IMPERIAL

Spirals, defects, rolls and bands; Transitional Rayleigh-Bénard Poiseuille flows using spectral/*hp* element methods

Imperial College London
Department of Aeronautics

Supervisor:
Professor Spencer J. Sherwin

Author:
Chi Hin Chan

Co-Supervisor
Professor Yongyun Hwang

April 2025
Ph.D. Thesis

Abstract

The transitional regimes of Rayleigh-Bénard Poiseuille (RBP) flows and Rayleigh-Bénard convection (RBC) are investigated using direct numerical simulations and linear stability analysis. RBP flows serve as a paradigmatic configuration that describes fluid motion driven by shear and buoyancy forces, a combination of the classical buoyancy-driven RBC and shear-driven plane Poiseuille flow (PPF). While the transitional regime of RBC and PPF have been well studied over the past century, the transitional regime where both forces interact remains largely unexplored beyond linear instability, investigated in the first part of this thesis.

Following a review of the relevant literature and numerical methods, we conduct direct numerical simulations using Nektar++, a spectral/*hp* element package. The simulations span over a range of Rayleigh numbers, $Ra \in [0, 10000]$ and Reynolds number $Re \in [0, 2000]$, with unit Prandtl number in a large computational domain. Within this parametric space, we identify five distinct regimes: (1) bistable SDC & ISRs, (2) ISRs, (3) wavy rolls, (4) intermittent rolls, and (5) shear-driven turbulence. The intermittent rolls regimes represent a newly identified state characterised by the spatio-temporal breakdown and regeneration of longitudinal rolls. In the transitional shear-driven turbulent regime, we also observe that longitudinal rolls may coexist with turbulent-laminar bands. The spatio-temporal dynamics of longitudinal rolls highlight its dominant role in transitional RBP flows. We examine the unstable manifolds of the longitudinal rolls in a confined domain, integrating along which leads to turbulence. Depending on Re, this turbulence may occur transiently, decaying towards the unstable laminar base state where the longitudinal rolls can be excited again, forming a cyclic process referred to as the *thermally-assisted sustaining process (TASP)*. We present a state space sketch of the dynamical process, and discuss the relevance of the *TASP* to larger domains.

In the second part of the thesis, we explore the state space structure of the bistable system between spiral defect chaos (SDC) and ideal straight rolls (ISRs) of Rayleigh-Bénard convection within large domains. By systematically reducing the domain size, we observe that SDC occurs transiently, eventually stabilising into multiple stable invariant solutions, referred to as *elementary* states. These *elementary* states are visually and statistically similar to the localised features of SDC. We also examined the edge between the basin of attractions of ISRs and the elementary states, revealing multiple edge states. The unstable invariant manifolds of ISRs exhibit two distinct behaviours: unstable ISRs near the Busse balloon connect to stable ISRs and base state via heteroclinic connections, while unstable ISRs further from the Busse balloon lead to SDC, indicating such ISRs lie on boundary between ISRs and SDC. We conclude by presenting a state space sketch highlighting the organisation of SDC, ISRs, elementary and edge states.

Acknowledgements

There are many people who have shaped my experience as a PhD student whom I would like to express my gratitude for. First and foremost, I want to express my deepest gratitude to my supervisors, Spencer and Yongyun. I have learned and grown a lot under their tremedously kind and patient supervision, for allowing me to explore research ideas in shark skins and hydrodynamic instabilities, to contributing to the large codebase of Nektar++, and at times canning ideas that did not work! (Hint: the topic that did not make it to this thesis). I will miss the our constant weekly catch-ups and engaging deep-dives. Apart from the PhD, they also openly supported personal development opportunities, allowing me to pursue interests such as the Nektar++ redesign project, engaging in graduate teaching assistant roles, pursuing a summer research exchange at LadHyX and an internship at IBM Research, significantly enhancing the PhD experience which I am deeply grateful for.

I would like to thank Yongyun, who also happens to be my MSc thesis supervisor, for introducing the topic of reduced-order modelling of turbulent Couette flows as part of my MSc thesis. Perhaps more importantly, for giving me the confidence I needed to pursue a PhD program at Imperial.

I would also like to thank Lutz, for his kind hospitality, ensuring that I was comfortable at LadHyX in the summer of 2024. Despite being a short three months stint, I learned about advanced modal decomposition techniques and was exposed to the broad fluid mechanics research activities at LadHyX.

I would also like to thank Basman, my Bachelor's thesis supervisor at Nanyang Technological University (NTU), during which we worked on humpback whale leading-edge turbercles. I believe it is fair to say that he was partly responsible for planting the "seed" for my fascination for computational fluid dynamics.

I would like to also thank Eloisa and Geeth, my project supervisor and manager at IBM Research, for giving the opportunity to experience, learn and contribute to AI research activities within a corporate company.

I would also like to thank the Nektar++ community, particularly Chris, Jacques and Dave for their patient guidance on how to do contribute to Nektar++ properly.

I would also like to thank my collaborators, Mohammad and Raj, for providing guidance and advice during the PhD and also providing the opportunity to work on physical informed neural networks.

The PhD experience isn't complete without the camaraderie between friends which include, unwinding over a cheeky pint on Friday evenings, Mario karting in the common room, going to Sagar and exploiting its 50% discount, honest burgers student discounts, going to Wasabi at South Kensington at closing to get half-priced bentos, countless muffin and coffee breaks, peaceful walks around campus, Hyde park runs, booking the swimming pool sessions just to use the sauna, many

gym/swim/badminton/climbing sessions, appearing decently surprise to bump into each other in the office on the weekends, and bantering over the weather, reviewers and PhD life. I will forever cherish these unique experiences together with (in no particular order) Henrik, Parv, Alex, Kaloyan, Ganlin, Jing Tian, Steffi, Lidia, Christian, Kazuki, Priyam, Sid, Elise, Yu Han, Francisco, Cheng Wei, Filippo, Bo Han, Pat, Grace, Benita, Zhao, Zilin, Victor, Mohsen, Guglielmo, João and Yacine (Sorry if I missed out your name).

The person who probably deserves the most credit would be my fiancée, Angelica, who have been there atthe beginning, ready to lend a listening ear to my monologue on how cool CFD is. The distance between Singapore and London over the years have been tough, and I am happy that we will be closing the gap and I am looking forward to our shared future. Last but not least, I am grateful to my loving parents and family, for supporting my academic interests aboard despite the distance.

This thesis is dedicated to everyone who have impacted my life.

Contents

1 Numerical Methods **6**

1.1 Method of weighted residuals 6

1.2 Galerkin Projection 8

1.3 Spectral/*hp* element method 10

1.3.1 Domain partition 10

1.3.2 Standard Elements 10

1.3.3 Global assembly 12

1.3.4 Local basis expansions 14

1.3.5 Gaussian quadrature 17

1.3.6 Numerical differentiation 18

1.3.7 Example in 1D 19

1.3.8 Error properties 21

1.4 Numerical techniques for solving the Navier-Stokes equations 22

1.4.1 Velocity Correction Scheme 22

1.4.2 Fourier spectral/*hp* modes 25

1.4.3 Maintaining fluid flow through a channel 26

1.5 Stability analysis of the Navier-Stokes equations 30

1.5.1 Algorithms for linear stability analysis 30

1.5.2 Edge tracking 34

Chapter 1

Numerical Methods

In this chapter, we will discuss the fundamentals of numerical methods relevant for solving the Navier-Stokes equations. We begin the discussion of the method weighted of residuals (§1.1) and the spatial discretisation using spectral/*hp* element methods in one dimension (§1.3). This is followed by techniques for solving the Navier-Stokes equations (§1.4), such as the velocity-correction scheme, enforcing a constant flow rate and the quasi-3D approach for semi-homogeneous domains. This chapter concludes with numerical techniques for performing stability analysis of the Navier-Stokes equations (§1.5), including eigenvalue computation and edge tracking.

1.1 Method of weighted residuals

Spatial discretisation errors, or residuals, arises as one seeks an approximate solution to some partial differential equation (PDE). The method of weighted residuals provides a generic mathematical framework in which constraints on the residual could be applied flexibly, defining the spatial discretisation scheme and its convergence properties. We approximate the solution of PDE by considering a finite expansion of a suitable basis, to which its coefficients are sought after by minimising the inner product between the PDE and a test (or weight) function. To demonstrate this, we consider a linear partial differential equation as,

$$\mathbf{L}[u(x)] = 0, \quad x \in \Omega, \tag{1.1}$$

where $\mathbf{L}$ refers to a linear spatial differential operator subjected to some boundary conditions within the domain, Ω, while $u(x)$ refers to the exact solution of $\mathbf{L}$. Examples of PDEs with linear spatial differential operators include the Laplace equation, $\nabla^2 u = 0$, Poisson equation, $\nabla^2 u - f = 0$, and the Helmholtz equation, $\nabla^2 u - \lambda u + f = 0$. We suppose that the exact solution $u(x)$ can be approximated (discretised) by N finite number of global basis (or expansion) functions, $\Phi(x)$.

$$u(x) \approx u^\delta(x) = \sum_{i=0}^{N-1} \hat{u}_i \Phi_i(x), \tag{1.2}$$

where $u^\delta(x)$ refers to the approximate solution of $u(x)$, consisting of a linear combination of the product between the i^{th} basis coefficient, $\hat{u}_i$, and the i^{th} global basis expansion, $\Phi_i(x)$, defined within

Ω. Since $u^{\delta}(x)$ is an approximate solution of equation (1.5), we expect a residual (or error) between the exact solution, $u(x)$, and $u^{\delta}(x)$,

$$\mathbf{L}[u^{\delta}(x)] = R[u^{\delta}(x)], \tag{1.3}$$

where $R[u^{\delta}(x)]$ refers to the residual which depends on the approximate solution $u^{\delta}(x)$ and varies within Ω. In other words, equation (1.3) might not be satisfied everywhere in Ω. Next, we need to place restrictions on the residual, such that it the residual approaches zero, $R \to 0$, and the approximate solution approaches the exact solution, $u^{\delta}(x) \to u(x)$. The method of weighted residuals places a restriction on the residual by applying an inner product between the governing equation, and N test (or weight) functions, $v_j(x)$, and setting it to zero,

$$(v_j(x), R[u^{\delta}(x)]) = 0, \quad j = 0, ..., N-1. \tag{1.4}$$

Definition 1.1.1 (Inner product)**.** The inner product between two functions $f(x)$ and $g(x)$ is,

$$(f, g) = \int_{\Omega} f(x)g(x)dx.$$

By enforcing equation (1.4), the discrete problem becomes a system of N ordinary differential equations to solve for the N basis coefficients, $\hat{u}_i$. The choice of test function defines the projection methods, and examples of different projection methods are shown in table 1.1. We emphasise that the method of weighted residuals only describes the projection method, but does not specify the type of basis expansions, as we will discuss later in §1.3. The choice of projection method coupled with suitable basis expansions will have different solution convergence properties. Of particular interest is on how quickly the residual vanishes as the number of basis expansions increases. For instance, by considering the Galerkin method coupled with Fourier expansions, one can expect exponential convergence for a sufficiently smooth problem which is desirable for an efficient representation of turbulent dynamics.

Weight functions	Projection method
$v_j(x) = \delta(x - x_j)$	Collocation
$v_j(x) = \begin{cases} 1 & \text{if } x \in \Omega_j \\ 0 & \text{if } x \notin \Omega_j \end{cases}$	Finite-Volume
$v_j(x) = \phi_j$	Galerkin
$v_j(x) = \frac{\partial R}{\partial \hat{u}_j}$	Least-squares

Table 1.1: Examples of weight functions and projection methods

1.2 Galerkin Projection

Galerkin projection remains as a standard projection method in the context of the finite element methods, where the test functions, $v(x)$, are chosen to be lie in the same functional space as the global basis functions, $\Phi(x)$. To demostrate the Galerkin projection method, we consider that the differential operator earlier in equation (1.1) as a 1D Helmholtz equation,

$$\mathbf{L}[u(x)] \equiv \frac{\partial^2 u(x)}{\partial x^2} - \lambda u(x) + f(x) = 0, \quad x \in \Omega := [0, l] \tag{1.5a}$$

$$u(0) = g_D, \quad \left.\frac{\partial u}{\partial x}\right|_{x=l} = g_N. \tag{1.5b}$$

where λ is a real positive constant, $f(x)$ is a forcing function, and Ω refers to the spatial domain bounded between 0 and l. To ensure that problem is well posed, Dirichlet and Neumann boundary conditions, g_D and g_N, are imposed at $x = 0$ and $x = l$ respectively. Equation 1.5 is commonly referred to as the strong or classical form.

The subsequent step is to take the inner product of the equation (1.5) with a test function, $v(x)$, that satisfies the homogeneous Dirichlet boundary conditions by definition, i.e. $v(0) = 0$, and setting the inner product to zero,

$$(v(x), \mathbf{L}[u(x)]) = \int_0^l v(x) \left[\frac{\partial^2 u(x)}{\partial x^2} - \lambda u(x) + f(x)\right] \mathrm{d}x = 0, \tag{1.6}$$

akin to applying the method of weighted residuals (§1.1). Next, we perform integration by parts,

$$\underbrace{\int_0^l \frac{\partial v}{\partial x}\frac{\partial u}{\partial x}\,\mathrm{d}x + \int_0^l \lambda v u\,\mathrm{d}x}_{a(v,u)} = \underbrace{\int_0^l v f\,\mathrm{d}x + \left[v\frac{\partial u}{\partial x}\right]_0^l}_{f(v)}. \tag{1.7a}$$

$$a(v, u) = f(v). \tag{1.7b}$$

Equation (1.7a) is typically referred to as the weak [1] form of equation (1.5). The weak form can be represented in compact notation in equation (1.7b), where $a(v, u)$ is bilinear and symmetric. In structural mechanics, $a(u, u)$ is commonly referred to as the *strain* energy. To ensure that the *strain* energy is finite, we restrict the choice of solutions $u(x)$ to lie in the solution space,

$$\mathcal{U} := \{u \,|\, u \in H^1(\Omega), u(0) = g_D\}, \tag{1.8}$$

where $u \in H^1$ refers to functions of u belonging to Sobolev space of order 1, and satistfying the Dirichlet condition, $u(0) = g_D$, at $x = 0$.

[1] The notions of the *weak* and *strong* forms refer to the smoothness (regularity) required of admissible solutions. In the weak formulation, the highest derivative involved is up to first-order, so the solution space is H^1. This space is generally larger than that of the strong formulation, which required $u \in \mathcal{H}^2(\Omega)$. Since $H^2(\Omega) \subset H^1(\Omega)$ the weak formulation imposesd a 'less stringent' constraint of the solution space of admissible functions.

Definition 1.2.1 (Sobolev space)**.** We define Sobolev space of order $n \geq 1$ on Ω,

$$H^n(\Omega) = \{u \,|\, u \in L_2(\Omega), D^\alpha u \in L_2(\Omega), \forall\alpha : \alpha \leq n\},$$

where $D^\alpha u$ refers to derivatives up to order α and $L_2(\Omega)$ refers to functions that are square integrable.

