion modes (p even) for three expansion bases $\Phi_p^A(x)$(moment), $\Phi_p^B(x)$ (Lagrange), and $\Phi_p^C(x)$ (Legendre) of order $P = 10$ in the region $(-1 \leq x \leq 1)$. (b) Lin-log plot of the condition number of the mass matrix versus polynomial order for the bases $\Phi_p^A(x), \Phi_p^B(x)$, and $\Phi_p^C(x)$.

We see that the condition number of the mass matrix for the moment expansion grows as $\kappa_2 \propto 10^P$. Initially, the conditioning of the equi-spaced Lagrange basis is relatively good; however, after about $P \approx 5$ the condition number also starts to grow as $\kappa_2 \propto 10^P$. In contrast, the Legendre basis is very well conditioned for all values of P. The poor conditioning of the moment and Lagrange expansion reflects the fact that the basis is becoming numerically linearly dependent.

For higher p values the moment expansion only contributes in the region $x = \pm 1$ and so $\Phi_p^A(x) \approx \Phi_{p+2}^A(x)$. In the equispaced Lagrange expansion oscillations appear at the ends of the domain which heavily biases this region, thereby making the modes nearly linearly dependent. The Legendre modes tend to cover the region in a relatively uniform manner although they do bias the ends of the domain more than the interior.

6.2 Boundary Interior Decomposition of p-Type Modes

From the previous discussion we might presume that the "best" choice for our expansion set is the Legendre polynomial or more generally a system of orthogonal polynomials. This is true in so far as the hierarchy and orthogonality tend to lead to well conditioned matrices. However, we also want to combine the expansion with the h-type elemental decomposition. The difficulty arises when we try to ensure a degree of continuity in the global expansion at elemental boundaries. Typically in the finite element methods this is satisfied by imposing a C^0 continuity between elemental regions, that is, the global expansion modes are continuous everywhere in the solution domain although the derivatives may not be.

A numerically efficient way of achieving this is to design an expansion where only some modes have a magnitude at an elemental boundary then the condition can be imposed far more easily. This type of decomposition is known as *boundary* and *interior* decomposition. Boundary modes have magnitude at one of the elemental boundaries and are zero at all other boundaries. Interior modes, sometimes known as *bubble* modes, only have magnitude in the interior of the element and are zero along all boundaries.

6.3 Modal p-Type Expansion

It is advantageous to consider orthogonal polynomials when constructing p-type expansions. The most commonly used modal p-type elemental expansions are based upon the orthogonal set of polynomials called the Jacobi polynomials. In the standard interval $\Omega_{st} = \{\xi \mid -1 < \xi < 1\}$ we denote by $\psi_p(\xi)$ the p-type modal expansion defined as:

$$\phi_p(\xi) \mapsto \psi_p(\xi) = \begin{cases} \left(\frac{1-\xi}{2}\right) & p = 0 \\ \left(\frac{1-\xi}{2}\right)\left(\frac{1+\xi}{2}\right) P_{p-1}^{1,1}(\xi) & 0 < p < P \\ \left(\frac{1+\xi}{2}\right) & p = P. \end{cases} \tag{62}$$

Note that $\phi_p(\xi)$ will be used to denote a general definition of a local polynomial basis whereas $\psi(\xi)$ has the specific definition given above. The shape of all modes for $P = 5$, normalized to have a maximum value of one, is shown in figure 11. The lowest expansion modes $\psi_0(x)$ and $\psi_P(x)$ are the the same as the linear finite element expansion. These are boundary modes since they are the only modes

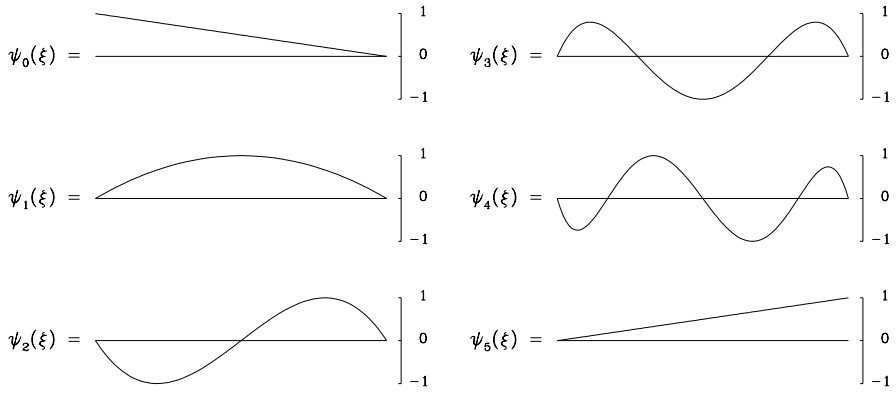


Figure 11: Shape of modal expansion modes for a polynomial order of $P = 5$.

which have magnitude at the ends of the interval. The remaining interior modes, by definition, are zero at the ends of the interval and increase in polynomial order as is typical in a hierarchical expansion. Clearly, the only choice for the quadratic mode, $\psi_1(x)$, is the shape $\left(\frac{1-\xi}{2}\right)\left(\frac{1+\xi}{2}\right)$ and this is the usual hierarchical expansion for quadratic elements.

