Adaptive large-scale mantle convection simulations

Thesis submitted for the degree of

Doctor of Philosophy

at the University of Leicester

by

Samuel Peter Cox MMath (Oxon)

Department of Mathematics

University of Leicester

2017

Adaptive large-scale mantle convection simulations

ABSTRACT

Samuel P. Cox

The long-term motion of the Earth's mantle is of considerable interest to geologists and geodynamists in explaining the evolution of the planet and its internal and surface history. The inaccessible nature of the mantle necessitates the use of computer simulations to further our understanding of the processes underlying the motion of tectonic plates.

Numerical methods employed to solve the equations describing this motion lead to linear systems of a size which stretch the current capabilities of supercomputers to their limits. Progress towards the satisfactory simulation of this process is dependent upon the use of new mathematical and computational ideas in order to bring the largest problems within the reach of current computer architectures.

In this thesis we present an implementation of the discontinuous Galerkin method, coupled to a more traditional finite element method, for the simulation of this system. We also present an *a posteriori* error estimate for the convection-diffusion equation without reaction, using an exponential fitting technique and artificial reaction to relax the restrictions upon the derivative of the convection field that are usually imposed within the existing literature. This error bound is used as the basis of an h-adaptive mesh refinement strategy. We present an implementation of the calculation of this bound alongside the simulation and the indicator, in a parallelised C++ code, suitable for use in a distributed computing setting.

Finally, we present an implementation of the discontinuous Galerkin method into the community code ASPECT, along with an adaptivity indicator based upon the proven *a posteriori* error bound. We furnish both implementations with numerical examples to explore the applicability of these methods to a number of circumstances, with the aim of reducing the computational cost of large mantle convection simulations.

Acknowledgements

Firstly, I would like to thank my supervisors Dr. Tiffany Barry, Dr. Andrea Cangiani, and Prof. Emmanuil Georgoulis at the University of Leicester for their support and guidance throughout my time as their student. Their expertise in their respective areas has been invaluable, and their ideas have deepened my own understanding of a wide variety of subject matter, spanning mathematics, geology, and many other related areas.

I am eternally grateful to my wife, Naomi, for her love, support, kindness, and unfailing patience throughout my studies and beyond. She has lived with this PhD as much as I have, and has sacrificed her own wishes daily for mine. For that, I am greatly in her debt.

I would like to thank all of my family, including those I have gained through marriage, for their support. This is especially true of my parents, who gave me the best start in life, have always encouraged me to explore and succeed, and have offered assistance when I needed it.

I must also thank the members of the Department of Mathematics, particularly Scott Balchin, Peter (Zhaonan) Dong, and Oliver Sutton, who have been sources of mathematical genius and coffee in equal quantities; excellent company through almost continuous tea breaks; and a ready ear for complaints about broken code.

I would also like to thank the British Geological Survey, NERC, and the University of Leicester for their generous funding of my studies, and support in travel and equipment costs, without which my studies could never have happened, nor been such a great experience.

To the members of Jarvis Avenue and Westleigh Christadelphian Ecclesias, I offer my warm thanks for keeping me sane, and helping to make the Midlands feel like home. Thank you for your friendship, offering sage advice, being open to debate, and for helping me to focus upon what's truly important.

Finally, I thank God, with whom all things are possible, for all of these blessings which he has brought into my life.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 The Boussinesq system

The Boussinesq system of equations is the mostly widely used simplification of the mathematical model of the convective flow of the Earth's mantle. Under some simplifying assumptions to be examined later, the dynamics of the mantle temperature θ, velocity $\mathbf{u}$, and pressure p are governed by the following equations:

$$\begin{aligned} \theta_t - \varepsilon\Delta\theta + \mathbf{u}\cdot\nabla\theta &= f(\mathbf{x},t), \\ -\nabla\cdot(2\mu(\theta,\mathbf{x})\kappa(\mathbf{u})) + \nabla p &= -\rho(\theta,\mathbf{x})\mathbf{g}, \\ \nabla\cdot\mathbf{u} &= 0, \end{aligned}$$

where the symmetric gradient operator, viscosity, density and gravity are denoted by $\kappa(\cdot)$, μ, ρ and $\mathbf{g}$, respectively.

The numerical treatment of the Earth's mantle flow problem as described by these equations is difficult, particularly given the greatly varying parameter values, the existence of boundary and interior layers, the nonlinear dependencies, and the vastly differing scales upon which the constituent processes are set. In this work, we aim to extend the current state of knowledge as to how to solve these problems while making best use of computational resources. In particular, we aim to explore how

rigorous error estimation can be used to guide the resolution of computation within a simulation.

We are interested in the use of parallel numerical algorithms to solve this problem, and so we begin with an introduction to the hardware and software setting of such simulations.

1.2 An introduction to distributed-memory computing

In order to simulate increasingly large models within a reasonable period of time, computers are required to perform increasingly large numbers of calculations.

The simplest approach to satisfying this demand is to increase the clock speed of a single processor, which allows greater *flops* (floating-point operations per second). However, we are reaching the limits of engineering in terms of single processor speed. Additionally, an increase in clock speed translates into a cubic increase in power consumption [25], and thus vast energy requirements, which is neither a desirable nor sustainable state of affairs. As a consequence, processor clock rates have not increased in more than a decade.

Since a single-processor system is doomed by such flaws, it is established practise to use multiple processors working in parallel to increase the feasible model size. For small numbers of processors, up to perhaps 32, a reasonable approach is to allow all the processors to access a single pool of *shared memory*. However, beyond a small number of cores, the speed and efficiency of this approach quickly falls, for two reasons. Firstly, the speed of data access falls as the available storage increases, and the number of connections to the memory reaches physical limits due to size. These reduce the speed at which a single data storage can be physically accessed by multiple cores. Secondly, issues of data concurrency occur, with multiple processes attempting to access the same data simultaneously. As we are interested in a class of problems that are very large, this is not a reasonable approach, since we will need many processors to tackle this size of problems.

Instead, for the largest problems, we embrace a more scalable approach: that of a *distributed problem*. Under this paradigm, each processor has access only to a restricted portion of the problem. This reduces the need to broadcast large quantities of data, instead allowing us to localise portions of the code, and only communicate the bare minimum of information when necessary. However this does place additional requirements upon the data structures used, and complicates the process of solution.

Fortunately, in the setting of finite element simulation, implementations of these required data structures, and the necessary logic to co-ordinate the progress of the simulation in parallel, before combining the local solves into a global solution, exist in several software libraries. In particular, the deal.II library [15] builds upon the `p4est` algorithms [24], along with the Trilinos [53, 54] and PETSc [14, 13, 12] libraries to implement this infrastructure. Combining this with their abilities in thread-based parallelisation within processors, they provide the high-level structure for our implementation (we refer the reader to the references provided for a fuller understanding than the minimum presented in Chapter 5 of this thesis).

1.3 Discussion of previous numerical models in software, and their limitations

The study of numerical modelling of mantle convection began in the late 1960s and early 1970s, with 2D finite difference codes such as those of Minear and Toksöz [77], Torrance and Turcotte [102], Mckenzie et al. [74], and Schmeling and Jacoby [89]. These generally used the stream function formulation to eliminate the pressure from the Navier-Stokes equations and reduce 2D velocity vectors to scalars. More recent attempts to use finite differences have used staggered grids (e.g., [43]), equivalent to using a finite volume method.