Definition 1.2.2 (L_2 space)**.** The space $L_2(\Omega)$ refers to functions that are square integrable,

$$(u, u)_{L_2} = \int_\Omega |u(x)|^2 \, \mathrm{d}\Omega < \infty. \tag{1.9}$$

In other words, H^1 functions satisfy the condition that the integral of the square of the function and its first derivative is finite, consistent with the highest order derivative in the weak formulation of equation (1.6). Similarly, the space of test functions, $\mathcal{V}$, is defined as,

$$\mathcal{V} := \{v \,|\, v \in H^1, v(0) = 0\}, \tag{1.10}$$

where $v \in H^1$ ~~are~~ refer to test functions belonging to the Sobolev space of order 1, and is defined to be zero, $v(0) = 0$ on Dirichlet boundary condition, $x = 0$. The generalised weak form is finding $u(x) \in \mathcal{U}$, such that

$$a(v, u) = f(v), \quad \forall v \in \mathcal{V}. \tag{1.11}$$

At this point, function spaces $\mathcal{U}$ and $\mathcal{V}$, contain infinitely many possible functions and equation (1.11) is therefore infinite dimension. To obtain an approximate solution, $u^\delta(x)$, we restrict ourselves to finite dimensional subspaces, $\mathcal{U}^\delta \subset \mathcal{U}$, and $\mathcal{V}^\delta \subset \mathcal{V}$. Then, the problem is to find $u^\delta \in \mathcal{U}^\delta$, such that

$$a(v^\delta, u^\delta) = f(v^\delta), \quad v^\delta \in \mathcal{V}^\delta. \tag{1.12}$$

The subspaces $u^\delta \in \mathcal{U}^\delta$ and $v^\delta \in \mathcal{V}^\delta$ are not the same (compare the Dirichlet boundary conditions of equations (1.8) and (1.10)), necessary for the standard Galerkin projection procedure where they should lie in the same subspace. To ensure that they belong to the same space, we lift the solution u^δ into two parts,

$$u^\delta = u^\mathcal{H} + u^\mathcal{D}. \tag{1.13}$$

where $u^\mathcal{H} \in \mathcal{V}^\delta$ satisfies the homogeneous Dirichlet condition (e.g. is zero on Dirichlet boundaries), belonging to the same subsape as $v^\delta \in \mathcal{V}^\delta$, while $u^\mathcal{D} \in \mathcal{U}^\delta$ satisfies the Dirichlet boundary conditions $u^\mathcal{D}(0) = g_D$. Finally, the standard Galerkin projection method is to solve for $u^\mathcal{H} \in \mathcal{V}^\delta$,

$$a(v^\delta, u^\mathcal{H}) = f(v^\delta) - a(v^\delta, u^\mathcal{D}). \tag{1.14}$$

This concludes the classical Galerkin formulation, where we have converted a continuous PDE into a discrete problem amenable to standard direct or iterative linear solvers. Under certain assumptions of a, a solution is guaranteed under the Lax-Milgram theorem [Lax and Milgram, 1955]. The order in which we perform integration by parts followed by defining the finite solution space is crucial. Reversing this order can introduce jumps in the solution derivatives between the element boundaries.

1.3 Spectral/*hp* element method

While the procedure for approximating a solution of a PDE using the classical Galerkin projection technique has been described, the spatial discretisation scheme, related to the choice of basis (and test) functions, remains undiscussed.

In this section, we introduce the spectral/*hp* element method [Patera, 1984], a spatial discretisation scheme in which the solution domain is partitioned into a set of non-overlapping finite elements of size h, within which the solution represented as a linear combination of continuous orthogonal polynomial functions up to order P. This approach combines the geometric flexibility of classical finite-element methods [Strang and Fix, 2008], which allow complex geometries to be represented, with the exponential convergence properties of spectral methods [Gottlieb and Orszag, 1977]. For the reasons above, the spectral/*hp* element method provides an attractive framework for approximating PDE solutions at a given cost, as we shall see later. This section is organised as follows: domain partition, standard elements, assembly process, modal and nodal expansion functions, numerical integration and differentiation, a 1D example before concluding with error properties of the spectral/*hp* element method.

1.3.1 Domain partition

The first step concerns the partitioning the domain into a set of finite elemental regions. We consider an example in one dimension within Ω, and partition it into a set of N_{el} elements, where Ω^e, refers to the elemental partitions with $1 \geq e \geq N_{el}$, such that they meet at their boundaries and do not overlap,

$$\Omega = \bigcup_{e=1}^{N_{el}} \Omega^e, \quad \text{where} \bigcap_{e=1}^{N_{el}} \Omega^e = \emptyset \tag{1.15}$$

where the e^{th} element is defined as,

$$\Omega^e = \{x \,|\, x_{e-1} \geq x \geq x_e\}. \tag{1.16}$$

Each element can be represented by a linear combination of orthogonal basis expansions. The basis expansions can be either modal or nodal expansions, as we shall see later.

1.3.2 Standard Elements

In general, we expect to work with non-uniform elements that may have arbitrarily shapes, making the definition of basis expansions potentially unwieldy. To simplify the formulation, it is convenient to define a *standard* element,

$$\Omega_{st} = \{\xi \,|\, -1 \geq \xi \geq 1\}, \tag{1.17}$$

where Ω_{st} refers to the standard element defined in local coordinates, $\xi \in [-1, 1]$. Within this standard element, the formulation of basis expansions, as well as differential and integration operations, can be carried out in the local coordinate system ξ, before mapping the solution back to the global domain,

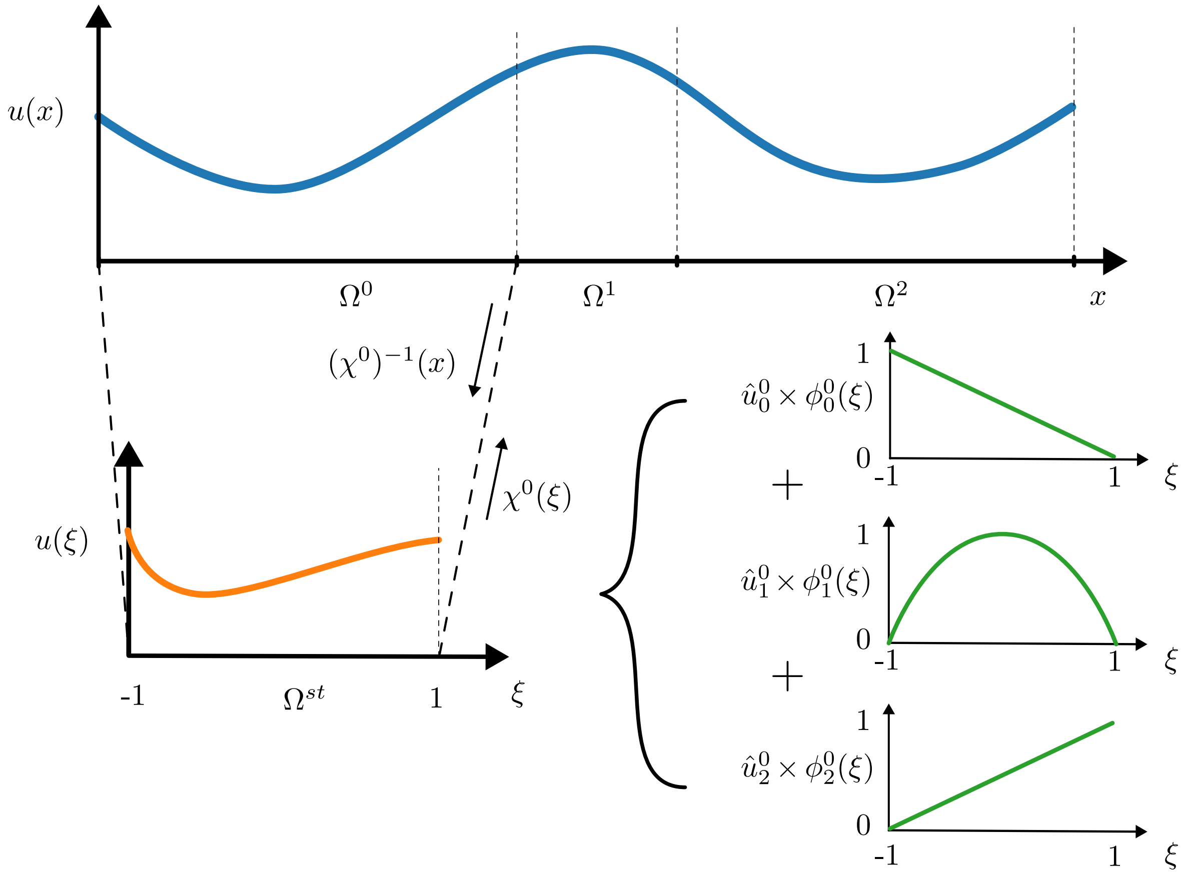

Figure 1.1: A spectral/*hp* element representation of $u(x)$, consisting of three non-overlapping finite elements, each containing a linear combination of local expansion bases of up to $P = 2$.

x. We can map the standard element into any arbitrary global coordinates based on a linear mapping $\chi^e : \Omega_{st} \to \Omega$,

$$x = \chi^e(\xi) = \frac{1-\xi}{2}x_e + \frac{1+\xi}{2}x_{e+1}, \quad \xi \in \Omega_{st} \tag{1.18}$$

which for this case has an analytical inverse, $(\chi^e)^{-1}(x)$,

$$\xi = (\chi^e)^{-1}(x) = 2\frac{x - x_{e-1}}{x_e - x_{e-1}} - 1, \quad x \in \Omega^e. \tag{1.19}$$

For illustration purposes, we consider that the standard element can be represented by three local basis expansions of polynomial order of up to $P = 2$,

$$\phi^e_0(\xi) = \frac{1-\xi}{2}, \quad \phi^e_1(\xi) = (1+\xi)(1-\xi), \quad \phi^e_2(\xi) = \frac{1+\xi}{2}, \tag{1.20}$$

where ϕ^e_0, ϕ^e_2 and ϕ^e_1 refers to the linear and quadratic local basis expansions of the e^{th} element. These local basis expansions is illustrated in figure 1.1. The approximate solution is now represented as,

$$u^\delta(x) = \sum_{e=1}^{N_{el}} \sum_{i=0}^{P} \hat{u}^e_i \phi^e_i(\xi). \tag{1.21}$$

where $\hat{u}_i^e$ refers to the local expansion basis coefficients. The approximate solution, $u^\delta(x)$, now lie within the solution space $\mathcal{U}^\delta$ defined as,

$$\mathcal{U}^\delta := \left\{u^\delta \,\middle|\, u^\delta \in H^1,\ u^\delta(\chi^e(\xi)) \in \phi_i^e(\xi), \forall i : 0 \le i \le P, \forall e : 1 \le e \le N_{el}\right\} \tag{1.22}$$

We note that the local basis expansion shown here are not strictly orthogonal polynomials. By modifying them with Jacobi polynomials, they may become orthogonal with increasing P as we see later.

1.3.3 Global assembly

In this section, we introduce the concept of global assembly (or direct stiffness summation) which relates the global basis expansions (equation (1.2)), $\Phi_i(x)$, to the local basis expansions (equation (1.21)), $\phi_i^e(x)$, where the solution can be approximated using either formulation,

$$u^\delta(x) = \sum_{i=0}^{N-1} \hat{u}_i \Phi_i(x) = \sum_{e=1}^{N_{el}} \sum_{i=0}^{P} \hat{u}_i^e \phi_i^e(\chi^e(\xi)). \tag{1.23}$$

In general, we can represent the global and local basis coefficients each as a column vector,

$$\hat{\mathbf{u}}_g = \begin{pmatrix} \hat{u}_0 \\ \vdots \\ \hat{u}_N \end{pmatrix}, \quad \hat{\mathbf{u}}_l = \begin{pmatrix} \hat{\mathbf{u}}^0 \\ \vdots \\ \hat{\mathbf{u}}^{N_{el}-1} \end{pmatrix}, \tag{1.24}$$

where $\hat{\mathbf{u}}^e = (\hat{u}_0^e, ..., \hat{u}_P^e)^T$, $\hat{\mathbf{u}}_g \in \mathbb{R}^N$, $\hat{\mathbf{u}}_l \in \mathbb{R}^{N_{loc}}$ and $N_{loc} = N_{el}(P+1)$. As there can be more local degrees of freedom than global degrees of freedom, $N_{loc} > N$, we need to impose some conditions on the local expansion coefficients. One of the common approach is to enforce C^0 continuity across elemental boundaries, referred to as the continous Galerkin projection. Following the definition of local basis expansions in equation (1.20), we can enforce the solution to be equivalent at the elemental boundary using the condition,

$$\hat{u}_P^{e-1} = \hat{u}_0^e. \tag{1.25}$$

The graphical representation of this condition enforcing C^0 continuity between the element boundaries for three finite elements with $P = 2$ local basis expansions, and the relationship between global and local basis coefficients are shown in figure 1.2. We can relate the global and local basis coefficients with an assembly matrix, $\mathbf{A} \in \mathbb{R}^{N_{loc} \times N}$,

$$\hat{\mathbf{u}}_l = \mathbf{A}\hat{\mathbf{u}}_g. \tag{1.26}$$

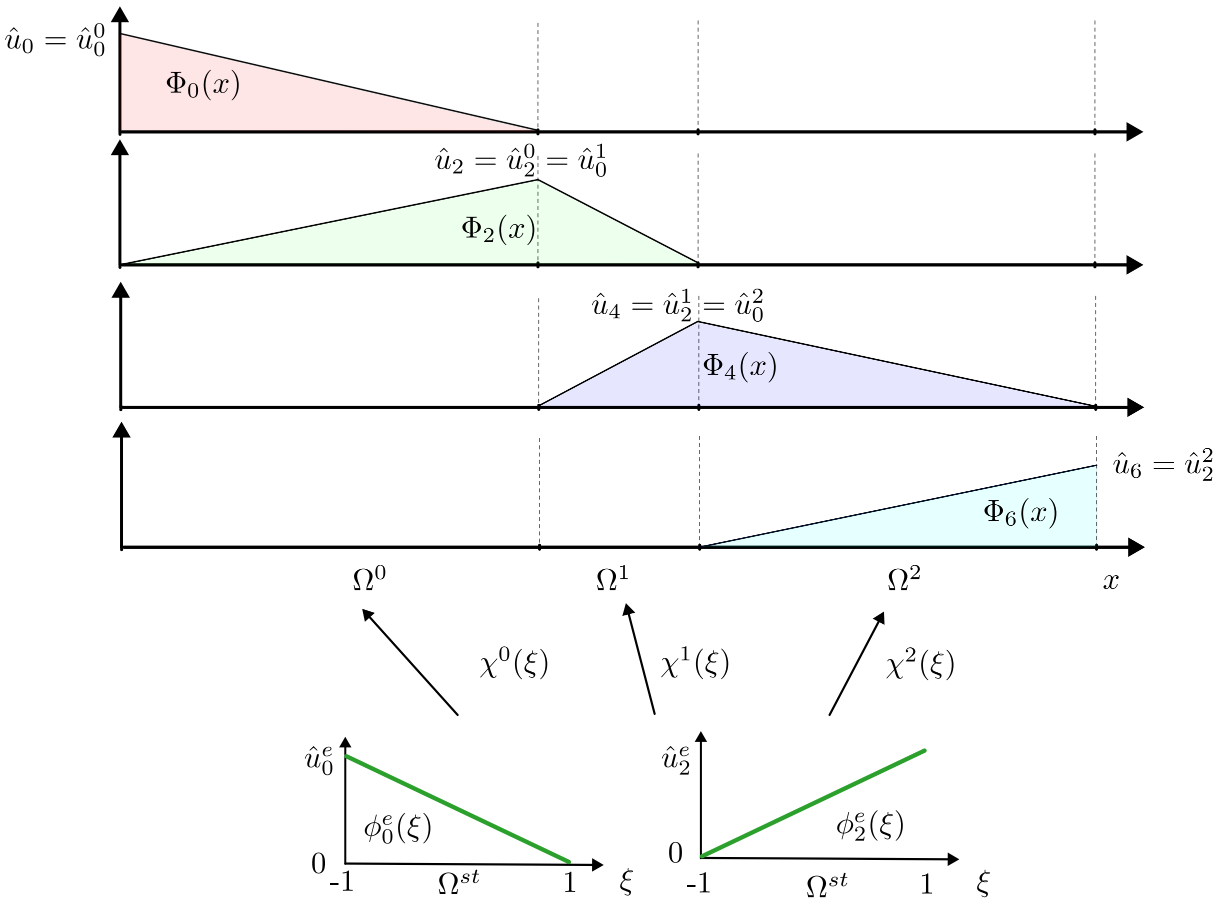

Figure 1.2: A graphical representation of C^0 across elemental boundaries and the relationship between local basis coefficients, u_0^e, u_P^e, and global basis expansions, u_i.