The shape of interior modes could be defined as any polynomial which satisfies the end conditions, however, using the Jacobi polynomial $P_{p-1}^{1,1}(\xi)$ maintains a high degree of orthogonality and therefore better conditioning.

6.4 P Element: Nodal - Spectral Elements

Polynomial nodal expansions are based upon the Lagrange polynomials which are associated with a set of nodal points. The nodal points must include the ends of the domain if the expansion is to be decomposed into boundary and interior modes. Apart from this restriction we are free to choose the location of the interior nodal points. The choice of these points, however, plays an important role in the stability of the approximation and the conditioning of the system. Using nodal points at the zeros of the Gauss-Legendre-Lobatto integration rule (see section 4.2) produces a particularly efficient expansion which does not exhibit the oscillations seen when equi-spaced points are chosen.

The shape of the modes for an expansion with $P = 5$ is shown in figure 12.

Unlike the modal p-type expansion shown in figure 11, all modes are polynomials of order P. The boundary modes are $h_0(\xi)$ and $h_5(\xi)$. As seen in figure 10 the equispaced Lagrange polynomial oscillates at the ends of the domains. The Lagrange polynomial through the Gauss-Lobatto-Legendre points (see section 4.2) does not exhibit this type of oscillation.

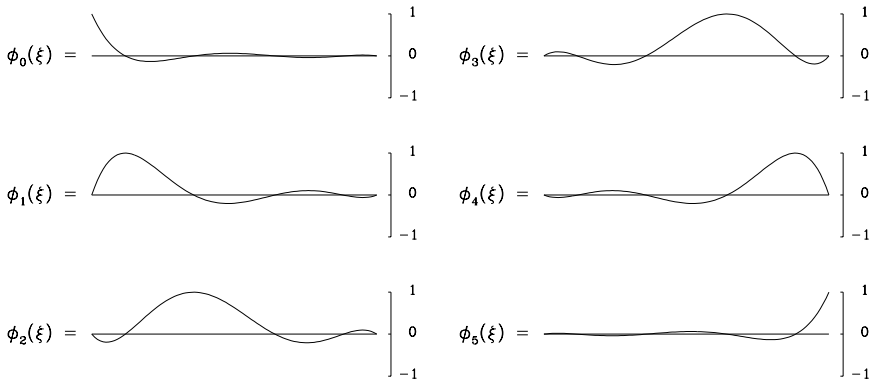


Figure 12: Nodal expansion modes for a polynomial order of $P = 5$.

6.5 What is the advantage

For the the one-dimensional Helmholtz problem and assuming a discretization on a uniform mesh of equispaced elements of size h, the general error estimate in the energy norm for the p- and h-type extension process can be written as

$$||\varepsilon||_E \leq Ch^{\mu-1}P^{-(k-1)}||u||_k,$$

where $\varepsilon = u - u^\delta$, $\mu = \min(k, P+1)$ and C is independent of h, P and u, but depends on k.

Clearly, if the solution is smooth enough to have bounded derivatives such that $k \geq P+1$ then this error estimate shows us that we can achieve exponential convergence as we increase the polynomial order P (p-type extension).

Starting with the prototype problem

$$\nabla^2 u(x) - \lambda u(x) = f(x), \quad \lambda \geq 0, \tag{63}$$

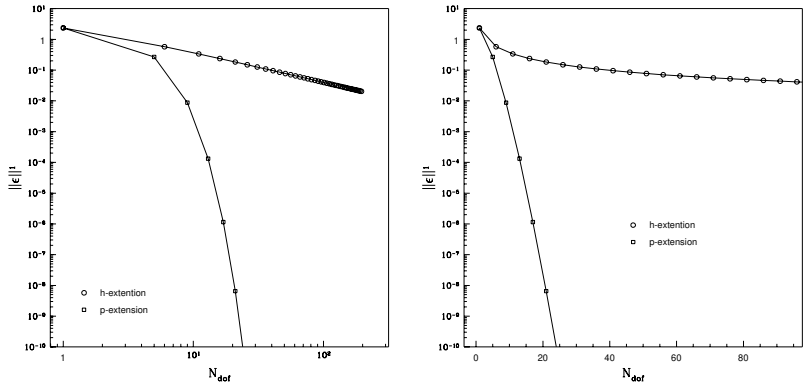


Figure 13: Convergence in the discrete energy norm $||\epsilon||_E$ as a function of degrees of freedom N_{dof}. Two tests were performed using the h-type extension with a fixed polynomial order $p = 1$ and the p-type extension with two elemental domains. (a) Error on a Log-Log axis demonstrating the algebraic convergence of the h-type extension. (b) Error on a semi-Log axis demonstrating the exponential convergence of the p-type extension for smooth solutions.

we consider a solution which is infinitely differentiable having the form $u(x) = \sin(\pi x)$ with $\lambda = 1$.