Spectral methods were employed in mantle simulations as early as 1974 [112], and enjoyed much popularity during the 1980s and early 1990s for both 3D Cartesian and spherical geometries, due to their power in splitting a 3D problem into several 1D problems (e.g., [18, 100]), but they have largely fallen out of favour due to difficulties with handling large lateral heterogeneities in viscosity.

Finite volume methods enjoyed a lot of popularity from the early 1990s, and continue to be used, e.g., the Stag3D code of Tackley [99], but not to the same extent as finite element methods.

Finite element (FE) methods were first used in the early 1980s, often solving for a stream function, (e.g., [48]). Most FE codes now solve instead for the primary variables of temperature, velocity, and pressure. There are a growing number of codes that are well documented and have been widely used in the mantle convection modelling community, as well as a number of newer codes that are relevant to this work. We refer the interested reader to [73], and the references therein, for an excellent discussion of the history of the FE method and the use of mesh adaptivity in geodynamics. Below are presented details of a number of the most widely used of these codes.

1.3.1 TERRA

TERRA is an FE code developed by John Baumgardner in 1985 [16]. It uses a fixed mesh based on an icosahedral partition of the sphere, with each triangular face regularly subdivided a fixed number of times, to create a nearly uniform mesh on the surface of the sphere. This is then extended down into the bulk of the sphere a given distance and subdivided in that direction, to give a mesh covering the entire mantle shell. This was parallelised in 1995 using message passing and domain decomposition.

1.3.2 CitCom

Another very influential FE code, CitCom [80] was developed originally in the mid-1990s to solve 2D mantle convection problems. This was shown to be an easily extensible code, and was soon converted to solve on three-dimensional Cartesian domains in parallel. It spawned a large number of spin-offs, most notably CitComS [117], which solves in parallel on a fully spherical domain, using a multigrid algorithm, and Ellipsis3d [83] which uses the particle-in-cell approach to track particle flow through the mesh.

1.3.3 Fluidity

Fluidity [30] differs from most FE mantle convection codes in that it is based on an unstructured mesh; the concept of this method is to utilise h-adaptivity (that is, adaptive mesh refinement) as well as r-adaptivity (moving nodes of the mesh to follow the movement of material) to resolve the complex geometries of emerging structures within the mantle. It solves the incompressible Boussinesq equations (see Section 2.4) using the conjugate gradient method with algebraic multigrid (AMG) preconditioner for the solution of the Stokes system, with pressure correction introduced through a preconditioned FGMRES solver. The mesh is iteratively adapted to minimise a functional that represents the sub-optimality of the mesh.

1.3.4 Rhea

Rhea [23] is a highly parallel, 3D, octree-based h-adaptive FE code capable of resolution down to 1.5km locally, running on tens of thousands of cores with hundreds of millions of elements. It solves the Boussinesq system with space-and-temperature-dependent viscosity, using pressure-stabilisation for the Stokes system. This is solved using a preconditioned MINRES solver, with the preconditioner using an approximation to the Schur complement of a lumped mass matrix weighted by the inverse viscosity.

1.3.5 ASPECT

ASPECT [67] is a fully parallelised FE code for solving the Boussinesq equations. It is based on the deal.II [15] C++ library with the Trilinos [53, 54] library. The system is stabilised with a nonlinear artificial diffusion [47], and solved on an h-adaptive mesh using AMG to precondition the system. It is designed to be modular, using the deal.II, Trilinos, and `p4est` libraries to handle their specialised areas. The h-adaptivity is by default directed by use of the so-called Kelly error estimator [64], although other options are available. Refinement and coarsening take place at intervals within the simulation.

In summary, there are a number of software packages that model mantle convection numerically, and a wide range of techniques used. Earlier models relied upon fixed meshes to harness the computing power available at the time. More recently, a push for adaptive methods has allowed far greater accuracy with the same computational resources, along with improved solvers and preconditioners. Yet still there remains a large gap between the desired resolution and current abilities. Further improvements in adaptivity will form part of the further work required to bring the largest problems into the reach of our computational resources.

1.4 Proposed work

We propose the use of adaptive finite element methods driven by rigorous *a posteriori* error estimates in modelling mantle convection, with a view to reducing the computational cost, and thus helping to bring larger problems within the reach of current computing abilities. We will derive a new *a posteriori* error bound for a convection-diffusion equation without reaction, discretised by the discontinuous Galerkin method. The current literature does not cover the case of *a posteriori* error bounds for general steady-state flows without restrictions upon the divergence of the convection field. This, in turn, results in Gronwall-type exponential components in the resulting *a posteriori* error bounds for standard norms for the respective transient PDE problems. The new error bound does not rely on such restrictions, allowing it to be applied to situations of scientific interest, where we do not *a priori* know the behaviour of the flow. This is exactly the situation encountered in mantle convection simulation, since the temperature and velocity are tightly coupled in a nonlinear fashion. The practical aim of this work is to use this *a posteriori* error bound as the basis of an adaptivity indicator strategy. The use of an indicator designed for this problem should result in meshes which more accurately target areas of interest and computational difficulty than an *ad hoc* indicator, and thus improve the accuracy of simulation obtained for a given computational cost.

We present an implementation of the discontinuous Galerkin method for the convection-diffusion equation, coupled to a finite element Stokes flow solver, for use in exploring mantle convection problems. We use this to explore the behaviour of the

discontinuous Galerkin method and the associated error bound under a number of flow patterns, along with an indicator strategy based upon the error bound.

Finally, we present an implementation of the discontinuous Galerkin method in the community mantle convection simulation code ASPECT, along with an adaptivity indicator based upon the derived error bound, and discuss its use through a series of numerical examples.

1.5 Outline of this thesis

In Chapter 2, we present an outline of the current understanding of the motion of the Earth's mantle, and a derivation of the partial differential equations which govern the Boussinesq approximation to its properties and movement. We then report previous results showing the well-posedness of the underlying Stokes and convection-diffusion problems, in addition to well-posedness of the full coupled system.

In Chapter 3, we introduce the finite element and discontinuous Galerkin discretisation methods, and derive their formulations of the Stokes and convection-diffusion problems, respectively. We then comment upon the solution methods for the linear systems, and the use of adaptive mesh refinement to reduce the computational cost of such simulations.

In Chapter 4, we derive a new *a posteriori* error bound for the stationary convection-diffusion equation in the convection-dominated regime, without the traditional restrictions placed upon the divergence of the convection field. We then convert this into an *a posteriori* error bound upon the non-stationary convection-diffusion problem, with the same relaxation of restrictions.

In Chapter 5, we present an implementation of the discontinuous Galerkin method into a small mantle convection simulation code, based on a tutorial code from the deal.II library. This is coupled to a finite element Stokes solver. We present the implementation of an explicit computation of the error bound alongside the simulation, with particular emphasis placed upon the parallelisable nature of the implementation, demonstrating a novel auxiliary-mesh method to accurately calculate all the

required terms. Finally, we demonstrate the behaviour of the derived error bound through numerical examples.