In the case for $P = 2$ and three finite elements as in the case of figures 1.1and 1.2, the assembly matrix and the vectors of global and local basis coefficients are given as,

$$\hat{\mathbf{u}}_l = \begin{pmatrix} \hat{u}_0^1 \\ \hat{u}_1^1 \\ \hat{u}_2^1 \\ \hat{u}_0^2 \\ \hat{u}_1^2 \\ \hat{u}_2^2 \\ \hat{u}_0^3 \\ \hat{u}_1^3 \\ \hat{u}_2^3 \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} 1\,0\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,0\,0\,1 \end{pmatrix}, \quad \hat{\mathbf{u}}_g \begin{pmatrix} \hat{u}_0 \\ \hat{u}_1 \\ \hat{u}_3 \\ \hat{u}_4 \\ \hat{u}_5 \\ \hat{u}_6 \end{pmatrix}, \tag{1.27}$$

The assembly matrix $\mathbf{A}$ ‘scatters’ the global degrees of freedom to local degrees of freedom, while the transpose of it, $\mathbf{A}^T$, performs the reverse, referred to as global assembly. For example, we wish to perform integration in the domain Ω,

$$\mathbf{I}_g[j] = (\Phi_j(x), u^\delta(x)), \tag{1.28}$$

where $\mathbf{I}_g \in \mathbb{R}^N$ refers to a vector containing the integral between $\Phi_i(x)$ and $u^\delta(x)$. This is related to first performing integration using local expansion basis within standard elements, and then assembling using $\mathbf{A}^T$,

$$\mathbf{I}_g = \mathbf{A}^T \mathbf{I}_l, \tag{1.29a}$$

where,

$$\mathbf{I}_g = \begin{bmatrix} \mathbf{I}_0 \\ \vdots \\ \mathbf{I}_{N_g-1} \end{bmatrix}, \quad \mathbf{I}_l = \begin{bmatrix} \mathbf{I}^0 \\ \vdots \\ \mathbf{I}^{N_{el}-1} \end{bmatrix}, \quad \text{with} \quad \mathbf{I}^e = \begin{bmatrix} \int_{-1}^{1} \phi_0^e(\xi) u(\chi^e) \frac{\mathrm{d}\chi^e}{\mathrm{d}\xi}\,\mathrm{d}\xi \\ \vdots \\ \int_{-1}^{1} \phi_{P-1}^e(\xi) u(\chi^e) \frac{\mathrm{d}\chi^e}{\mathrm{d}\xi}\,\mathrm{d}\xi \end{bmatrix}, \tag{1.29b}$$

and $\mathbf{I}_l \in \mathbb{R}^{N_{loc}}$ refer to the vector of integration operations performed within a standard element. In the spectral/*hp* element approach, we perform integration and differentiation using local basis expansions within a standard element. After doing so, we assemble the local operations from the standard element to the global domain by using $\mathbf{A}^T$, as we shall show later using a 1D example. We note that the structure of assembly matrix is generally sparse, where the entries either contain 0, 1 or -1 in multidimensional formulation. Therefore, the assembly matrix is not constructed in practice, and a mapping array is used instead.

1.3.4 Local basis expansions

The choice of local basis expansions, $\phi_i^e(\xi)$, concerns the representation of the solution, and the convergence properties of the numerical solver, in particular, the condition number of the mass and laplacian matrices. In general, the the local basis expansions can be classified into two groups, either *modal* or *nodal* expansions.

Modal expansions

Modal expansions, or hierarchical expansions, describes a set of expansion basis where an expansion set ($\mathcal{X}_{P-1}^\delta$) of order $P-1$, is contained within a set ($\mathcal{X}_P^\delta$) of order P, e.g. $\mathcal{X}_{P-1}^\delta \subset \mathcal{X}_P^\delta$. An example of modal expansions are the Jacobi polynomials, $P_p^{\alpha,\beta}(x)$, representing a family of solutions to the Sturm-Liouville problem within, $x \in [-1, 1]$. The Jacobi polynomials become symmetric for $\alpha = \beta$, referred to ultraspheric polynomials. Special cases of ultraspheric polynomials are the Legendre polynomials, $\alpha = \beta = 1$, and the Chebyshev polynomials, $\alpha = \beta = 1/2$. O Within the Nektar++ framework, we utilise the *modified* basis, based on Jacobi polynomials that are modified (hence the name),

$$\phi_p(\xi) \to \psi_p(\xi) = \begin{cases} \frac{1-\xi}{2} & \text{for } p = 0 \\ \left(\frac{1-\xi}{2}\right)\left(\frac{1+\xi}{2}\right) P_{p-1}^{1,1}(\xi) & \text{for } 0 < p < P \\ \frac{1+\xi}{2} & \text{for } p = P, \end{cases} \tag{1.30}$$

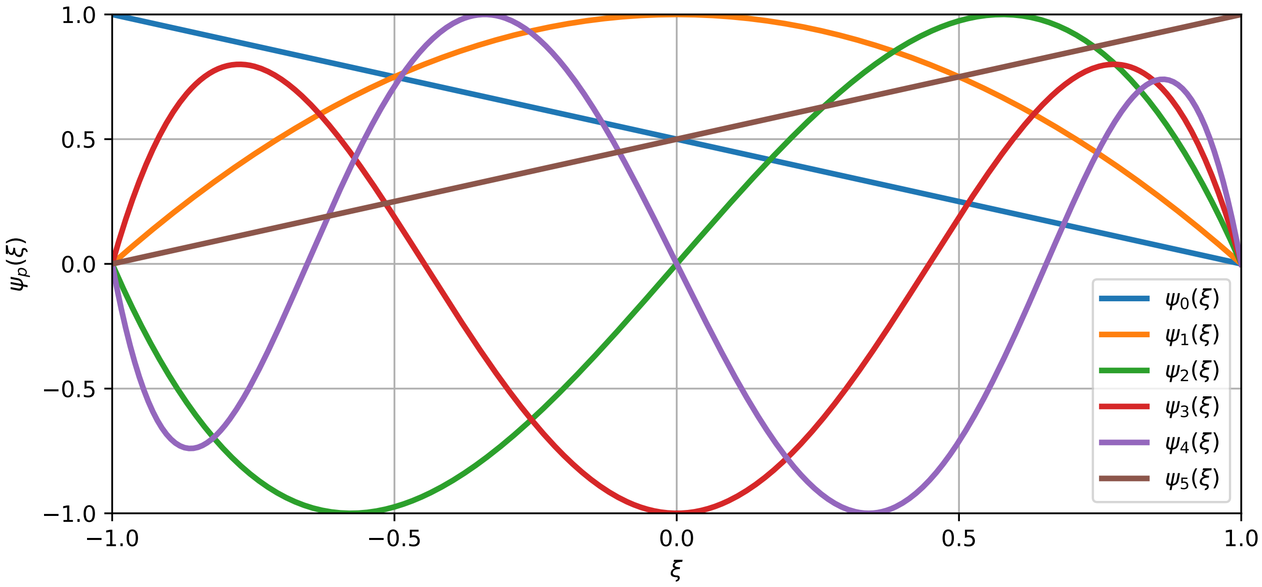

Figure 1.3: The modified basis for up to $P = 5$ normalised to $-1 \leq \psi_p \leq 1$.

We note that $\phi_p(\xi)$ refers to a general local expansion basis while $\psi_p(\xi)$ to definition of the modified basis. The one-dimensional expansion modes of the modified basis of up to $P = 5$ is shown in figure 1.3. The linear modes, corresponding to $p = 0$ and $p = P$, are the only expansions which has a magnitude of at the boundaries, referred to as boundary modes. The modified basis for $0 < p < P$, are hierarchical, and have non-zero values except at the boundaries, referred to as interior/bubble modes. Decomposing the local expasions into interior and boundary mode essentially via a partition of unity, ensures that the linear modes, $P = 1$, supports the C^0 continuity across element boundaries. So strictly speaking, the modified basis is only modal for $P > 1$.

Nodal expansions

Nodal expansions are basis expansions that are non-hierchical, $\mathcal{X}^{\delta}_{P-1} \not\subset \mathcal{X}^{\delta}_{P}$. An example of nodal expansions are the Lagrange polynomials,

$$\phi_p(\xi) \rightarrow h_p(\xi) = \frac{\prod_{q=0,q\neq p}^{P}(\xi - \xi_q)}{\prod_{q=0,q\neq p}^{P}(\xi_p - \xi_q)} \tag{1.31}$$

The Lagrange polynomials, $h_p(\xi)$, are particular attractive as it has a unit value at discrete nodal values, ξ_q, and zero everywhere else, $h_p(\xi_q) = \delta_{pq}$, which implies that

$$u^{\delta}(\xi_q) = \sum_{p=0}^{P} \hat{u}_p h_p(\xi_q) = \sum_{p=0}^{P} \hat{u}_p \delta_{pq} = \hat{u}_q, \tag{1.32}$$

where the Lagrange coefficient $\hat{u}_q$ is the same as the value evaluated at the node ξ_q. The nodal values, ξ_q, are based on the Gauss-Lobatto-Legendre (GLL) points which will be defined later in §1.3.5. The nodal distribution based on GLL points is crucial for obtaining well-behaved Lagrange polynomials, as the use of equispaced nodal points leads to oscillations between nodes. Figure 1.4 presents Lagrange expansions evaluated along the GLL points.

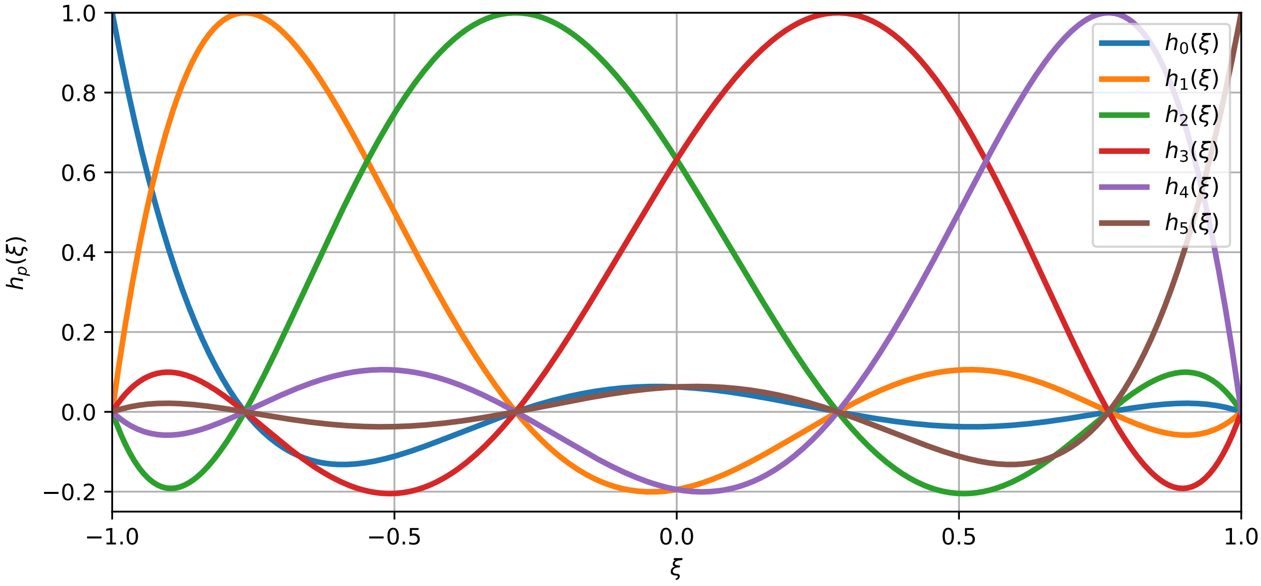

Figure 1.4: Lagrange polynomials for $P = 5$ with nodal values along GLL points.

Multi-dimensional expansions

We have introduced modal and nodal expansions in one dimension, and its extension to multi-dimensions bases can be generalised using a tensorial expansion of the local expansion bases. The standard element in a two dimensional quadrilateral, $\mathcal{Q}^2$, and a three dimensional hexahedral $\mathcal{H}^3$, are given as,

$$\mathcal{Q}^2 = \{-1 \leq \xi_1, \xi_2 \leq 1\}, \quad \mathcal{H}^2 = \{-1 \leq \xi_1, \xi_2, \xi_3 \leq 1\} \tag{1.33}$$

where ξ_1, ξ_2, ξ_3 refers to the local coordinates in multi-dimensions. Thus, the multi-dimensional expansion bases for quadrilaterals and hexadrals using modified bases are simply a tensor product of the one dimensional modified bases,

$$\phi_{pq}(\xi_1, \xi_2) = \psi_q(\xi_1)\psi_q(\xi_2), \quad \text{and} \quad \phi_{pqr}(\xi_1, \xi_2, \xi_3) = \psi_q(\xi_1)\psi_q(\xi_2)\psi_r(\xi_3). \tag{1.34}$$

An example of the modal tensorial bases, for $p = q = 4$ in a standard quadrilateral element in shown in figure 1.5. While we have discussed the tensorial the expansions for regular domains such as the standard quadrilateral and hexahedral elements, the extensions for simplex domains such as triangles, tehtrahedrals, prisms and pyramids commonly used to represent complex geometries, are less straightforward. The challenge for simplexes is that the local coordinates, ξ_1, ξ_2, ξ_3, become dependent where a direct tensorial expansion becomes unwieldy. Instead, a collapsed coordinate system is introduced, providing a transformation from a standard simplex element to a standard regular element. In this thesis, we ultilise quadrilateral elements. The reader is referred to Karniadakis and

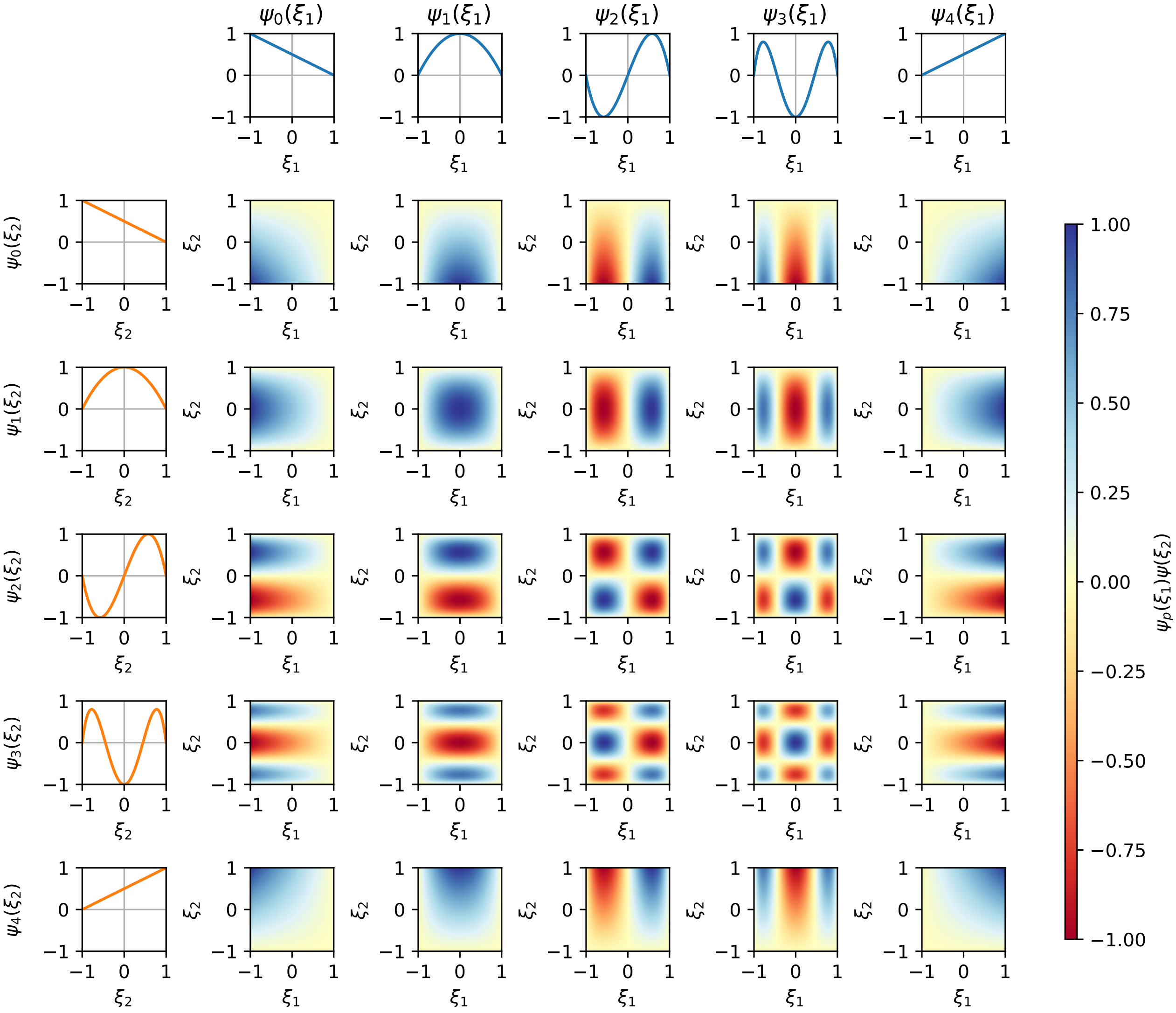

Figure 1.5: Two dimensional modified basis with $p = q = 4$ in a standard quadrilateral, $-1 \leq \xi_1, \xi_2 \leq 1$. The modified bases are normalised to $-1 \leq \phi_{pq} \leq 1$.