Since we know the solution is of the form $eu(x) = \sin(\pi x)$ we also know that $f(x) = -(\pi^2 + \lambda)\sin(\pi x)$. The numerical solution $u^\delta(x)$ to this problems using both h- and p-type extensions in an interval $x = [-1, 1]$ with Dirichlet boundary conditions is shown in figure 13. The error is measured in the discrete energy (or equivalently H^1) norm and is normalized by the length of the domain l, that is,

$$||\epsilon||_E = \frac{||u - u^\delta(x)||_E}{||1||^1} = \frac{||u - u^\delta(x)||_E}{\sqrt{l}}.$$

The h-type convergence tests were performed using a polynomial order $P = 1$ and the p-type convergence tests were performed using two elements.

In the first plot of figure 13 we see the error plotted on a Log-Log axis as a function of the total degrees of freedom N_{dof}. As shown in section 4.5 if the second derivative of the solution can be bounded by a constant ($|u''| < C$), then

the error in energy norm of the h-extension process is

$$||\varepsilon||_E \leq K_1 C h.$$

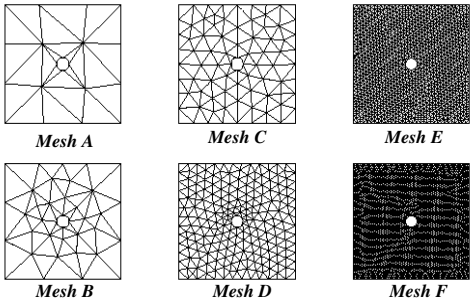
Since $h \approx 1/N_{dof}$ when P is fixed we expect the error to behave as a linear function of N_{dof} on the Log-Log plot. From the results given in section 6.5 we see that the slope (or convergence rate) of the h-type extension process is related to the minimum of the polynomial order P plus one and the smoothness of the solution. Since the solution $u(x) = \sin(\pi x)$ is an infinitely differentiable function the polynomial order dictates the convergence rate. Also shown in figure 13(a) is the p-type extension process for a domain containing two elements. The smooth property of the solution means that an exponential rate of convergence is achieved in terms of the polynomial order. Since the number of elements is fixed then $N_{dof} \approx P$, and so if we plot the error on a semi-Log axis as shown in figure 13(b) we observe the exponential decay in the error due to the p-type extension process.

In figure 14 we see a comparison between standard finite element and spectral/hp finite element approaches for the test case of potential flow around a circular cylinder where the potential is given as $\phi(r,\theta) = U_0(r + a^2/r)cos(\theta)$. The standard finite element converges as the characteristic size of each element in the mesh, denoted by h, is reduced. Therefore a series of six mesh refinements was considered as shown in figure 14(a). In the spectral/ hp finite element approach, a fixed resolution mesh was considered and convergence was achieved by increasing the order of the polynomial expansion within each element.

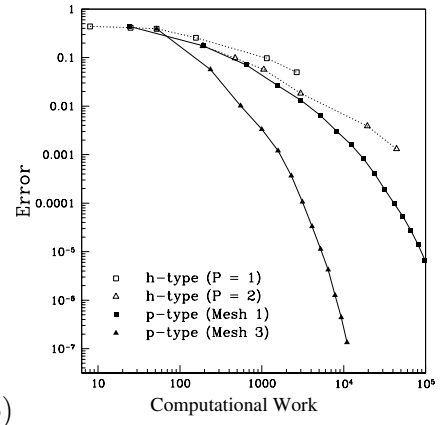
The computational cost per degree of freedom in the spectral/hp element method is proportional to $N_{el}\ P^{DIM+1}$ where N_{el} is the number of elemental subdomains and DIM is the spatial dimension of the approximation. Although the computational cost per piece of information is higher using the spectral/hp element method for smooth solutions the rate of convergence is exponential in the polynomial order. The error per unit of computational work is accordingly less. This point is illustrated in figure 14(b) which shows the error in the solution as a function of computational work. The dotted lines show the error per unit of computational work for the standard finite element approach (h-type convergence) using a first and second order polynomial approximation. For these cases we have considered computational work to scale as the number of degrees of freedom. The solid lines show the error per unit of computational work when the higher polynomials order is increased on fixed meshes A and C in figure

14(b) (p-type convergence). For these cases we have scaled the computational work as $N_{el}\ P^3$. We see that for larger errors it is impossible to distinguish between the two approaches since the exponential convergence is only observed when the solution is captured spatially. However if we require a lower error then the spectral/hp element method can achieves this requirement at a lower computational cost as the exponential rate of convergence is realised.

These methods provide an efficient algorithm to perform accurate time dependent simulations and have been used to solve the unsteady Navier-Stokes equations with great success, particularly in the study of fundamental fluid dynamics through the use of direct numerical simulations.



(a)



(b)

Figure 14: A model problem of potential flow around a circular cylinder was considered on a series of six meshes as shown in (a). The error in the solution as a function of idealised computational cost for the standard finite element approach (dotted lines) and the spectral/hp finite element approach (solid lines) is shown in figure (b).

Index