In Chapter 6, we present the implementation into the community code ASPECT of the discontinuous Galerkin method for the temperature equation, and present numerical examples demonstrating its use. Finally, we also explore the use of an adaptivity indicator based upon the error estimate derived in Chapter 4.

In Chapter 7, we discuss conclusions of the work undertaken and areas for further exploration and research.

Chapter 2

The Boussinesq model of mantle convection

In this chapter, we motivate the use of mathematical modelling to understand mantle convection, and survey the current state of knowledge about the mechanism of mantle convection. We then introduce the Boussinesq model for mantle convection, and discussing its well-posedness.

2.1 Geological knowledge

The deepest man-made hole on land, the Kola superdeep borehole, reached a depth of 12,262m in 1994 before drilling was halted, while the deepest oceanic hole (drilled on the Tiber Oil Field in the Gulf of Mexico) reaches 10,680m from the seabed, or 11,939m from sea level. Meanwhile the lowest known natural point is Challenger Deep in the Mariana Trench at 10,911m. By comparison, the mean radius of the Earth is approximately 6,371km, meaning that the greatest depth man has reached is something less than 0.2% of the distance to the centre. This demonstrates the inaccessible nature of the interior of the Earth, which means that currently (and for the foreseeable future) the only practical ways to study it are by indirect methods, such as remote sensing (e.g., seismological, magnetic and gravitational readings); studying rocks that have come from the interior (through volcanism, uplift, or other

Region	Depth (km)	Radius (km)	Pressure range (GPa)
Crust	0–24	6347–6371	0–0.60
Upper mantle	24–670	5701–6347	0.60–24
Lower mantle	670–2891	3480–5701	24–136
Outer core	2891–5150	1221–3480	136–329
Inner core	5150–6371	0–1221	329–364

TABLE 2.1: A simplified version of the PREM model

processes); studying meteorite compositions and extraterrestrial planets and moons; and mathematical modelling.

The interior of the Earth is by no means homogeneous. Variations occur at different latitudes and longitudes, but the largest heterogeneities are dependent upon depth. To this end, the standard models of the Earth interior are based upon changes at prescribed depths. There are several levels of model for the Earth. The most basic level consists of a core of radius 3480km, surrounded by a layer of mantle 2810km thick, with a crust of depth 80km on top. A more in-depth model, based on the PREM model of Dziewonski & Anderson [33], is presented in a modified form in Table 2.1. This PREM model is based on inversion of astronomic-geodetic data and seismic wave data, coupled with knowledge of the positions of seismic discontinuities based on many other previous works.

In this discussion, it must be understood that there are variations in the depth profiles beneath different points on the surface, and thus estimates of thickness vary dependent on individual preference, and boundaries between layers often occur over perhaps tens of kilometres or more. Thus, defining transition depths is bound to admit a range of plausible values. However, the simplified PREM scheme is a useful initial reference setting. Figure 2.1 illustrates the PREM scheme with additional information near the surface.

At the surface of the Earth lies the crust, within which there are distinct differences between oceanic crust and continental crust – oceanic crust is denser, and thinner, at between 5 and 10km thick; continental crust is lighter, and varies between 20 and 70km in thickness, typically 30-45km, with the thickest parts beneath young

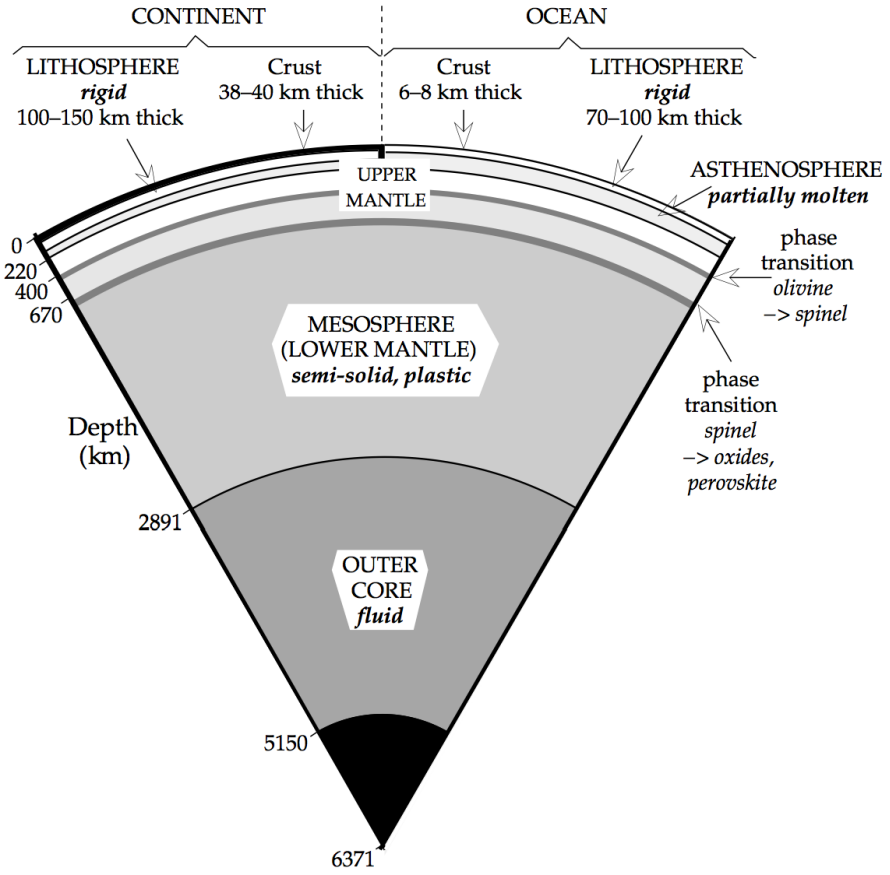


FIGURE 2.1: Illustration of the PREM model, demonstrating the differences between the oceanic and continental lithosphere [70].

mountain ranges. Crust varies greatly in composition, but, on average, continental crust is silica-rich while oceanic crust is basaltic and contains more iron- and magnesium-based minerals.

Beneath the crust lies the lithospheric mantle. The transition from crust to lithospheric mantle is known as the Mohorovičić discontinuity (Moho), generally defined as the layer where P-wave (pressure wave) velocities exceed 7.6km/s. This can be a sharp boundary, but can also be a more gradual transition, probably up to a thickness of about 2km. Below the lithospheric mantle is the upper mantle, which extends to around 670km. The upper mantle is primarily made of magnesium silicate, largely in the form of olivine. Below this is the lower mantle, which reaches to a depth of 2891km. The separation at 670km is due to a distinct discontinuity in seismic velocities measured globally at that depth. It is a matter of current debate to what extent the 670 discontinuity prevents the transfer of material between upper

and lower mantle, with different results and disciplines suggesting different regimes of mixing.