Sherwin [2005] for more details about the multi-dimensional formulation of regular and simplex elements.

1.3.5 Gaussian quadrature

In the Galerkin formulation, we perform integration between basis functions rountinely, and an efficient numerical technique is therefore sought. Suppose we want to approximate the integral of a function, $u(\xi)$, in a standard element numerically given as,

$$\int_{-1}^{1} u(\xi)\ \mathrm{d}\xi = \sum_{i=0}^{Q-1} w_i u(\xi_i) + R(u). \tag{1.35}$$

The premise is determine the optimal number of quadrature points, Q, integration weights, w_i, and zeros, ξ_i, in which the integral error, $R(u)$, can be minimised. If $u(\xi)$ is of polynomial order of P, we may expect that we require at least $P+1$ equipspaced points to represent $u(\xi)$ sufficiently. Using Gaussian quadrature rules, we can approximate an integral of a function of order P, with far fewer than $P+1$ points with specific integration weights and zeros. In general, Gaussian quadrature rules can be grouped into three catergories: Gauss, Gauss-Radau and Gunderauss-Lobatto quadrature. The main difference between the three categories are on the inclusion of the end points. Gauss quadrature rule evaluates the integral without the end points $\xi = \pm 1$. Gauss-Radau quadrature rule either select one of the end points, often at $\xi = -1$. Gauss-Lobatto quadrature rule consider both end points. We will only focus on describing the Gauss-Lobatto quadrature rules and the zeros of Jacobi polynomials known as the Gauss-Lobatto-Jacobi quadrature rules given as,

$$\xi_i^{\alpha,\beta} = \begin{cases} -1 & i = 0, \\ \xi_{i-1,Q-2}^{\alpha+1,\beta+1} & i = 1, ..., Q-2, \\ 1, & i = Q-1, \end{cases} \tag{1.36a}$$

$$w_i^{\alpha,\beta} = \begin{cases} (\beta+1)C_{0,Q-2}^{\alpha,\beta}, & i = 0, \\ C_{i,Q-2}^{\alpha,\beta}, & i = 1, ..., Q-2, \\ (\alpha+1)C_{Q-1,Q-2}^{\alpha,\beta}, & i = Q-1, \end{cases} \tag{1.36b}$$

$$C_{i,Q-2}^{\alpha,\beta} = \frac{2^{\alpha+\beta+1}\Gamma(\alpha+Q)\Gamma(\beta+Q)}{(Q-1)(Q-1)!\Gamma(\alpha+\beta+Q+1)[P_{Q-1}^{\alpha,\beta}(\xi_i)]^2} \tag{1.36c}$$

where $w_i^{\alpha,\beta}, \xi_i^{\alpha,\beta}$ are the zeros (or sometimes referred to as quadrature points) and weights of the Gauss-Lobatto-Jacobi quadrature rules, and Γ refers to the Gamma function. For $\alpha = \beta = 0$, the quadrature points is known as the Gauss-Lobatto-Legendre (GLL) points, typically employed for Lagrange polynomials to ensure stability. By evaluating the discrete integral (equation (1.35)) using the zeros and integrations weights (equation (1.36)), we can obtain an exact integral of the function $u(\xi)$, of polynomial order P, with at least $Q \geq (P+3)/2$ quadrature points.

1.3.6 Numerical differentiation

In the same fashion as Gaussian quadrature rules, we want to evaluate the derivative of a function, $u^\delta(\xi)$ numerically. Suppose that we want to differentiate in x using local coordinates given as,

$$\frac{\mathrm{d}u^\delta(\xi)}{\mathrm{d}x} = \frac{\mathrm{d}u^\delta(\xi)}{\mathrm{d}\xi}\frac{\mathrm{d}\xi}{\mathrm{d}x} = \sum_{p=0}^{P} \hat{u}_p \frac{\mathrm{d}\phi_p(\xi)}{\mathrm{d}\xi}\frac{\mathrm{d}\xi}{\mathrm{d}x}, \tag{1.37}$$

where $\mathrm{d}\xi/\mathrm{d}x$ is the jacobian. The primary step involves evaluating the derivative of the local expansion bases, $\mathrm{d}\phi_p(\xi)/\mathrm{d}\xi$. By exploiting the Kronecker delta property of Lagrange polynomials, any polynomials can be conveninently represented by a sum of Lagrange polynomials. Suppose that the solution has polynomial order P, we can represent exactly it with $P+1$ Lagrange polynomials, $\phi_p(\xi) \to h_p(\xi)$. Its derivative through $Q \geq P+1$ discrete nodal points, ξ_i for all $i = 0, 1, \ldots, Q-1$, is expressed as,

$$\left.\frac{\mathrm{d}u^\delta(\xi)}{\mathrm{d}\xi}\right|_{\xi=\xi_i} = \sum_{j=0}^{P} \hat{u}_j \left.\frac{\mathrm{d}h_j(\xi)}{\mathrm{d}\xi}\right|_{\xi=\xi_i} = \sum_{j=0}^{P} d_{ij}\hat{u}_j, \tag{1.38}$$

where $\mathbf{D} = [d_{ij}] \in \mathbb{R}^{Q\times Q}$ refers to the differential matrix containing values of the derivative of the Lagrange polynomials at the quadrature points. Since we are evaluating the derivative along nodal (collocated) points, it is referred to as *collocation* differentiation. The general form for collocation differentiation based on Lagrange polynomials can be expressed as,

$$d_{ij} = \left.\frac{\mathrm{d}h_j(\xi)}{\mathrm{d}\xi}\right|_{\xi=\xi_i} = \begin{cases} \frac{g_Q'(\xi_i)}{g_Q'(\xi_j)}\frac{1}{\xi_i-\xi_j}, & i \neq j, \\ \frac{g_Q''(\xi_i)}{2g_Q'(\xi_i)}, & i = j, \end{cases} \qquad g_Q(\xi) = \prod_{j=0}^{Q-1}(\xi - \xi_j). \tag{1.39}$$

Since d_{ij} only depends on the choice of zeros, alternative forms of $g_q'(\xi_j)$ and $g_q''(\xi_j)$ for the a choice of zeros can be found in Appendix C of Karniadakis and Sherwin [2005]. To perform collocation differentiation using modal basis such as the modified basis, $u^\delta(\xi) = \sum_{j=0}^{P}\hat{u}_i\psi_i(\xi)$, we need to represent the solution in nodal coordinates,

$$\left.\frac{\mathrm{d}u^\delta(\xi)}{\mathrm{d}\xi}\right|_{\xi=\xi_i} = \frac{\mathrm{d}}{\mathrm{d}\xi}\left.\sum_{j=0}^{P}\hat{u}_j\psi_j(\xi)\right|_{\xi=\xi_i} = \mathbf{D}\mathbf{B}\hat{\mathbf{u}}, \tag{1.40}$$

where $\mathbf{B} \in \mathbb{R}^{Q\times(P+1)}$ refers to the standard matrix which transforms the solution from coefficient space, $\hat{\mathbf{u}} \in \mathbb{R}^{P+1}$, to nodal space, $\mathbf{u} \in \mathbb{R}^{Q}$, known as the backwards transform operation, $\mathbf{u} = \mathbf{B}\hat{\mathbf{u}}$.

1.3.7 Example in 1D

We have outlined the basic formulation of spectral/*hp* element methods in a single dimension. To conclude the section on spectral/*hp* element methods, we will describe its solution procedure, converting the weak form of the Helmholtz equation into a system of linear equations, and introduce the mass and laplacian matrices. Starting from the weak form,

$$\lambda\int vu\,\mathrm{d}x + \int\frac{\partial v}{\partial x}\frac{\partial u}{\partial x}\,\mathrm{d}x = \int vf\,\mathrm{d}x + \left[v\frac{\mathrm{d}u}{\mathrm{d}x}\right]_0^l, \qquad x \in [0, l], \tag{1.41a}$$

with boundary conditions,

$$u(0) = g_D, \qquad \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_{x=l} = g_N. \tag{1.41b}$$

We wish to seek the approximate solution $u(x) \approx u^\delta(x)$. Suppose they can be discretised by spectral/*hp* elements with N_{el} elements and $P+1$ local expansion basis,

$$u^\delta(x) = \sum_{e=1}^{N_{el}}\sum_{i=0}^{P}\hat{u}_i^e\Phi_i^e(x) = \sum_{e=1}^{N_{el}}\sum_{i=0}^{P}\hat{u}_i^e\phi_i^e(\xi), \tag{1.42a}$$

with test functions similarly defined as,

$$v^{\delta}(x) = \sum_{e=1}^{N_{el}} \sum_{i=0}^{P} \hat{v}_i^e \Phi_i^e(x) = \sum_{e=1}^{N_{el}} \sum_{i=0}^{P} \hat{v}_i^e \phi_i^e(\xi) \tag{1.42b}$$

Substituting the expansions of u^{δ}, v^{δ} into the weak form (equation (1.41a)) and evaluating the first term on the left hand side using numerical quadrature over Q quadrature points within a standard region, the e^{th} elemental contribution becomes,

$$\begin{aligned}\int_{-1}^{1} \sum_{i=0}^{P} \hat{v}_i^e \phi_i^e(\xi) \sum_{j=0}^{P} \hat{u}_j^e \phi_j^e(\xi) \frac{\mathrm{d}\chi^e}{\mathrm{d}\xi}\,\mathrm{d}\xi &= \sum_{e=0}^{N_{el}-1} \sum_{q=0}^{Q-1} \left[\sum_{i=0}^{P} \hat{v}_i^e \phi_i^e(\xi_q) \sum_{j=0}^{P} \hat{u}_j^e \phi_j^e(\xi_q) \right] J^e w_q^e \\ &= (\hat{\mathbf{v}}^e)^T (\mathbf{B}^e)^T \mathbf{W}^e \mathbf{B}^e \hat{\mathbf{u}}^e \\ &= \hat{\mathbf{v}}^T \mathbf{M}^e \hat{\mathbf{u}}^e\end{aligned} \tag{1.43}$$

where the elemental mass matrix is defined as

$$\mathbf{M}^e = (\mathbf{B}^e)^T \mathbf{W} \mathbf{B}^e \in \mathbb{R}^{(P+1)\times(P+1)}. \tag{1.44}$$

Here, $\mathbf{B}^e \in \mathbb{R}^{Q\times(P+1)}$ refers to the elemental basis matrix, and $\mathbf{W}^e \in \mathbb{R}^{Q\times Q}$, the elemental weight matrix, a diagonal matrix consisting of the product between integration weights, w_q^e, and the element's jacobian, $J^e = \frac{\mathrm{d}\chi^e}{\mathrm{d}\xi}$.

$$\mathbf{B}^e = \begin{bmatrix} \phi_0^e(\xi_0) & \cdots & \phi_P^e(\xi_0) \\ \vdots & \ddots & \vdots \\ \phi_0^e(\xi_Q) & \cdots & \phi_P^e(\xi_Q) \end{bmatrix}, \quad \mathbf{W}^e = \begin{bmatrix} w_0^e J^e & & 0 \\ & \ddots & \\ 0 & & w_{Q-1}^e J^e \end{bmatrix} \tag{1.45}$$

Next, we evaluate the e^{th} elemental contribution of the second term on the left hand side,

$$\begin{aligned}\int_{-1}^{1} \sum_{i=0}^{P} \hat{v}_i^e \frac{\mathrm{d}\phi_i^e}{\mathrm{d}\xi} \sum_{j=0}^{P} \hat{u}_j^e \frac{\mathrm{d}\phi_j^e}{\mathrm{d}\xi} \left(\frac{\mathrm{d}\chi^e}{\mathrm{d}\xi} \right)^{-1} \mathrm{d}\xi &= \sum_{e=0}^{N_{el}-1} \sum_{q=0}^{Q} \left[\sum_{i=0}^{P} \hat{v}_i^e D_{qi}^e \phi_i^e(\xi_q) \sum_{j=0}^{P} \hat{u}_j^e D_{qj}^e \phi_j^e(\xi_q) \right] \frac{w_q^e}{J^e} \\ &= (J^e)^{-1} \hat{\mathbf{v}}^T (\mathbf{B}^e)^T (\mathbf{D}^e)^T \mathbf{W}^e \mathbf{D}^e \mathbf{B}^e \hat{\mathbf{u}}^e \\ &= (J^e)^{-1} \hat{\mathbf{v}}^T \mathbf{L}^e \hat{\mathbf{u}}^e\end{aligned} \tag{1.46}$$

where the elemental Laplacian matrix is defined as

$$\mathbf{L}^e = (J^e)^{-1} (\mathbf{B}^e)^T (\mathbf{D}^e)^T \mathbf{W} \mathbf{D}^e \mathbf{B}^e \in \mathbb{R}^{(P+1)\times(P+1)}, \tag{1.47}$$

and $\mathbf{D}^e \in \mathbb{R}^{Q\times Q}$ refers to the differential matrix defined in equation (1.39). Moving onto the first term on the right hand side,

$$\begin{aligned}
\int_{-1}^{1} \sum_{i=0}^{P} \hat{v}_i^e \phi_i^e(\xi) f^e(\xi) \frac{\mathrm{d}\chi^e}{\mathrm{d}\xi}\,\mathrm{d}\xi &= \sum_{q=0}^{P}\sum_{i=0}^{P} \hat{v}_i^e \phi_i^e(\xi_q) f^e(\xi_q) J^e w_q^e, \\
&= \hat{\mathbf{v}}^T (\mathbf{B}^e)^T \mathbf{W}^e \mathbf{f}^e \\
&= \hat{\mathbf{v}}^T \hat{\mathbf{f}}^e,
\end{aligned} \tag{1.48}$$

where $\hat{\mathbf{f}}^e$ is the elemental forcing vector. By assembling all elemental contributions, $1 \le e \le N_{el}$, we obtain the local system of linear equations,

$$\begin{bmatrix} \left[\lambda\mathbf{M}^1 + \mathbf{L}^1\right] & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \left[\lambda\mathbf{M}^{N_{el}} + \mathbf{L}^{N_{el}}\right] \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}^1 \\ \vdots \\ \hat{\mathbf{u}}^{N_{el}} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{f}}^1 \\ \vdots \\ \hat{\mathbf{f}}^{N_{el}} \end{bmatrix} \tag{1.49a}$$

or in compact form,

$$[\lambda\mathbf{M}_l + \mathbf{L}_l]\hat{\mathbf{u}}_l = \hat{\mathbf{f}}_l. \tag{1.49b}$$

where $\mathbf{M}_l \in \mathbb{R}^{N_{el}(P+1)\times N_{el}(P+1)}$, $\mathbf{L}_l \in \mathbb{R}^{N_{el}(P+1)\times N_{el}(P+1)}$, $\hat{\mathbf{u}}_l \in \mathbb{R}^{N_{el}(P+1)}$ and $\hat{\mathbf{f}}_l \in \mathbb{R}^{N_{el}(P+1)}$ refers to the local mass matrix, local laplacian matrix, vector of local expansion coefficients and local forcing vector. To enforce C^0 continuity across elemental boundaries, we pre-multiply equation (1.49) with the assembly map, $\mathbf{A}^T$, and express local coefficients as global coefficients,

$$\mathbf{A}^T[\lambda\mathbf{M}_l + \mathbf{L}_l]\mathbf{A}\hat{\mathbf{u}}_g = \mathbf{A}^T\hat{\mathbf{f}}. \tag{1.50}$$