Other important definitions are those of the lithosphere and asthenosphere. The lithosphere is the mechanically strong, uppermost layer of the Earth. Extending down to around 80km depth, it comprises the entirety of the crust and the lithospheric mantle. Within the lithosphere, heat transport is primarily through conduction. Beneath the lithosphere, the asthenosphere is the mechanically weak material between the depths of about 80km and 200km. In the asthenosphere and below, convection is the dominant method of heat transport. Note that the lithosphere and asthenosphere definitions refer to the mechanical properties of the layers, while the core-mantle-crust definitions refer to the composition of the material.

Both upper and lower mantle contain a number of distinct layers with different seismic properties. For example the layer known as D″ is an approximately 150km thick layer directly above the core-mantle boundary showing distinct seismic velocity behaviours in distinct lateral portions. The size of the areas over which the velocities vary in D″ are on a similar scale to the size of today's tectonic plates and continents, suggesting that the remains of previous crustal plates may lie there. Other examples of distinct layers include the Lehmann seismic discontinuity at approximately 200km (which is not found everywhere), and the 410km discontinuity, which is believed to result from a change of crystal lattice structure within the material.

Beneath the mantle is the core. Meteorite composition studies, gravity calculations, and the presence of Earth's magnetosphere indicate that this is made primarily of iron, with perhaps up to 10% nickel and other dense elements, and a few other, lighter elements present, such as silicon, sulphur and oxygen. It is not known whether there may be radioactive elements present in the core at high enough levels to contribute significantly to heat generation [70, p. 197].

The core has 2 distinct zones: the inner core, with a radius of 1220km, at a depth of 5150km, is solid, while surrounding this is the outer core which is liquid (this is rendered necessary to account for effects in tide motion, and for the presence of the magnetosphere, in addition to being indicated by a lack of S-wave propagation). This outer core is around 2260km thick, between the depths of 2891km and 5150km, and is thought to be nearly homogeneous in composition. Pressures at the core-mantle

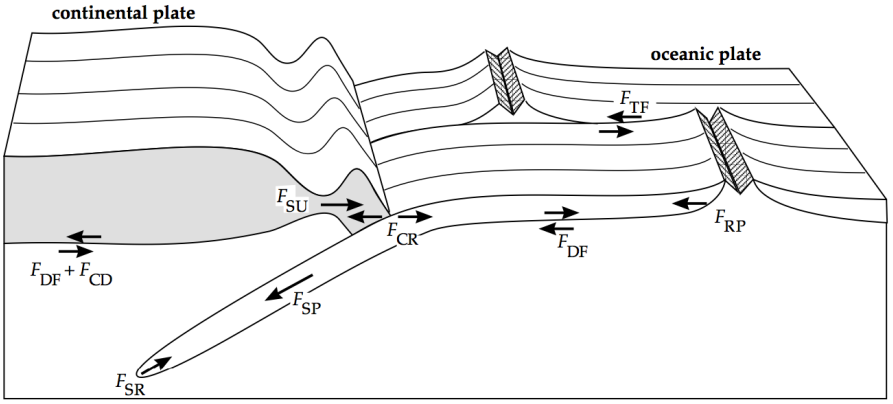


FIGURE 2.2: Example geometry demonstrating the different fault types possible between plates [70]. Force labels are explained in the text.

boundary (CMB, also known as the Gutenberg discontinuity) are estimated at 136 GPa, while at the centre of the Earth the pressure is around 364 GPa.[39, p. 371]

Variation in viscosity is one of the most important features in the structure of the mantle. However, it is very difficult to calculate a unique viscosity profile based upon experimental evidence, since the evidence suggests lateral variation in viscosity in addition to vertical variation. (see Section 2.2). Temperature is also linked to both viscosity and depth, but temperature profiles are ill-constrained, due to an incomplete understanding of the formation of the planet and its subsequent history, as well as uncertainties in the parameter values and inter-dependencies of temperature, viscosity, buoyancy and diffusion.

The lithosphere is split into a number of high-strength tectonic plates, separated by low-strength fault zones; these allow the plates to move independently of each other. This means that there are three basic types of fault mechanism between plates: spreading, where two plates are moving apart; subduction, where two colliding plates force one plate to sink beneath the other; and transform, where the two plates slide past each other. In the case of a continental plate colliding with an oceanic plate, the oceanic plate's higher density will force it beneath the less dense continental crust.

To demonstrate these fault types, an example geometry is used in Figure 2.2. This shows an oceanic plate being subducted beneath a continental plate, with a third plate on the right moving away, creating a spreading ridge. This is a fairly common

setup in reality, found for example off the coast of Japan. The forces acting upon the subducting plate are as follows: *mantle drag force* (F_{DF}) results from the difference in flow rates of the mantle and the plate. Depending on the relative velocities, this force can act towards or away from the subduction zone. The *collision resistance force* (F_{CR}) is imparted by the resistance between the two plates. The *ridge push force* (F_{RP}) is the result of upwelling material under the spreading ridge spreading sideways, forcing the plate sideways. There is another resistive force, in the form of the *slab resistance force* (F_{SR}) acting upon the leading edge of the subducting slab. Finally, there is the *slab pull force* (F_{SP}) which, if the slab remains connected to the plate, pulls the plate down. This force is caused by the difference between the densities of the slab, which is made of cold, dense crust, and the surrounding material, which is warmer and less dense. It is essentially then the slab's own increased density that drags it down. Current thinking is that in most cases this slab pull is the main driving force behind the movement of subducting plates, as opposed to the typical view that the mantle pushes the plates, the so-called 'crustal conveyor belt' mechanism. However, it is still an open question as to how this explanation allows for the initial development of plate tectonics.

In addition to these three plates, the fourth plate in the background induces (and is subjected to) a *transform force* (F_{TF}) as it slides parallel to the subducting plate. The plate on top of the subducting plate also experiences a *trench suction* (F_{SU}) as the weight of the subducting plate pulls it away from the upper plate, effectively pulling the upper plate towards the trench.

Earthquakes commonly occur near to transform and subduction faults, being generated by the buildup and sudden release of energy as neighbouring plates slip past each other after being held in position by friction. Volcanoes are also often located around 150 km inland from subducting faults. This is believed to result from the dehydration of the descending slab; the increasing pressure forces water out of the slab, reducing the melting temperature of the material directly above it. This results in upwellings of less viscous material, manifesting themselves as volcanoes at the surface.

Oceanic plate is largely generated at spreading faults; as two plates separate, new, less viscous material rises to fill the gap, and solidifies to form new oceanic crust. The fact that oceanic crust subducts beneath continental crust means that continental

crust is far older than the majority of oceanic crust; the oldest oceanic crust is currently dated at around 160Mya, while the oldest continental crust is dated at 4Ga old, and the cratons (the stable parts of the continental lithosphere that largely form the basis of today's continents) probably formed in the period 3.5 - 2.7 Gya [70, p. 596].

In the mantle, material is transported by buoyancy differences between different temperatures of rock: cool, dense material sinks down, while hotter, less dense material is more buoyant and therefore rises. This is the primary transporter of both material and heat: thermal conduction is orders of magnitude slower than advection of the hot material itself. Debate still continues as to how strongly coupled the mantle and plates are. The general consensus is that the plates are not as strongly coupled as previously thought, and that it is the plates themselves that generate the majority of their motion.

Plates typically move relative to each other at an average of something less than 5cm per year, which demonstrates the range of lengthscales and timescales to be considered – movement per year is on the order of centimetres, while the mantle itself is on the order thousands of kilometres. Thus, the timescale of years is not really suitable – models will have to consider timescales much longer than this.

Given the inaccessible nature of the interior of the Earth as demonstrated above, the uncertainties in the remote measurements and subsequent inversions of data, and the fact that the processes being studied occur over far longer timescales than the study itself, modelling is a useful tool to test possible explanations, and to simulate processes faster than they occur and can be measured by humans. A brief introduction to the previous and ongoing work undertaken towards modelling the interior is presented in Section 2.2.

2.2 Geological modelling history and literature review

Some of the earliest work on modelling of the physical properties of the mantle was done by Haskell in three papers between 1935 and 1937 [49, 50, 51]. This was

theoretical work based on understanding the process of continental uplift after ice sheets melt and drain from continents. This work gave an estimate of the average viscosity over the top 1000-1200km as roughly 10^{21} Pascal-seconds (1 Pascal-second = 1 kg/m/s is the viscosity of a fluid placed between two plates one metre apart which requires a shear stress of one pascal on one plate to move the plate sideways at 1 m/s).

Unfortunately, several mis-applications of this result, and confusion over the resolving power of various authors' data, led to this average value only being attributed to the mantle above the 670km discontinuity. This misinterpretation skewed several later models, as there is in reality a large jump in viscosity moving from the upper to lower mantle [79] (thus attributing the result up to just the 670km level overestimated the result for the upper mantle viscosity). Since then, there have been many more attempts to invert various data sets to give new estimated radial viscosity profiles based on various techniques, such as isostatic adjustment and seismic tomography, e.g., [29, 62, 78, 75], as well as analysis on the ability to infer layers from available data [84]. A reasonable estimate at this time would be a viscosity of $10^{23} - 10^{25}$ Pa $\cdot$ s at a depth of 2000km, reducing to 10^{21} Pa $\cdot$ s at 700km depth. The volume around the 670km discontinuity has a reduced viscosity of perhaps 10^{19} Pa $\cdot$ s. Above this discontinuity, the viscosity is again at least 10^{21} Pa $\cdot$ s rising as high as perhaps 10^{23} or 10^{24} Pa $\cdot$ s in the crust. However, these figures and profiles are highly uncertain, and the available data may not even be able to resolve the viscosity profile into any more than 2 layers. So the choice of viscosity profile is a large uncertainty in any model.

Alongside the attempts to decode the viscosity profile of the mantle, there has been much work undertaken attempting to model convection within the mantle.

2.3 Derivation of mantle equations

The basic model of the motion of the mantle is as one of an extremely viscous fluid. We take the approach of modelling the entire system by means of the interaction of three primary quantities: temperature, velocity, and pressure. These are supplemented by relations prescribing the parameters governing the flow, such as viscosity,

density, and diffusivity. In this section, we state the equations that describe the motion of the mantle from this viewpoint: conservation of momentum, conservation of mass, and conservation of energy. We largely follow the approach presented in [91], while another useful resource with additional details is [38].

2.3.1 Conservation of mass

The conservation of mass requires that the change of mass in a volume is equal to the quantity of mass flowing into or out of the volume. This is written

$$\rho_t + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{2.3.1}$$

where $\mathbf{u}$ is the velocity, ρ the density, and the subscript t denotes the time derivative. In the case of an incompressible fluid, there is no change in density in time or space, and then (2.3.1) reduces to

$$\nabla \cdot \mathbf{u} = 0.$$

2.3.2 Conservation of momentum

The conservation of momentum, for a fluid that is Newtonian and isotropic, is embodied in the identity

$$\rho \left(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}\right) - \nabla \cdot \left(\mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^\intercal\right) + \left(k_B - \frac{2}{3}\mu\right) \nabla \cdot \mathbf{u}\right) = -\nabla p + \rho \mathbf{g}, \tag{2.3.2}$$

where $\mathbf{g}$ is the gravity vector, p pressure, μ viscosity, and k_B is the bulk viscosity of the material.

2.3.3 Conservation of energy

By combining the second law of thermodynamics with the notion that energy must be conserved, it can be shown (e.g., [91, Chapter 6]) that, assuming again that the

fluid is Newtonian,

$$\rho C_p \left(\theta_t + \mathbf{u} \cdot \nabla\theta\right) - \alpha\theta \left(p_t + \mathbf{u} \cdot \nabla p\right) \\ = \nabla \cdot (k\nabla\theta) + k_B \left(\nabla \cdot \mathbf{u}\right)^2 + 2\mu \left(\frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^\intercal\right) - \frac{1}{3}\nabla \cdot \mathbf{u}\right)^2 + \rho f, \quad (2.3.3)$$

where θ is the temperature, C_p is the specific heat, α the thermal expansivity, k the thermal conductivity, k_B the bulk viscosity as above, and f the internal heating term.

2.4 Assumptions and Boussinesq approximation

In order to present our analysis, we simplify the three equations (2.3.1), (2.3.2), (2.3.3) by using the so-called Boussinesq approximation [21]. This greatly reduces the complexity of the equations, while also retaining the most important behaviours. The rigorous analysis of this approximation was developed in [94, 76], showing that the approximation is valid in the case where $\alpha\theta \ll 1$ and $\frac{\varepsilon^2}{C_p\theta} \ll L^2$, where L is a typical lengthscale of the system. For the case of mantle convection, these quantities are as indicated, ensuring that the Boussinesq approximation is valid for our purposes (we comment briefly upon this later within this section).

The Boussinesq approximation stems from the following assumptions:

- density is predominantly dependent on thermal effects rather than pressure-based effects, meaning that we may neglect the latter;
- we may take density to be constant, except in the terms where it modifies gravity to give a buoyancy term.

By taking these assumptions, we may infer the following.

In the conservation of mass, by the product rule

$$0 = \rho_t + \nabla \cdot (\rho\mathbf{u}) = \rho_t + \mathbf{u} \cdot \nabla\rho + \rho\nabla \cdot \mathbf{u}.$$

Then taking the density ρ to be constant in this equation, we see that

$$\nabla \cdot \mathbf{u} = 0. \tag{2.4.1}$$

Thus the equation of conservation of mass collapses, in the Boussinesq approximation, to the requirement for a divergence-free velocity field. This is the *incompressibility assumption*.

Applying the same approximation to the left hand side of the equation for conservation of momentum (2.3.2), and combining this with the divergence-free quality of $\mathbf{u}$ from above, we have

$$\rho\left(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}\right) - \nabla \cdot \left(\mu\left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\intercal}\right)\right) = -\nabla p + \rho \mathbf{g}.$$

By nondimensionalising this equation around a reference state, we see that the leftmost quantity $\rho\left(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}\right)$ is premultiplied by the reciprocal of the dimensionless Prandtl number $Pr = \mu_0 / \left(\rho_0 \varepsilon_0\right)$, where subscript 0 denotes a reference value. This quantity is a characteristic of the fluid itself, and denotes the ratio between momentum diffusivity and thermal diffusivity. In the mantle the Prandtl number is in the order of 10^{23}, ensuring that this term is utterly negligible. Thus, we have

$$-\nabla \cdot \left(\mu\left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\intercal}\right)\right) = -\nabla p + \rho \mathbf{g}.$$

(This can alternatively be justified by considering that the ratio of inertial forces to viscous forces is very small, with the same result.)