Finally, we consider the boundary conditions by lifting $u^\delta(x) = u^{\mathcal{H}}(x) + u^{\mathcal{D}}(x)$

$$\mathbf{A}^T[\lambda\mathbf{M}_l + \mathbf{L}_l]\mathbf{A}\hat{\mathbf{u}}_g^{\mathcal{H}} = \mathbf{A}^T\hat{\mathbf{f}} + \mathbf{A}^T[\lambda\mathbf{M}_l - \mathbf{L}_l]\mathbf{A}\hat{\mathbf{u}}_g^{\mathcal{D}}. \tag{1.51}$$

We then solve this system for the homogeneous coefficients $\hat{\mathbf{u}}_g^{\mathcal{H}}$.

1.3.8 Error properties

Suppose we consider $P+1$ orthogonal polynomials spanning the polynomial space of $\mathcal{P}_P$, the energy norm of the error, $||\varepsilon||_E = ||u(x) - u^\delta(x)||_E$[2] for element size of h and polynomial order P, satisfies the convergence property [Babuška and Suri, 1994],

$$||\varepsilon||_E \le C h^{\mu-1} P^{-(k-1)} ||u||_k, \tag{1.52}$$

[2] The energy norm is defined as $||u||_E = \sqrt{a(u,u)}$, where $a(u,u)$ is the bilinear operator introduced in equation (1.7b). It is a measure of functions belonging to the *energy* space which are H^1 functions.

where $\mu = \min(k, P+1)$, and C is independent of h, P and u. To illustrate this convergence property, we consider the exact solution, $u = \sin(\pi x)$, to the 1D Helmholtz problem (equation (1.5)) on the domain $x \in [-1, 1]$ with Dirichlet boundary conditions on both ends and $\lambda = 1$. This convergence property arising from the h- and p-type refinements is shown in figure 1.6. To ensure consistency between two types of refinement, the number of degrees of freedom, N_{dof}, is shown on the absicca, which scales as $\mathcal{O}(h) \sim \mathcal{O}(N_{dof}^{-1})$ and $\mathcal{O}(p) \sim \mathcal{O}(N_{dof})$. For h-refinement, the number of elements is doubled on a mesh with fixed polynomial order $P = 1$, demonstrating algebraic convergence, $||\varepsilon||_E \sim \mathcal{O}(h)$ in figure 1.6(a). For p-refinement, the polynomial order P is increased on a mesh with two equispaced elements, showing exponential convergence, $||\varepsilon||_E \sim \mathcal{O}(c^p)$ where c is a constant, in figure 1.6(b).

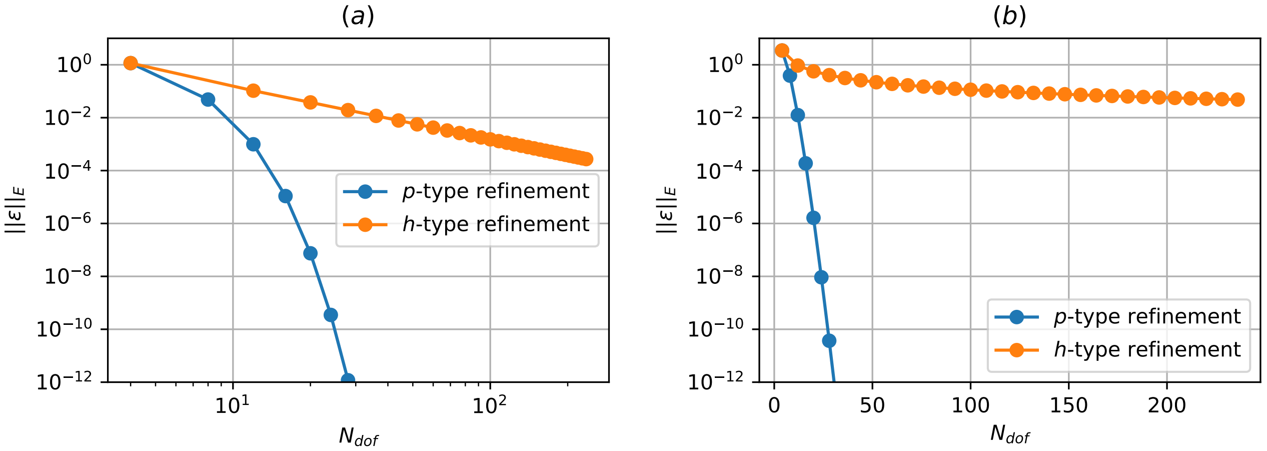

Figure 1.6: Convergence properties of an exact solution $u = \sin(\pi x)$ to a 1D Helmholtz equation with $\lambda = 1$ and Dirichlet boundary conditions at $x \in [-1, 1]$. h-type refinement is performed on a mesh with fixed $P = 1$ and decreasing h (increasing elements), demonstrating algebraic convergence in a *log-log* plot of (a). p-type refinement is performed on a mesh with equipspaced two elements and increasing P, demonstrating exponential convergence on a *semi-log* plot of (b).

1.4 Numerical techniques for solving the Navier-Stokes equations

1.4.1 Velocity Correction Scheme

The spatial discretisation of the Helmholtz operator and its numerical solution procedure has been discussed using the spectral/*hp* element methods. Here, we describe the numerical methods that is used to solve the Navier-Stokes equations given as,

$$\frac{\partial u}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f} \tag{1.53a}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{1.53b}$$

with boundary conditions,

$$\mathbf{u} = 0, \quad \text{on } \partial\Omega. \tag{1.53c}$$

In the above, the primitive variables are velocity and pressure $(\mathbf{u}, p)$ and we assumed unit density, $\rho = 1$, with the kinematic viscosity appearing as the control parameter. The time evolution of velocity is explicit expressed in equation (1.53a), but does not appear for the pressure, which is coupled to the velocity field, enforcing the incompressibility condition. Several strategies exist for addressing the coupled velocity-pressure fields by

1. Solving the coupled system such as the Uzawa algorithm,

2. Splitting methods,

We adopt splitting methods, which solves the of the Navier-Stokes equation by splitting them into ‘subequations’, and solving them sequentially. These methods, belonging to the broader family of projection methods introduced by Chorin [1967] and Témam [1969], and can be further classified into pressure-correction or velocity-correction schemes. This thesis employs the use of the high-order velocity-correction scheme introduced by Karniadakis et al. [1991] and further explained by Guermond and Shen [2003]. We rewrite the incompressible Navier-Stokes equations in semi-discrete form with using linear and nonlinear operators as,

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{N}(\mathbf{u}) - \nabla p + \nu \mathbf{L}(\mathbf{u}), \tag{1.54a}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{1.54b}$$

with boundary conditions,

$$\mathbf{u}|_{\Omega} = 0, \quad \mathbf{u}(t=0) = \mathbf{u}_0. \tag{1.54c}$$

The nonlinear, $\mathbf{N}$, linear, $\mathbf{L}$, operators are obtained from a suitable spatial-discretisation method such as the spectral/*hp* element method. The nonlinear and linear operators are defined as,

$$\mathbf{N}(\mathbf{u}) \equiv -(\mathbf{u} \cdot \nabla)\mathbf{u}, \qquad \mathbf{L}(\mathbf{u}) \equiv \nabla^2 \mathbf{u}, \tag{1.55}$$

To minimise aliasing errors in the quadratic nonlinear terms, we performing numerical integration by increasing the number of quadrature points to $Q \geq (P+3)/2$. To advance the velocity at time step, $\mathbf{u}^n$, to the next time step, $\mathbf{u}^{n+1}$, we integrate equation (1.54) over a time step Δt,

$$\mathbf{u}^{n+1} - \mathbf{u}^n = \underbrace{\int_{t_n}^{t_{n+1}} \mathbf{N}(\mathbf{u})\,\mathrm{d}t}_{\Delta t \sum_{q=0}^{J_e-1} \beta_q \mathbf{N}(\mathbf{u}^{n-q})} - \underbrace{\int_{t_n}^{t_{n+1}} \nabla p\,\mathrm{d}t}_{\Delta t \nabla \bar{p}^{n+1}} + \nu \underbrace{\int_{t_n}^{t_{n+1}} \mathbf{L}(\mathbf{u})\,\mathrm{d}t}_{\Delta t \sum_{q=0}^{J_i-1} \gamma_q \mathbf{L}(\mathbf{u}^{n+1-q})} . \tag{1.56}$$

The velocity correction scheme ‘splits’ equation (1.54) into three steps shown as underbraces, which are evaluated successively from left to right. The first step we perform is to extrapolate the advection velocities, by approximating the nonlinear terms using an explicit scheme such as the Adams-Bashforth family of J_e order,

$$\frac{\hat{\mathbf{u}} - \sum_{q=0}^{J_e-1} \alpha_q \mathbf{u}^{n-q}}{\Delta t} = \sum_{q=0}^{J_e-1} \beta_q \mathbf{N}(\mathbf{u}^{n-q}), \tag{1.57}$$

where $\hat{\mathbf{u}}$ is denotes the primary intermediate velocity field desired and α_e, β_e refers to the time integration coefficients for a prescribe J_e-th order, described later. After evaluting $\hat{\mathbf{u}}$, we move onto the second term in equation (1.56), which defines the pressure at time step $n+1$ as,

$$\frac{\hat{\hat{\mathbf{u}}} - \hat{\mathbf{u}}}{\Delta t} = -\nabla p^{n+1}. \tag{1.58}$$

$\hat{\hat{\mathbf{u}}}$ denotes as the secondary intermediate velocity. In this single equation, we seek to obtain two unknown solutions, $\hat{\hat{\mathbf{u}}}$ and p^{n+1}, which is ill-posed, and seek to impose certain restrictions. The splitting method assumes that the secondary intermediate velocity is divergence free, $\nabla \cdot \hat{\hat{\mathbf{u}}} = 0$, and satisfies the Dirichlet boundary conditions normal to the boundary, $\hat{\hat{\mathbf{u}}} \cdot \mathbf{n} = \mathbf{u}|_{\Omega} \cdot \mathbf{n}$. By considering the assumptions above and the divergence of equation (1.58), we obtain the pressure Poisson equation with the primary intermediate velocity acting as the forcing term,

$$\nabla^2 p^{n+1} = \nabla \cdot \left(\frac{\hat{\mathbf{u}}}{\Delta t} \right) \tag{1.59a}$$

and boundary conditions,

$$\frac{\partial p^{n+1}}{\partial n} = \mathbf{n} \cdot \left(\frac{\hat{\hat{\mathbf{u}}} - \hat{\mathbf{u}}}{\Delta t} \right). \tag{1.59b}$$

While the pressure boundary condition (1.59b) is straightforward to evalute, it is sensitive to large splitting errors [Karniadakis et al., 1991]. To overcome this, we consider a high-order boundary condition of pressure, obtained by taking the dot product with respect to the normal of equation (1.54),

$$\frac{\partial p^{n+1}}{\partial t} = -\sum_{q=0}^{J_e-1} \beta_q \left[\frac{1}{\Delta t} \mathbf{u}^{n-q} + \nu [\nabla \times (\nabla \times \mathbf{u}^{n-q})] + (\mathbf{u}^{n-q} \cdot \nabla) \mathbf{u}^{n-q} \right] \cdot \mathbf{n}. \tag{1.60}$$

Notably, the linear operator is expressed as $\mathbf{L}(\mathbf{u}) = \nabla(\nabla \cdot \mathbf{u}) - \nabla \times (\nabla \times \mathbf{u})$, where the divergence is set to zero, favouring numerical stability [Orszag et al., 1986, Karniadakis et al., 1991]. J_e is the order the explicit scheme as in equation (1.57). After solving for the pressure Poisson equation, the secondary intermediate velocity could be subsequently obtained using equation (1.58). After which, we can move onto the final substep in equation (1.56), by solving a Helmholtz equation for $\mathbf{u}^{n+1}$,

$$\frac{\gamma_0 \mathbf{u}^{n+1} - \hat{\hat{\mathbf{u}}}}{\Delta t} = \nu \sum_{q=0}^{J_i-1} \gamma_q \mathbf{L}(\mathbf{u}^{n+1-q}), \tag{1.61}$$

where the linear terms are treated based similar to the family of Adams-Moulton implicit scheme and J_i, γ_q denotes the order of the scheme and time integration coefficients, completing the velocity correction scheme. The time integration coefficients are determined from stiffly stable schemes shown in table 1.2, an improvement from the Adams-family schemes [Karniadakis et al., 1991]. The high

Coefficients	1^{st} order	2^{nd} order	3^{rd} order
γ_0	1	3/2	11/6
α_0	1	2	3
α_1	0	-1/2	-3/2
α_2	0	0	1/3
β_0	1	2	3
β_1	0	-1	-3
β_2	0	0	1

Table 1.2: Integration coefficient of stiffly stable schemes from Karniadakis et al. [1991].

order velocity correction scheme and be summarised in a three step process of the following,

$$\mathbf{u}^n \xrightarrow{\mathbf{N}(\mathbf{u}^n)} \hat{\mathbf{u}} \xrightarrow{\nabla^2 p} \hat{\hat{\mathbf{u}}} \xrightarrow{\mathbf{L}(\hat{\hat{\mathbf{u}}})} \mathbf{u}^{n+1},$$

evolving the velocity fields at time step n to $n+1$.

1.4.2 Fourier spectral/*hp* modes

Fourier-Chebyshev-Fourier type discretisation have been recognised as preferred method for performing direct numerical simulations (DNS) of transitional or turbulent incompressible channel flows [Kim et al., 1987] owing to its efficient representation of the inhomogeneous wall-normal directions and the homogeneous streamwise and spanwise directions, using Chebyshev and Fourier expansions respectively.