Finally, simplifying (2.3.3) by assuming $\alpha\theta \ll 1$ and a divergence-free velocity, along with a constant density,

$$C_p\left(\theta_t + \mathbf{u} \cdot \nabla \theta\right) = \nabla \cdot \left(\frac{k}{\rho} \nabla \theta\right) + 2\frac{\mu}{\rho}\left(\frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\intercal}\right)\right)^2 + f.$$

Taking C_p constant, we neglect the viscous heating term since it scales like Ra^{-1}, where Ra is the Rayleigh number of the fluid. In the mantle, this has a value around 10^{-6}, and so discarding this term should not have too significant an effect. Doing

so reduces the equation to the classical convection-diffusion equation

$$\theta_t + \mathbf{u} \cdot \nabla\theta - \nabla \cdot (\varepsilon \nabla\theta) = f,$$

where ε is the diffusion coefficient. We refer to [94, 76] for the analysis of the applicability of this approximation, and refer again to [38, 91] for an explanation of the derivation via nondimensionalisation.

We comment here briefly upon the reasonableness of the assumption that density is constant throughout the mantle. The density is in fact expected to increase by around 50% between the surface and base of the mantle. However, given that the lengthscale of any variation in pressure is likely to be large, in the absence of sharp discontinuities due to phase changes, the effect of this approximation (namely, incompressibility as in (2.4.1)) is generally deemed reasonable.

Finally, we note here that, while these approximations greatly reduce the complexity of the equations we study, they do not overly restrict the regimes of flow that we wish to explore. Indeed, there are a number of material models that fit into this framework, and can be studied in this way through the variable viscosity, density, and heating terms and their possible dependence upon the primary fields of temperature, velocity and pressure.

We are now able to rigorously state the full system of equations that will be analysed in the following chapters.

2.5 The Boussinesq system

Let $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$, be a closed, bounded domain. We denote its boundary by Γ, which we split into two subsets, Γ_D and Γ_N, with $\Gamma_D \cup \Gamma_N = \Gamma$. Let $I = [0, T] \subset \mathbb{R}$, $T > 0$, be a time interval. The problem that we will consider is the following: given an initial temperature field $\theta_0(\mathbf{x})$ and time- and position-dependent forcing term

$f(\mathbf{x},t)$, find the triple $(\theta(\mathbf{x},t),\mathbf{u}(\mathbf{x},t),p(\mathbf{x},t))$ such that

$$\left.\begin{aligned}\theta_t - \varepsilon\Delta\theta + \mathbf{u}\cdot\nabla\theta &= f(\mathbf{x},t)\\ -\nabla\cdot(2\mu(\theta,\mathbf{x})\kappa(\mathbf{u})) + \nabla p &= -\rho(\theta,\mathbf{x})\mathbf{g}\\ \nabla\cdot\mathbf{u} &= 0\end{aligned}\right\}\ \text{in } \Omega\times I,$$

$$\begin{aligned}\theta(\mathbf{x},0) &= \theta_0(\mathbf{x}) && \text{in } \Omega,\\ \theta &= g_D(\mathbf{x},t) && \text{on } \Gamma_D\times I,\\ \varepsilon\frac{\partial\theta}{\partial\mathbf{n}} &= g_N(\mathbf{x},t) && \text{on } \Gamma_N\times I,\end{aligned}$$

$$\left.\begin{aligned}\mathbf{u}\cdot\mathbf{n} &= 0\\ \kappa(\mathbf{u})\mathbf{n}\times\mathbf{n} &= 0\end{aligned}\right\}\ \text{in } \Gamma\times I, \tag{2.5.1}$$

where $\theta(\mathbf{x},t)$ corresponds to temperature, $\mathbf{u}(\mathbf{x},t)$ velocity, and $p(\mathbf{x},t)$ pressure. The first equation is the energy equation; the second and third form the Stokes system. The thermal diffusion, which is constant, is denoted by ε, while $\kappa(\mathbf{u})$ is the symmetric gradient operator, $\kappa(\mathbf{u}) \coloneqq \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^\intercal)$, and $\mathbf{g}$ the gravity vector. In our models, we use $\mathbf{g} = 9.81e_r$, where e_r is the radial unit vector (in the case of annular or shell geometries) or the unit downwards vector (in a box geometry). The temperature- and position-dependent viscosity is denoted by $\mu(\theta,\mathbf{x})$ and density by $\rho(\theta,\mathbf{x})$. The precise functional analytic setting for the solution and problem data is discussed below, after the necessary definitions are introduced.

2.6 Some spaces and notation

In order to discuss the existence of solutions to the system (2.5.1) and the convection-diffusion and Stokes systems, as well as describing the finite element methods proposed in this work, we first briefly recollect standard definitions of Lebesgue and Sobolev spaces (see, for example, [1]).

Definition 2.1. Let $p \in \mathbb{R}\cup\{+\infty\}$ and ω a subset of $\mathbb{R}^d$. The L^p-norm is the functional

$$\|v\|_{\omega,p} \coloneqq \left(\int_\omega v(\mathbf{x})^p \,\mathrm{d}x\right)^{\frac{1}{p}},\ \text{for } 1\le p<\infty,$$

$$\|v\|_{\omega,\infty} \coloneqq \operatorname*{ess\,sup}_{\mathbf{x}\in\omega} v(\mathbf{x}).$$

The Lebesgue space $L^p(\omega)$ is then defined as the space of functions

$$L^p(\omega) \coloneqq \{v : \omega \to \mathbb{R} \text{ such that } \|v\|_{\omega,p} < \infty\}.$$

In particular, the space $L^2(\omega)$ is a Hilbert space when equipped with the inner product

$$(w, v)_\omega \coloneqq \int_\omega wv \, \mathrm{d}x.$$

We suppress the subscript ω when it is referring to the whole domain Ω.

In the $p = 2$ case, we also suppress the subscript p notation for the norm where it is unlikely to cause confusion.

Definition 2.2. Let $p \in \mathbb{R} \cup \{+\infty\}$, s be a non-negative integer, and ω be an open subset of $\mathbb{R}^d$. Also let α be a multi-index of dimension d. The Sobolev space $W^{s,p}(\omega)$ is defined as the set of functions

$$W^{s,p}(\omega) \coloneqq \{v \in L^p(\omega) : D^\alpha v \in L^p(\omega) \text{ for } 0 \le |\alpha| \le s\},$$

equipped with the norm

$$\|v\|_{\omega,W^{s,p}} \coloneqq \left(\sum_{0\le|\alpha|\le s} \|D^\alpha v\|^p_{\omega,p} \right)^{1/p}, \text{ for } 1 \le p < \infty,$$
$$\|v\|_{\omega,W^{s,\infty}} \coloneqq \max_{0\le|\alpha|\le s} \|D^\alpha v\|_{\omega,\infty},$$

where $D^\alpha v$ is the weak (or distributional) derivative of v of order α, and $\|v\|_{p,\omega}$ the L^p-norm on ω.

We also define the family of semi-norms $|\cdot|_{\omega,W^{s,p}}$:

$$|v|_{\omega,W^{s,p}} \coloneqq \left(\sum_{|\alpha|=s} \|D^\alpha v\|^p_{\omega,p} \right)^{1/p}, \text{ for } 1 \le p < \infty,$$
$$|v|_{\omega,W^{s,\infty}} \coloneqq \max_{|\alpha|=s} \|D^\alpha v\|_{\omega,\infty}.$$

In particular, since $W^{k,2}(\omega)$ is a Hilbert space, we use the shorthand $H^k(\omega) \coloneqq W^{k,2}(\omega)$ for any integer $k \geq 1$.

We also define the shorthand

$$H_0^1(\omega) \coloneqq \left\{ v \in H^1(\omega) : v|_\Gamma = 0 \right\},$$

$$H_D^1(\omega) \coloneqq \left\{ v \in H^1(\omega) : v|_{\Gamma_D} = 0 \right\},$$

where Γ_D is the portion of the boundary of domain ω that is subject to a Dirichlet boundary condition.

We define $C^0(\omega)$ to be the set of continuous functions over the domain ω.

For $1 \leq p \leq \infty$ we use the standard notation $L^p(0,T;X)$ to denote the Bochner space of functions which are p-integrable over the interval $(0,T)$ with values in a Banach space X.

Finally, we define $C(0,T;X)$ to be the set of continuous mappings of the interval $[0,T]$ into X.

2.7 The convection-diffusion problem

With a view to analyse the numerical discretisation, we first analyse the energy equation separately from the Stokes system, which together form the Boussinesq system (2.5.1).

We consider the closed, and bounded, domain $\Omega \subset \mathbb{R}^d$, $d \in \{2,3\}$ as before. Let $\mathbf{u}(\mathbf{x},t) = (u_1,\ldots,u_d)^\intercal \in [C(0,T;W^{1,\infty}(\Omega))]^d$ (and hence $\nabla \cdot \mathbf{u} \in L^\infty(0,T;L^\infty(\Omega))$) be given. We decompose the boundary Γ into two parts, Γ_D and Γ_N, such that $\Gamma_D \cap \Gamma_N = \emptyset$, $\Gamma_D \cup \Gamma_N = \Gamma$, and make the assumption that $\mathbf{u}(\mathbf{x},t) \cdot \mathbf{n}(\mathbf{x}) = 0$ for $(\mathbf{x},t)$ in $\Gamma_N \times I$, where $\mathbf{n}(\mathbf{x})$ is the outward normal from the boundary at point $\mathbf{x}$.

We consider the convection-diffusion initial-boundary value problem:

$$\theta_t - \varepsilon\Delta\theta + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta = f(\mathbf{x},t) \qquad \text{on } \Omega\times I, \tag{2.7.1}$$

$$\theta = g_D(\mathbf{x},t) \qquad \text{on } \Gamma_D\times I, \tag{2.7.2}$$

$$\varepsilon\frac{\partial\theta}{\partial\mathbf{n}} = g_N(\mathbf{x},t) \qquad \text{on } \Gamma_N\times I, \tag{2.7.3}$$

$$\theta(\mathbf{x},0) = \theta_0(\mathbf{x}) \qquad \text{on } \Omega. \tag{2.7.4}$$

Here, we take ε to be a small positive constant, $0<\varepsilon\ll 1$, $f(\mathbf{x},t)\in L^2(0,T;L^2(\Omega))$, and $\theta_0(\mathbf{x})\in L^2(\Omega)$. We also assume $g_D\in H^1(0,T;H^{\frac{1}{2}}(\Gamma))$.

In the following analysis, we make use of the solution to a related problem with an additional reaction term. To this end, we also consider the convection-diffusion-reaction equation

$$\theta_t - \varepsilon\Delta\theta + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta = f(\mathbf{x},t) \qquad \text{on } \Omega\times I, \tag{2.7.5}$$

where $b\in L^\infty(\Omega)$ is a reaction coefficient. This is supplemented with the same boundary conditions (2.7.2), (2.7.3) and initial condition (2.7.4).

We now derive the weak formulation of the convection-diffusion and convection-diffusion-reaction equations.

Taking the equation (2.7.5) we can multiply all terms by a function $v\in H^1(\Omega)$ and integrate over Ω to give

$$\int_\Omega \left(\theta_t v - \varepsilon\Delta\theta v + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta v + b(\mathbf{x},t)\theta v\right)\,\mathrm{d}x = \int_\Omega f(\mathbf{x})v\,\mathrm{d}x,$$

and apply the divergence theorem to the second term, to give

$$\int_\Omega \left(\theta_t v + \varepsilon\nabla\theta\cdot\nabla v + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta v + b(\mathbf{x},t)\theta v\right)\,\mathrm{d}x - \int_\Gamma \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right)v\,\mathrm{d}s = \int_\Omega f(\mathbf{x})v\,\mathrm{d}x. \tag{2.7.6}$$

Since we would like our solution to live in a closed solution space, we impose the Neumann boundary condition weakly, so that

$$\int_\Omega (\theta_t v + \varepsilon \nabla\theta \cdot \nabla v + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta v + b(\mathbf{x},t)\theta v)\ \mathrm{d}x = \int_\Omega f(\mathbf{x})v\ \mathrm{d}x + \int_{\Gamma_N} g_N v\ \mathrm{d}s.$$

We define the bilinear form $a(\theta, v)$ by

$$a(\theta, v) = (\varepsilon\nabla\theta, \nabla v) + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta, v),$$

and the linear functional $l(v)$ by

$$l(v) = \int_\Omega f(\mathbf{x})v\ \mathrm{d}x + \int_{\Gamma_N} g_N v\ \mathrm{d}s.$$

Then the *weak formulation* of the problem (2.7.1)-(2.7.4) reads: for each $t \in I$, find $\theta(t) \in H^1(\Omega)$ such that

$$\left.\begin{aligned} (\theta_t, v) + a(\theta, v) &= l(v), \\ \theta|_{\Gamma_D} &= g_D, \\ \theta(\mathbf{x}, 0) &= \theta_0(\mathbf{x}), \end{aligned}\right\} \tag{2.7.7}$$

for all $v \in H^1(\Omega)$.

We can in identical fashion derive the weak formulation of the convection-diffusion-reaction problem (2.7.2)–(2.7.5): for each $t \in I$ find $\theta(t) \in H^1(\Omega)$ such that

$$\left.\begin{aligned} (\theta_t, v) + a_{\mathrm{reac}}(\theta, v) &= l(v), \\ \theta|_{\Gamma_D} &= g_D, \\ \theta(\mathbf{x}, 0) &= \theta_0(\mathbf{x}), \end{aligned}\right\}$$

for all $v \in H^1(\Omega)$, where

$$a_{\mathrm{reac}}(\theta, v) = (\varepsilon\nabla\theta, \nabla v) + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta, v) + (b(\mathbf{x},t)\theta, v). \tag{2.7.8}$$

The existence and uniqueness of a solution to the problem (2.7.1)–(2.7.4) (and equivalently, (2.7.7)) on a spherical domain $\Omega = \{\mathbf{x} \in \Omega : R_1 < |\mathbf{x}| < R_2\}$ is shown in [97, Lemma 2], which is reproduced below.