The Fourier spectral/*hp* element method draws on this approach, where the homogeneous and the inhomogeneous directions are represented by the Fourier expansions and spectral/*hp* elements respectively. This approach has been commonly referred to as the Quasi-3D or (2.5D) approach, allowing for the representation of two inhomogeneous directions. For example, in the turbulent channel flows with riblets, the Fourier expansions are used to represent the periodic streamwise, while the spectral/*hp* elements are used to discretise the wall-normal direction. In the analysis of three-dimensional wakes of cylinders where the Fourier expansions are treated in the spanwise directions. In this thesis, we routinely use the the Quasi-3D approach, consisting of the 2D spectral/*hp* elements with 1D Fourier expansions are used to discretise the cross stream plane and streamwise flow respectively. The velocity and pressure in the spectral/*hp* plane is described by two dimensional modified bases with Fourier expansions,

$$\begin{bmatrix} \mathbf{u}^\delta(x,y,z,t) \\ p^\delta(x,y,z,t) \end{bmatrix} = \sum_{k=0}^{N_z-1} \sum_{p=0}^{P} \sum_{q=0}^{P} \psi_p(x)\psi_q(y) e^{ik\beta z} \begin{bmatrix} \hat{\mathbf{u}}_{p,q,k}(t) \\ \hat{p}_{p,q,k}(t) \end{bmatrix} = \sum_{k=0}^{N-1} e^{ik\beta z} \begin{bmatrix} \tilde{\mathbf{u}}_k(x,y,t) \\ \tilde{p}_k(x,y,t) \end{bmatrix} \tag{1.62}$$

where $\beta = \frac{2\pi}{L_z}$ is the spanwise wavenumber, L_z the spanwise length, N_z the number of Fourier expansions. Substituting equation 1.62 into the Navier-Stokes equations and taking the Fourier transform (equivalently to the Galerkin projection with respect to Fourier expansion as a test function) yields a system of N_z decoupled equations, amenable for parallel processing,

$$\frac{\partial \tilde{\mathbf{u}}_k}{\partial t} = -\tilde{\nabla}_k \tilde{p}_k + \nu(\nabla^2_{x,y} - k^2\beta^2)\tilde{\mathbf{u}}_k - [\widehat{(\mathbf{u}\cdot\nabla)\mathbf{u}}]_k \tag{1.63a}$$

$$-k\beta\tilde{\nabla}\cdot\tilde{\mathbf{u}}_k = 0, \quad k = 0, ..., N_z - 1 \tag{1.63b}$$

where, $\tilde{\nabla}_k = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, ik\beta)$, $\nabla^2_{x,y} = (\frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2})$ and $[\widehat{(\mathbf{u}\cdot\nabla)\mathbf{u}}]_k$ refers to the Fourier-transformed of the k^{th} nonlinear term.

1.4.3 Maintaining fluid flow through a channel

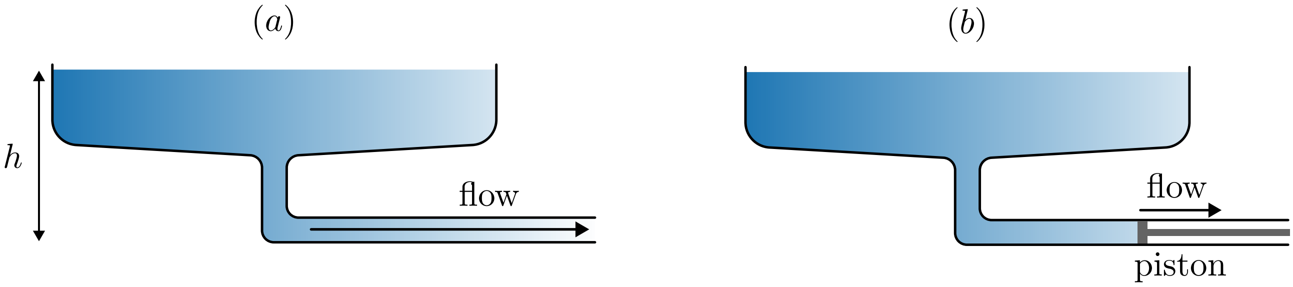

Figure 1.7: (a) Flow rate driven by a pressure gradient from an reservoir elevated by h. (b) Flow driven by a piston at a constant flow rate.

In general, there are two approaches to drive a fluid flow through a channel, either by maintaining a constant pressure drop, or a constant volumetric flux (flow rate). This difference is illustrated in figure 1.7, whereby the flow through the channel is driven by a constant pressure drop from an elevated reservoir of constant height h in figure 1.7(a), while a piston moves at a constant speed rightwards, drawing fluid through the channel at a constant volumetric flux in figure 1.7(b).

Constant pressure via body-forcing

If we were to assign the homogeneous direction (i.e. discretised based on Fourier expansions) as the streamwise direction, a pressure difference cannot be prescribe directly due to the enforced periodicity. Instead, we replace the constant pressure drop with a constant body force $\mathbf{f} = f_x\hat{\mathbf{e}}_x$ in the streamwise direction,

$$\frac{\partial u}{\partial t} + (u\cdot\nabla)u = -\nabla p + \nu\nabla^2 u + f_x, \tag{1.64}$$

The central question now becomes what is the magnitude of body force required to maintain a laminar or turbulent flow. To begin this discussion, we assume that we can decompose our flow variables into a mean and a fluctuating component,

$$u(x, y, t) = U(y) + u'(x, y, t), \tag{1.65}$$

where $U(y) = \langle u \rangle$ refers to the averaged velocity and $\langle \cdot \rangle = \frac{1}{TL_xL_z}\int \cdot \, dzdxdt$ refers to the temporal and span-averaged operator. The fluctuating component is defined with an average of 0, i.e. $\langle u' \rangle = 0$. Next, we substitute this decomposition into equation (1.64), and perform the averaging operation,

$$\begin{aligned}\Bigg\langle \frac{\partial(U+u')}{\partial t} + (U+u')\frac{\partial(U+u')}{\partial x} + (V+v')\frac{\partial(V+v')}{\partial y} \\ = -\frac{\partial(P+p')}{\partial x} + \nu\nabla^2(U+u') + F_x + f'_x \Bigg\rangle.\end{aligned} \tag{1.66}$$

For a statistically stationary turbulent (or laminar) channel flow with periodic streamwise boundary conditions, we can make the following assumptions:

1. stationary flow $\frac{\partial U}{\partial t} = 0$,
2. fully-developed in x, $\frac{\partial}{\partial x} \to 0$,
3. $\frac{\partial V}{\partial y} = 0$, as a consequence of continuity and the no-slip boundary condition.
4. $\langle u', v', w', p' \rangle = 0$, based on the definition of fluctuations,
5. $\frac{\partial p}{\partial x} = 0$ due to the enforced periodicity in x.

Applying the assumptions above, the mean momentum equations simplify into,

$$\langle F_x \rangle = \left\langle \frac{\partial(u'v')}{\partial y} \right\rangle - \nu\frac{\partial U^2}{\partial y^2}, \tag{1.67}$$

where the body force on the left-hand side balances the sum of Reynolds stresses and viscous diffusion on the right-hand side. Next, we integrate the expression from $y \in [-1, 1]$,

$$2F_x = [\langle u'v' \rangle]_{y=-1}^{y=1} + \nu\left[\frac{\partial U}{\partial y}\bigg|_{y=1} - \frac{\partial U}{\partial y}\bigg|_{y=-1}\right]. \tag{1.68}$$

The wall shear stress is defined by $\tau_w = \nu\frac{\partial U}{\partial y}|_{y=1}$ (ρ is assumed to be 1), and it is antisymmetric about the channel centreline, $\nu\frac{\partial U}{\partial y}|_{y=1} = -\nu\frac{\partial U}{\partial y}|_{y=-1}$. Due to the no-slip condition, the Reynolds shear stresses is zero, i.e. $[u'v']|_{y=-1,1} = 0$. Hence, the expression above simplifies to,

$$\tau_w = F_x. \tag{1.69}$$

In other words, the body force F_x is balanced by the wall shear stress (drag), τ_w, along the channel walls. In the case of laminar flow, τ_w can be determined analytically, and the body force required for sustaining a laminar flow for a velocity profile of $u(y) = 1 - y^2$, is $F_x = -2\nu$. However, to determine the wall shear stress (and hence the magnitude of body force) is not as straightforward task for transitional or turbulent channel flow as there isn't an analytical expression for τ_w and its dependence on Reynolds number. Instead, we can only rely on empirical relations of turbulent channel flow between the skin friction coefficient, $c_f = \tau_w / \frac{1}{2}\rho U_c^2$ and Reynolds number Re_c from Dean [1978].

$$c_f = 0.00302 Re_c^{-1/4}, \tag{1.70}$$

where Re_c is the Reynolds number based on the laminar centerline velocity. Similarly, the skin friction coefficient for the case of laminar flow is $c_f = 4/Re_c$ [Dean, 1978]. Figure 1.8 illustrates the relationship between τ_w and Re_c of channel flow using empirical relationship from Dean [1978] (here $\rho = U_c = 1$) and experimental data from Patel and Head [1969], Kim et al. [1987], Iida and Nagano [1998], Tsukahara et al. [2014]. While the empirial relation for laminar flow, $Re_c \lesssim 1000$ and turbulent flow $Re_c \gtrsim 2000$ appears reasonably robust, the wall shear stress in the transitional region is lacking therefore, the body forcing approach is not preferred.

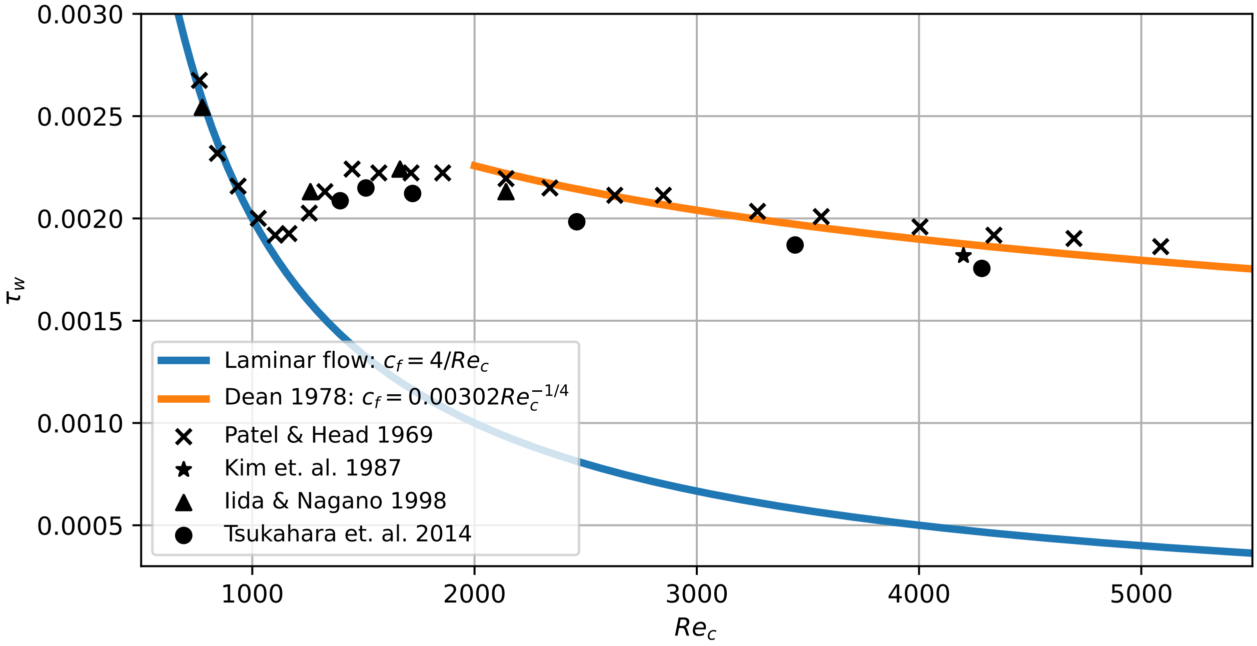

Figure 1.8: τ_w against Re_c using skin friction coefficients from Dean [1978] with $\rho = U_c = 1$. Experimental scatter points from [Patel and Head, 1969, Kim et al., 1987, Iida and Nagano, 1998, Tsukahara et al., 2014].

Constant volumetric flux

An alternative approach is to enforce a constant volumetric flux, illustrated using the piston method in figure 1.7(b). We employ the efficient Green's function approach introduced by Chu and Karniadakis [1993], and outline its solution procedure. The volumetric flux is defined as,

$$Q(\mathbf{u}) = \frac{1}{A_s} \int_R \mathbf{u} \cdot \mathrm{d}\mathbf{R}, \tag{1.71}$$

where $Q(\cdot)$ refer to the flow rate operator through the surface R with surface area of A_s. The idea is to append a correction velocity, $\mathbf{u}_{corr}$, to the velocity field at time step n, $\mathbf{u}^n$, such that the corrected solution, $\bar{\mathbf{u}}^n = \mathbf{u}^n + \mathbf{u}_{corr}$, has the desired volumetric flux $\bar{Q} = Q(\bar{\mathbf{u}}^n)$. While adding two solutions together is straightforward, the resultant velocity field may not directly satisfy the Navier-Stokes equations. Fortunately, we can leverage the velocity correction scheme which (in general) evaluates the nonlinear advection terms followed by a linear terms (pressure and dissipation). This process is summarised as,

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} = \mathbf{N}(\mathbf{u}) \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}^n \end{cases} \xrightarrow{\hat{\mathbf{u}}(\mathbf{x},\Delta t)} \begin{cases} \frac{\partial \mathbf{u}}{\partial t} = -\nabla p + \nu \mathbf{L}(\mathbf{u}) \\ \mathbf{u}(\mathbf{x}, 0) = \hat{\mathbf{u}}(\mathbf{x}, \Delta t), \end{cases} \tag{1.72}$$

where $\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}^n$ and $\hat{\mathbf{u}}(\mathbf{x}, \Delta t)$ refer to the initial condition for the nonlinear advection terms, and the intermediate velocity, the initial condition for the linear terms, respectively. Since the second step correspond to solving the linear Stokes equation, any solution of the linear Stokes (such as $\mathbf{u}_{corr}$) added to the final solution will still satisfy the linear Stokes equations - a property of linear differential equations. We consider the linear Stokes equation governing the evolution of the correction velocity,

$$\frac{\partial \mathbf{u}_{corr}}{\partial t} = -\nabla p_{corr} + \nu \mathbf{L}(\mathbf{u}_{corr}) + \alpha^n \hat{\mathbf{e}}_{\mathbf{x}}, \tag{1.73}$$

where α^n is the undetermined magnitude of body force at time step n in the streamwise direction, $\hat{\mathbf{e}}_x$, required to maintain the desired flow rate $\bar{Q} = Q(\mathbf{u}^n) + Q(\mathbf{u}_{corr})$. Since $\mathbf{u}_{corr}$ is appended to $\mathbf{u}^n$, the initial condition for $\mathbf{u}_{corr}$ must be $\mathbf{u}_{corr}(\mathbf{x}, 0) = 0$, so that $\mathbf{u}^n$ remains compatible with the initial conditions in equation (1.72). Since α^n is undetermined, we normalise the equation with respect to α^n, yielding the linear Stokes equations with unit forcing,

$$\frac{\partial \mathbf{v}}{\partial t} = -\nabla \hat{p} + \nu \mathbf{L}(\mathbf{v}) + \hat{\mathbf{e}}_{\mathbf{x}}, \quad \mathbf{v}(\mathbf{x}, 0) = \mathbf{0}, \tag{1.74}$$

where $\mathbf{v} = \mathbf{u}_{corr}/\alpha^n$ and $\hat{p} = p_{corr}/\alpha^n$. The corrected velocity field becomes

$$\bar{\mathbf{u}} = \mathbf{u} + \alpha^n \mathbf{v}^1, \tag{1.75}$$

where $\mathbf{v}^1$ is solution field obtained by solving equation (1.74) in the first time step. To match the target volumetric flux, $\bar{Q}$, we need to scale α^n such that,

$$\bar{Q} = Q(\bar{\mathbf{u}}^n) = Q(\mathbf{u}^n) + Q(\alpha^n \mathbf{v}^1). \tag{1.76}$$

which gives,

$$\alpha^n = \frac{\bar{Q} - Q(\mathbf{u}^n)}{Q(\mathbf{v}^1)}, \tag{1.77}$$

evaluated at every time step n. The Green's function approach is computationally efficient as we only need to compute $\mathbf{v}^1$ and $Q(\mathbf{v}^{\mathbf{1}})$ once during the first time step and reuse it for subsequent time steps. The process of adding the correction velocity at the end of velocity correction scheme can be summarised in the procedure as follows,

$$\mathbf{u}^n \xrightarrow{\mathbf{N}(\mathbf{u}^n)} \hat{\mathbf{u}} \xrightarrow{\nabla^2 p} \hat{\hat{\mathbf{u}}} \xrightarrow{\mathbf{L}(\hat{\mathbf{u}})} \mathbf{u}^{n+1} \xrightarrow{\alpha^{n+1}\mathbf{v}^1} \bar{\mathbf{u}}^{n+1}.$$

1.5 Stability analysis of the Navier-Stokes equations

1.5.1 Algorithms for linear stability analysis

In this section, we present a general overview of the numerical procedure for linear stability analysis. Linear stability analysis examines the stability of a base flow by considering the evolution of infinitesimal perturbations. These perturbations in general, may either grow or decay exponentially, indicating whether the base flow is linearly unstable or stable respectively. In §**??**, we introduced linear stability analysis in the context of wall-bounded shear flows leading to the Orr-Sommerfeld equations, where the base flows depend on a single inhomogeneous and two homogeneous directions, commonly referred to as local[3] stability analysis. For example, the laminar Poiseuille flow, $U(y) = 1 - y^2$ and the laminar Couette flow $U(y) = y$, $y \in [-1, 1]$. For some flows such as boundary layers, wakes and jets, their base flows are not strictly parallel. By considering a weak dependance on the stream and spanwise directions, their stability are described by the parabolised stability equations [Herbert, 1997]. When the base flow depends on two spatially inhomogeneous directions, $U(x, y)$, or three spatially inhomogeneous directions, $U(x, y, z)$, the analysis of such states are commonly referred to as biglobal or triglobal stability analysis, respectively [Theofilis, 2003]. If the base flow is time-dependent, such as in the secondary instability of cylinder flows, we use Floquet stability analysis [Henderson and Barkley, 1996].

In this section, we consider a time-independent base flow and consider a generic decomposition of the velocity field in three spatial dimensions,

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{U}(\mathbf{x}) + \mathbf{u}'(\mathbf{x}, t), \tag{1.78}$$

where $\mathbf{U}(\mathbf{x}), \mathbf{u}'(\mathbf{x}, t)$ refers to the base flow and perturbations. Substituting this into the Navier-Stokes equations and linearising,

$$\frac{\partial \mathbf{u}'}{\partial t} = -(\mathbf{U} \cdot \nabla)\mathbf{u}' - (\mathbf{u}' \cdot \nabla)\mathbf{U} - \nabla p' + \frac{1}{Re}\nabla^2 \mathbf{u}', \tag{1.79a}$$

$$\nabla \cdot \mathbf{u}' = 0. \tag{1.79b}$$

This can be rewritten in as,

$$\frac{\partial}{\partial t}\mathbf{q}' = \mathcal{L}\mathbf{q}', \quad \mathcal{L} = \begin{bmatrix} -(\mathbf{U} \cdot \nabla) - (\nabla \mathbf{U}) + \frac{1}{Re}\nabla^2 & -\nabla \\ \nabla \cdot & \mathbf{0} \end{bmatrix}, \tag{1.80}$$

[3] Referring to being spatially local in the context of 'real' flows which are typically inhomogeneous in all directions

where $\mathcal{L}$ refer to the linearised operator and, $\mathbf{q}' = (\mathbf{u}', p')^T$. Assuming an initial perturbation, $\mathbf{q}'(\mathbf{x}, t = 0) = \mathbf{q}_0$, its evolution to time T is given by,

$$\mathbf{q}(\mathbf{x}', T) = \mathcal{A}(T, Re)\mathbf{q}_0, \quad \text{where} \quad \mathcal{A}(T, Re) = \exp(\mathcal{L}T). \tag{1.81}$$

We assume that the perturbations can be represented as a normal mode,

$$\mathbf{q}'(\mathbf{x}, t) = \tilde{\mathbf{q}}(\mathbf{x}) \exp(\lambda t) + \text{c.c} \tag{1.82}$$

where $\lambda_j, \tilde{\mathbf{q}}_j(x)$ refer to the j^{th} eigenvalue and eigenmode, and c.c refers to the complex conjugate. Substituting the normal mode into equation (1.81), we obtain an eigenvalue problem,

$$\mathcal{A}(T, Re)\tilde{\mathbf{q}}_j = \mu_j \tilde{\mathbf{q}}_j, \quad \mu_j = \exp(\lambda_j T). \tag{1.83}$$

where μ_j refers to the eigenvalue of $\mathcal{A} = \exp(\mathcal{L}T)$, and we typically set $T = 1$ [Barkley et al., 2008]. The real component of the eigenvalues determine the stability of the base flow, which can be either,

1. Unstable: $\Re(\lambda) > 0$,
2. Stable: $\Re(\lambda) < 0$,
3. Neutral: $\Re(\lambda) = 0$.

This concludes the mathematical overview of linear stabiltiy analysis, and the challenge lies in the computing the eigenpairs of $\mathcal{A}$ efficiently. For large matrices, $\mathcal{A} \in \mathbb{R}^{M \times M}$ (assuming it is real here for simplicity), direct eigenvalue solvers such as the QR algorithm costing $O(M^3)$ might be computationally infeasible. Another concern is that we are typically only interested in the most dangerous (leading) eigenvalues of largest real parts, and not the full spectrum. Lastly, we do not have access to $\mathcal{A}$ in a time stepping based code.

Power Iteration Method

A simple method in computing the dominant eigenpair is the power iteration method,

Definition 1.5.1 (Power iteration)**.** Given a diagonalisable matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ and a non-zero vector $\mathbf{x}_0$, the sequence of matrix vector products between them (we neglect normalisation here),

$$\mathbf{A}\mathbf{x}_0, \mathbf{A}^2\mathbf{x}_0, \mathbf{A}^3\mathbf{x}_0, ..\mathbf{A}^k\mathbf{x}_0. \tag{1.84}$$

approaches the eigenvector of $\mathbf{A}$ with the largest magnitude. i.e. $\tilde{\mathbf{x}}_1 = \lim_{k\to\infty} \mathbf{A}^k\mathbf{x}_0$. The dominant eigenvalue, λ_1, can be computed using the Rayleigh quotient, $\lambda_1 = \frac{\tilde{\mathbf{x}}_1^T \mathbf{A} \tilde{\mathbf{x}}_1}{\tilde{\mathbf{x}}_1^T \tilde{\mathbf{x}}_1}$.

Arnoldi Method

A generalisation of the power method is the Arnoldi method [Arnoldi, 1951], belonging to a class of Krylov subspace iterative methods, for performing a Hessenberg reduction.

Definition 1.5.2 (Krylov Subspaces)**.** Given a matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ and a non-zero vector $\mathbf{x}_0 \in \mathbb{R}^n$, the k^{th}-Krylov subspace, $\mathcal{K}_n(\mathbf{A}, \mathbf{x}_0, k)$ is defined by,

$$\mathcal{K}_n(\mathbf{A}, \mathbf{x}_0, k) = \text{span}\{\mathbf{x}_0, \mathbf{A}\mathbf{x}_0, \mathbf{A}^2\mathbf{x}_0, \mathbf{A}^3\mathbf{x}_0, ..., \mathbf{A}^{k-1}\mathbf{x}_0\}. \tag{1.85}$$

Definition 1.5.3 (Hessenberg reduction)**.** The Hessenberg reduction is a matrix decomposition technique commonly used for the computing eigenpairs of matrices. Given a unsymmetric matrix $A \in \mathbb{R}^{N\times N}$ (we assume that $\mathbf{A}$ is real for simplicity), we seek a decomposition of the form,

$$\mathbf{A} = \mathbf{Q}\mathbf{H}\mathbf{Q}^T, \tag{1.86}$$

where,

- $\mathbf{H} \in \mathbb{R}^{N\times N}$ is an upper Hessenberg matrix (i.e. $a_{i,j} = 0$ for $i > j + 1$)
- $\mathbf{Q} \in \mathbb{R}^{N\times N}$ is an orthonormal matrix (i.e. $\mathbf{Q}^{-1} = \mathbf{Q}^T$), whose columns $\mathbf{q}_1, ..., \mathbf{q}_N$, form an orthonormal basis.

The Hessenberg reduction shows that $\mathbf{A}$ and $\mathbf{H}$ are similar matrices, which have the same eigenvalues. If $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$, using $\mathbf{Q}^T = \mathbf{Q}^{-1}$ and multiplying (1.86) by $\mathbf{x}$,

$$\mathbf{A}\mathbf{x} = \mathbf{Q}\mathbf{H}\mathbf{Q}^{-1}\mathbf{x} \quad \Rightarrow \quad \lambda\mathbf{x} = \mathbf{Q}\mathbf{H}\mathbf{Q}^{-1}\mathbf{x} \quad \Rightarrow \quad \lambda\mathbf{Q}^{-1}\mathbf{x} = \mathbf{H}\mathbf{Q}^{-1}\mathbf{x} \quad \Rightarrow \quad \lambda\mathbf{y} = \mathbf{H}\mathbf{y}. \tag{1.87}$$

Hence, $\lambda(\mathbf{A}) = \lambda(\mathbf{H})$, and their eigenvectors are related by $\mathbf{x} = \mathbf{Q}\mathbf{y}$.

The Arnoldi method generates a sequences of vectors $[\mathbf{u}_0, \mathcal{A}\mathbf{u}_0, .., \mathcal{A}^{k-1}\mathbf{u}_0]$ that spans the k-dimensional Krylov subspace. These vectors, are known as Arnoldi vectors [Golub and Van Loan, 2013], are used to construct an orthogonal matrix via the Gram-Schmidt process, $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_k]$ $\in \mathbb{R}^{M\times K}$. This is equivalent to performing a partial Hessenberg reduction of $\mathcal{A} = \mathbf{Q}\mathbf{H}\mathbf{Q}^T$, where the eigenvalues of $\mathcal{A} \in \mathbb{R}^{N\times N}$ can be approximated by a smaller Hessenberg matrix $\mathbf{H} \in \mathbb{R}^{k\times k}$, suitable for a direct eigenvalue computation using the QR algorithm, The $k-$step Arnoldi factorisation of $\mathcal{A}$ gives,

$$\mathcal{A}\mathbf{Q}_k = \mathbf{Q}_k\mathbf{H}_k + \mathbf{r}_k\mathbf{e}_k^T, \tag{1.88}$$

where $\mathbf{H} \in \mathbb{R}^{k\times k}$ refers to the upper Hessenberg matrix, $\mathbf{e}_k = [0, ..., 0, 1] \in \mathbb{R}^k$, and $\mathbf{r}_k \in \mathbb{R}^N$ is a residual vector. If $\mathbf{x} = \mathbf{Q}_k\mathbf{y}$, and $\mathbf{H}\mathbf{y} = \lambda\mathbf{y}$ then,

$$(\mathcal{A} - \mathbf{I}\lambda)\mathbf{x} = (\mathbf{e}_k^T\mathbf{y})\mathbf{r}_k. \tag{1.89}$$

In other words, the residual vector difference between the approximation of $\lambda(\mathcal{A})$, using $\lambda(\mathbf{H})$. If $||\mathbf{r}_k|| = 0$, then $\lambda(\mathbf{H}) \subseteq \lambda(\mathcal{A})$.

We now present the Arnoldi method by generating k Arnoldi vectors,

$$\mathbf{T}_k = [\mathbf{u}_0, \mathbf{u}_1, ..., \mathbf{u}_{k-1}] = \left[\mathbf{u}_0, \frac{\mathcal{A}(T, Re)\mathbf{u}_0}{\alpha_1}, \frac{\mathcal{A}(T, Re)\mathbf{u}_1}{\alpha_2}, ..., \frac{\mathcal{A}(T, Re)\mathbf{u}_{k-1}}{\alpha_k}\right], \tag{1.90}$$

where α_j is scaled such that $||\mathbf{u}_j|| = 1$. Following [Barkley et al., 2008], the projection of $\mathcal{A}$ onto the Krylov subspace is given as,

$$\mathcal{A}\mathbf{T}_k = \mathbf{T}_{k+1} D_k^{(k+1)}, \tag{1.91}$$

where $D_k^{(k+1)} \in \mathbb{R}^{(k+1)\times k}$ is a shifted diagonal matrix with entries $D_{ij} = \alpha_i \delta_{i,j+1}$. We assume that $\mathbf{T}_k$ and $\mathbf{T}_{k+1}$ admit QR decompositions,

$$\mathcal{A}\mathbf{Q}_k\mathbf{R}_k = \mathbf{Q}_{k+1}\mathbf{R}_{k+1}\mathbf{D}_k^{(k+1)}, \tag{1.92}$$

where $\mathbf{Q}_k \in \mathbb{R}^{N\times k}$, $\mathbf{R}_k \in \mathbb{R}^{k\times k}$ and $\mathbf{Q}_{k+1}, \mathbf{R}_{k+1}$ are similarly defined. The upper Hessenberg matrix $\mathbf{H}_k^{(k+1)} \in \mathbb{R}^{(k+1)\times k}$ is defined as,

$$\mathbf{H}_k^{(k+1)} = \mathbf{R}_{k+1}\mathbf{D}_k^{(k+1)}\mathbf{R}_k^{-1}, \tag{1.93}$$

in which the last row of $\mathbf{H}_k^{(k+1)}$ only contains a single non-zero entry, $h^* = h_{k,k-1}$. By substituting the definition of the upper Hessenberg matrix and separating the last row of $\mathbf{H}_k^{(k+1)}$ we obtain,

$$\mathcal{A}\mathbf{Q}_k = \mathbf{Q}_k\mathbf{H}_k + h^*\mathbf{q}_k\mathbf{e}_k^T. \tag{1.94}$$

Equation (1.94) describes the projection of $\mathcal{A}$ onto the Krylov subspace spanned by orthornomal bases $\mathbf{Q}_k$, yielding a smaller $\mathbf{H}_k$ matrix. The accuracy of this approximation is dictated by the magnitude of the residual term, $h^*\mathbf{q}_k\mathbf{e}_k^T$. Assuming that $\mathbf{H}_k$ is diagonalisable as $\mathbf{H}_k = \mathbf{\Psi}_k\mathbf{\Lambda}_k\mathbf{\Psi}_k^{-1}$, we multiply equation (1.94) by $\mathbf{\Psi}_k$,

$$\mathcal{A}\mathbf{Q}_k\mathbf{\Psi}_k = \mathbf{Q}_k\mathbf{\Psi}_k\mathbf{\Psi}_k^{-1}\mathbf{H}_k\mathbf{\Psi}_k + h^*\mathbf{q}_k\mathbf{e}_k^T\mathbf{\Psi}_k. \tag{1.95}$$

Simplifying the expression above we get,

$$\mathcal{A}\mathbf{V}_k = \mathbf{V}_k\mathbf{\Lambda}_k + h^*\mathbf{q}_k\mathbf{e}_k^T\mathbf{\Psi}_k, \tag{1.96}$$

where $\mathbf{\Lambda}_k$ contains the k eigenvalues and $\mathbf{V}_k = \mathbf{Q}_k\mathbf{\Psi}_k$ the eigenvectors of $\mathcal{A}$. The error in approximating the j^{th} eigenpair is given by,

$$\varepsilon_j = ||\mathcal{A}\mathbf{v}_j - \lambda_j\mathbf{v}_j|| = ||h^*\mathbf{q}_k\mathbf{e}_k^T\psi_j|| = |h^*||\mathbf{v}_j[k-1]|, \tag{1.97}$$

where $\mathbf{v}_j[k-1]$ is the last component of the eigenvector $\mathbf{v}_j$.

Lastly, we are generally interested in obtaining the eigenpairs with the the largest real part. We introduce exponential power method [Tuckerman and Barkley, 2000], which is naturally considered by time stepping an initial perturbation $\mathbf{q}_0'$ from $t = 0$ to T,

$$\mathbf{q}'(T) = \exp(\mathcal{L}T)\mathbf{q}_0' = \mathcal{A}(T)\mathbf{q}_0'. \tag{1.98}$$

The dominant eigenvalues, μ, of $\mathcal{A}$ obtained from the Arnoldi method described above, which correspond to the eigenvalue of the largest real part λ, of $\mathcal{L}$ by $\mu = \exp(\lambda T)$, where T is typically set to 1. For further details on this algorithm, the reader is referred to Tuckerman and Barkley [2000] and

Barkley et al. [2008] for more details.

In summary, the algorithm described above have been implemented in Nektar++, referred to as the 'modified' Arnoldi algorithm, which modifies the existing time-stepper code with a wrapper function that generates Arnoldi vectors and solves the Hessenberg matrix using the subroutine `dgeev` from LAPACK (Linear Algebra PACKage, [Anderson et al., 1999]). The modified Arnoldi algorithm has been verified against using a separate implementation based on the third-party package ARPACK (ARnoldi PACKage [Lehoucq et al., 1998]) [Rocco, 2014].

1.5.2 Edge tracking

In this section, we consider the dynamical system perspective on the subscritial transitional shear flows where the linearly stable laminar co-exist with the chaoctic attractor of the turbulent dynamics. The edge is defined as the region that separates the basin of attraction between the laminar and turbulent state. Along this edge, sit local attractors, referred to as edge states which could could manifest as travelling-waves, tori, and high-order invariant sets. A common technique employed for edge tracking is often based on the bisection method [Skufca et al., 2006, Schneider et al., 2007, Khapko et al., 2016], which is performed by repeatedly bisecting the initial conditions given as,

$$\mathbf{q}_0(\mathbf{x}, 0; \chi) = \chi\mathbf{q}_L(\mathbf{x}) + (1-\chi)\mathbf{q}_T(\mathbf{x}) \tag{1.99}$$

where $\mathbf{q}_0$ refers to an initial condition consisting of a weighted sum of the bisection parameter, $\chi \in [0, 1]$, between a laminar state, $\mathbf{q}_L$, and a turbulent state, $\mathbf{q}_T$. For an initial condition given by $\mathbf{q}_0(\mathbf{x}, 0; \chi = 0) = \mathbf{q}_T$, or $\mathbf{q}_0(\mathbf{x}, 0; \chi = 1) = \mathbf{q}_L$ the solution trajectory will continue along the turbulent attractor or relaminarise respectively. Hence, there could be (at least) one intermediate value of the bisection parameter, $\chi^* \in [0, 1]$, where the solution trajectory evolves along the edge, neither falling towards the turbulent nor laminar attractor. The method of finding χ^* is based on the bisection, where we perform n successive bisections between χ_L^n, χ_T^n, where χ_L^n and χ_T^n leads to relaminarisation and turbulence respectively. This necessitates n direct numerical simulations using initial conditions updated by $\mathbf{q}_0(\mathbf{x}, 0; \chi^n = \frac{1}{2}(\chi_L^n + \chi_T^n))$ and having a stopping criteria to determine if the solution is about to relaminarise or become turbulent. A stopping criteria based on turbulent kinetic energy and wall shear stress is often used. This edge is often unstable and we often require χ^* to be considerably accurate to remain such that the initial condition could evolve along the edge for sufficient time, T. After we have determine χ^* and T, the bisection method can be restarted between by replacing the $\mathbf{q}_L(\mathbf{x}) = \mathbf{q}_0(\mathbf{x}, T, \chi_L)$ and $\mathbf{q}_T(\mathbf{x}) = \mathbf{q}_0(\mathbf{x}, T; \chi_T)$, which are close to each other in state space but diverge to the laminar and turbulent states after a long time $t > T$. After a certain number of r restarted bisections, the solution trajectory may convergence towards an attractor along the edge - an edge state, acting as a separatrtix betweent he turbulent and laminar attractor. We describe the algorithm of edge tracking in algorithm 1.5.2. To disambiguate between the iterative processes, we perform r number of restarted bisections in an 'outer-loop' which contains n number of bisections within an 'inner-loop'.

Algorithm 1 Algorithm for edge tracking between a turbulent and laminar state

```
Initialise maxBisects, maxRepeat                          ▷ Maximum n and repeated bisections
Initialise stopCriteria                                   ▷ Tolerance for stopping criteria (e.g., wall-shear stress)
r ← 0
while r < maxRepeat do                                    ▷ Iterating over $r$ repeated bisections
    if r == 0 then
        $\mathbf{q}_L, \mathbf{q}_T \leftarrow$ input()                        ▷ Initial laminar and turbulent states
    end if
    $\chi_L \leftarrow 0,\ \chi_T \leftarrow 1,\ \chi \leftarrow \frac{1}{2}(\chi_L + \chi_T)$     ▷ Initialise bisection coefficients
    $\mathbf{q}_0 \leftarrow \chi \mathbf{q}_T + (1 - \chi)\mathbf{q}_L$                  ▷ Initialise initial condition
    n ← 0
    while n < maxBisects do                               ▷ Iterating over $n$ bisections
        t ← 0, $\Delta \leftarrow 10^6$
        while $\Delta >$ stopCriteria do                  ▷ Assuming that the edge is unstable
            $\mathbf{q}_{t+1} \leftarrow$ TimeIntegrate($\mathbf{q}_t, \Delta t$)      ▷ Perform DNS for a single time-step
            $\Delta \leftarrow |\mathbf{q}_{t+1} - \mathbf{q}_0|$                ▷ Measuring the deviation from initial condition
            t++
        end while
        if isTurbulent($\mathbf{q}_t$) then                      ▷ Check if terminal state is turbulent
            $\chi_L \leftarrow \chi$                                     ▷ $\mathbf{q}_L$ gets a larger weight
            if n == maxBisects - 1 then
                $\mathbf{q}_T \leftarrow \mathbf{q}_t$                             ▷ Save turbulent-leaning initial condition
                break
            end if
        else
            $\chi_T \leftarrow \chi$
            if n == maxBisects - 1 then
                $\mathbf{q}_L \leftarrow \mathbf{q}_t$                             ▷ Save laminar-leaning initial condition
                break
            end if
        end if
        $\chi \leftarrow \frac{1}{2}(\chi_L + \chi_T)$
        $\mathbf{q}_0 \leftarrow \chi \mathbf{q}_L + (1 - \chi)\mathbf{q}_T$              ▷ Update initial conditions
        n++
    end while
    r++
end while
```

Bibliography

Edward Anderson, Zhaojun Bai, Christian Bischof, L Susan Blackford, James Demmel, Jack Dongarra, Jeremy Du Croz, Anne Greenbaum, Sven Hammarling, and Alan McKenney. *LAPACK users' guide*. SIAM, 1999. ISBN 0-89871-447-8.

Walter Edwin Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of applied mathematics*, 9(1):17–29, 1951. ISSN 0033-569X.

Ivo Babuška and Manil Suri. The p and h - p Versions of the Finite Element Method, Basic Principles and Properties. *SIAM Review*, 36(4):578–632, December 1994. ISSN 0036-1445, 1095-7200. doi: 10.1137/1036141. URL http://epubs.siam.org/doi/10.1137/1036141.

D. Barkley, H. M. Blackburn, and S. J. Sherwin. Direct optimal growth analysis for timesteppers. *International Journal for Numerical Methods in Fluids*, 57(9):1435–1458, July 2008. ISSN 0271-2091, 1097-0363. doi: 10.1002/fld.1824. URL https://onlinelibrary.wiley.com/doi/10.1002/fld.1824.

Alexandre Joel Chorin. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 2(1):12–26, August 1967. ISSN 00219991. doi: 10.1016/0021-9991(67)90037-X. URL https://linkinghub.elsevier.com/retrieve/pii/002199916790037X.

Douglas C. Chu and George Em Karniadakis. A direct numerical simulation of laminar and turbulent flow over riblet-mounted surfaces. *Journal of Fluid Mechanics*, 250:1–42, May 1993. ISSN 0022-1120, 1469-7645. doi: 10.1017/S0022112093001363. URL https://www.cambridge.org/core/product/identifier/S0022112093001363/type/journal_article.

R. B. Dean. Reynolds Number Dependence of Skin Friction and Other Bulk Flow Variables in Two-Dimensional Rectangular Duct Flow. *Journal of Fluids Engineering*, 100(2):215–223, June 1978. ISSN 0098-2202, 1528-901X. doi: 10.1115/1.3448633. URL https://asmedigitalcollection.asme.org/fluidsengineering/article/100/2/215/409704/Reynolds-Number-Dependence-of-Skin-Friction-and.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013. ISBN 1-4214-0859-7.

David Gottlieb and Steven A. Orszag. *Numerical Analysis of Spectral Methods: Theory and Applications*. Society for Industrial and Applied Mathematics, January 1977. ISBN 978-0-89871-023-

6 978-1-61197-042-5. doi: 10.1137/1.9781611970425. URL http://epubs.siam.org/doi/book/10.1137/1.9781611970425.

J. L. Guermond and Jie Shen. Velocity-Correction Projection Methods for Incompressible Flows. *SIAM Journal on Numerical Analysis*, 41(1):112–134, January 2003. ISSN 0036-1429, 1095-7170. doi: 10.1137/S0036142901395400. URL http://epubs.siam.org/doi/10.1137/S0036142901395400.

Ronald D. Henderson and Dwight Barkley. Secondary instability in the wake of a circular cylinder. *Physics of Fluids*, 8(6):1683–1685, June 1996. ISSN 1070-6631, 1089-7666. doi: 10.1063/1.868939. URL https://pubs.aip.org/pof/article/8/6/1683/260183/Secondary-instability-in-the-wake-of-a-circular.

Thorwald Herbert. PARABOLIZED STABILITY EQUATIONS. *Annual Review of Fluid Mechanics*, 29(1):245–283, January 1997. ISSN 0066-4189, 1545-4479. doi: 10.1146/annurev.fluid.29.1.245. URL https://www.annualreviews.org/doi/10.1146/annurev.fluid.29.1.245.

O. Iida and Y. Nagano. The Relaminarization Mechanisms of Turbulent Channel Flow at Low Reynolds Numbers. *Flow, Turbulence and Combustion*, 60(2):193–213, 1998. ISSN 13866184. doi: 10.1023/A:1009999606355. URL http://link.springer.com/10.1023/A:1009999606355.

George Karniadakis and Spencer J Sherwin. *Spectral/hp element methods for computational fluid dynamics*. Oxford University Press, 2005. ISBN 0-19-852869-8.

George Em Karniadakis, Moshe Israeli, and Steven A Orszag. High-order splitting methods for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 97(2): 414–443, December 1991. ISSN 00219991. doi: 10.1016/0021-9991(91)90007-8. URL https://linkinghub.elsevier.com/retrieve/pii/0021999191900078.

T. Khapko, T. Kreilos, P. Schlatter, Y. Duguet, B. Eckhardt, and D. S. Henningson. Edge states as mediators of bypass transition in boundary-layer flows. *Journal of Fluid Mechanics*, 801:R2, August 2016. ISSN 0022-1120, 1469-7645. doi: 10.1017/jfm.2016.434. URL https://www.cambridge.org/core/product/identifier/S0022112016004341/type/journal_article.

John Kim, Parviz Moin, and Robert Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics*, 177:133–166, April 1987. ISSN 0022-1120, 1469-7645. doi: 10.1017/S0022112087000892. URL https://www.cambridge.org/core/product/identifier/S0022112087000892/type/journal_article.

P. D. Lax and A. N. Milgram. IX. Parabolic Equations. In *Contributions to the Theory of Partial Differential Equations. (AM-33)*, pages 167–190. Princeton University Press, December 1955. ISBN 978-1-4008-8218-2. doi: 10.1515/9781400882182-010. URL https://www.degruyter.com/document/doi/10.1515/9781400882182-010/html.

Richard B Lehoucq, Danny C Sorensen, and Chao Yang. *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM, 1998. ISBN 0-89871-407-9.

Steven A. Orszag, Moshe Israeli, and Michel O. Deville. Boundary conditions for incompressible flows. *Journal of Scientific Computing*, 1(1):75–111, 1986. ISSN 0885-7474, 1573-7691. doi: 10.1007/BF01061454. URL http://link.springer.com/10.1007/BF01061454.

V. C. Patel and M. R. Head. Some observations on skin friction and velocity profiles in fully developed pipe and channel flows. *Journal of Fluid Mechanics*, 38(1):181–201, August 1969. ISSN 0022-1120, 1469-7645. doi: 10.1017/S0022112069000115. URL https://www.cambridge.org/core/product/identifier/S0022112069000115/type/journal_article.

Anthony T Patera. A spectral element method for fluid dynamics: Laminar flow in a channel expansion. *Journal of Computational Physics*, 54(3):468–488, June 1984. ISSN 00219991. doi: 10.1016/0021-9991(84)90128-1. URL https://linkinghub.elsevier.com/retrieve/pii/0021999184901281.

Gabriele Rocco. Advanced Instability Methods using Spectral/hp Discretisations and their Applications to Complex Geometries. 2014.

Tobias M. Schneider, Bruno Eckhardt, and James A. Yorke. Turbulence transition and the edge of chaos in pipe flow. *Physical Review Letters*, 99(3):034502, July 2007. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.99.034502. URL http://arxiv.org/abs/nlin/0703067. arXiv:nlin/0703067.

Joseph D. Skufca, James A. Yorke, and Bruno Eckhardt. Edge of Chaos in a Parallel Shear Flow. *Physical Review Letters*, 96(17):174101, May 2006. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.96.174101. URL https://link.aps.org/doi/10.1103/PhysRevLett.96.174101.

Gilbert Strang and George J. Fix. *An analysis of the finite element method*. Wellesley-Cambridge Press, Wellesley, Mass, 2. ed edition, 2008. ISBN 978-0-9802327-0-7.

Vassilios Theofilis. Advances in global linear instability analysis of nonparallel and three-dimensional flows. *Progress in Aerospace Sciences*, 39(4):249–315, May 2003. ISSN 03760421. doi: 10.1016/S0376-0421(02)00030-1. URL https://linkinghub.elsevier.com/retrieve/pii/S0376042102000301.

Takahiro Tsukahara, Kaoru Iwamoto, Hiroshi Kawamura, and Tetsuaki Takeda. DNS of heat transfer in a transitional channel flow accompanied by a turbulent puff-like structure, September 2014. URL http://arxiv.org/abs/1406.0586. arXiv:1406.0586 [physics].

Laurette S. Tuckerman and Dwight Barkley. Bifurcation Analysis for Timesteppers. In *Numerical Methods for Bifurcation Problems and Large-Scale Dynamical Systems*, volume 119, pages 453–466. Springer New York, New York, NY, 2000. ISBN 978-1-4612-7044-7 978-1-4612-1208-9. doi: 10.1007/978-1-4612-1208-9_20. URL http://link.springer.com/10.1007/978-1-4612-1208-9_20. Series Title: The IMA Volumes in Mathematics and its Applications.

R. Témam. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (II). *Archive for Rational Mechanics and Analysis*, 33(5):377–385, January 1969. ISSN 0003-9527, 1432-0673. doi: 10.1007/BF00247696. URL http://link.springer.com/10.1007/BF00247696.