Lemma 2.3 (Well-posedness of convection-diffusion problem)**.** *Let* $\Omega = \{\mathbf{x} \in \Omega : R_1 < |\mathbf{x}| < R_2\}$ *and suppose that* $f \in L^2(0,T;H^{-1}(\Omega))$*,* $g_D \in H^1(0,T;H^{\frac{1}{2}}(\Gamma))$*,* $\mathbf{u} \in L^2(0,T;[L^3(\Omega))^3]$*,* $\nabla \cdot \mathbf{u} \in L^2(0,T;L^3(\Omega))$*, and* $\theta_0 \in L^2(\Omega)$*. Then there exists a unique solution* $\theta \in L^2(0,T;H^1(\Omega)) \cap L^\infty(0,T;L^2(\Omega))$ *to the system* (2.7.1)*,* (2.7.2)*,* (2.7.4)*.*

As the assumptions on the data regularity will be stronger than those in Lemma 2.3 (these are needed below), this can be applied directly to our situation, along with a standard treatment of Neumann boundary conditions, and we have the existence and uniqueness of a solution $\theta \in L^2(0,T;H^1(\Omega)) \cap L^\infty(0,T;L^2(\Omega))$ to the problem (2.7.1)–(2.7.4).

2.8 The Stokes system

Having discussed the well-posedness for the convection-diffusion equation on a spherical domain, we now consider the associated stationary Stokes system in the Boussinesq system (2.5.1) at every time $t \in I$: given a temperature field $\theta \in L^2(\Omega)$, viscosity $\mu(\theta,\cdot) \in L^\infty(\Omega)$ with a minimum $\mu(\theta,\cdot) \geq \underline{\mu} > 0$ and density term $\rho(\theta,\cdot) \in L^2(\Omega)$, find the pair $(\mathbf{u}(\mathbf{x}), p(\mathbf{x}))$ such that

$$\left.\begin{aligned} -\nabla \cdot (2\mu(\theta,\mathbf{x})\,\kappa(\mathbf{u})) + \nabla p &= -\rho(\theta,\mathbf{x})\mathbf{g} \\ \nabla \cdot \mathbf{u} &= 0 \end{aligned}\right\} \text{ in } \Omega \times I, \tag{2.8.1}$$

$$\left.\begin{aligned} \mathbf{u} \cdot \mathbf{n} &= 0 \\ \kappa(\mathbf{u})\mathbf{n} \times \mathbf{n} &= 0 \end{aligned}\right\} \text{ in } \Gamma \times I, \tag{2.8.2}$$

where $\mathbf{u}(\mathbf{x})$ is the velocity, and $p(\mathbf{x})$ the pressure. The gravity vector is denoted by $\mathbf{g}$, and $\kappa(\mathbf{u})$ is the symmetric gradient operator.

The system (2.8.1)–(2.8.2) does not necessarily admit a unique solution, for example in the case of a thick-shell domain. Indeed, in this case, defining the three rigid body motions $\mathbf{v}^{(i)}, i = 1,2,3$ by $\mathbf{v}^{(i)}(\mathbf{x}) := \mathbf{e}^{(i)} \times \mathbf{x}$ where $\mathbf{e}^{(i)}$ is the unit vector in the i-th coordinate direction and $(\mathbf{u}(\mathbf{x}), p(\mathbf{x}))$ a solution at time $t \in I$, gives us that $\mathbf{u} + \sum_{i=1}^3 c_i \mathbf{v}^{(i)}$ is also a solution, for $c_i \in \mathbb{R}$. In addition, the pressure solution is only unique up to an additive constant.

In the case of a thick-shell domain, to circumvent this, we can introduce three natural spaces for this problem:

$$\begin{aligned} W &\coloneqq \left\{ \mathbf{w} \in \left[H^1(\Omega)\right]^3 : \mathbf{w} \cdot \mathbf{n} = 0 \text{ on } \Gamma \right\}, \\ U &\coloneqq \left\{ \mathbf{w} \in W : (\mathbf{w}, \mathbf{v}^{(i)}) = 0 \text{ for } i = 1, 2, 3 \right\}, \\ Q &\coloneqq \left\{ q \in L^2(\Omega) : (q, 1) = 0 \right\}. \end{aligned}$$

We define the norms on U and Q by

$$\begin{aligned} \|\mathbf{v}\|_U &\coloneqq \|\mathbf{v}\|_{H^1}, \\ \|q\|_Q &\coloneqq \|q\|. \end{aligned}$$

Our solution will be non-unique if we look for $\mathbf{u} \in W$, but eliminating the degrees of freedom associated to the rigid body motions allows a unique solution in U. Other domains may be handled in a similar way if they are rotationally invariant under rigid-body motions. Similarly, the condition $(q, 1) = 0$ for $q \in Q$ removes the freedom to add a constant for the pressure field.

In standard fashion, we define

$$\int_\Omega \kappa(\mathbf{w}) : \kappa(\mathbf{v}) \, \mathrm{d}x \coloneqq \sum_{i,j=1}^{d} \int_\Omega \frac{\partial w_i}{\partial x_j} \frac{\partial v_i}{\partial x_j} \, \mathrm{d}x.$$

Then the weak formulation of this problem is derived in the following manner.

Multiplying (2.8.1) by a test function $\mathbf{v} \in U$ and $q \in Q$, and integrating over the domain, we have

$$\begin{aligned} \int_\Omega -\nabla \cdot (2\mu(\theta, \mathbf{x}) \kappa(\mathbf{u})) \mathbf{v} \, \mathrm{d}x + \int_\Omega \nabla p \cdot \mathbf{v} \, \mathrm{d}x &= \int_\Omega -\rho(\theta, \mathbf{x}) \mathbf{g} \cdot \mathbf{v} \, \mathrm{d}x, \qquad (2.8.3) \\ \int_\Omega q \nabla \cdot \mathbf{u} \, \mathrm{d}x &= 0. \end{aligned}$$

Using integration by parts on the left-hand side of (2.8.3) gives

$$\begin{aligned} (2\mu(\theta, \mathbf{x}) \kappa(\mathbf{u}), \kappa(\mathbf{v})) - (\nabla \cdot \mathbf{v}, p) &= (-\rho(\theta, \mathbf{x}) \mathbf{g}, \mathbf{v}), \\ -(\nabla \cdot \mathbf{u}, q) &= 0. \end{aligned}$$

We define the following bilinear forms:

$$\begin{aligned} s(\mathbf{u},\mathbf{v}) &:= (2\mu(\theta,\mathbf{x})\kappa(\mathbf{u}),\kappa(\mathbf{v})), \\ b(\mathbf{v},p) &:= -(\nabla\cdot\mathbf{v},p). \end{aligned} \tag{2.8.4}$$

The weak formulation of the Stokes equation is thus: find $\mathbf{u}\in U$, $p\in Q$ such that

$$\left.\begin{aligned} s(\mathbf{u},\mathbf{v})+b(\mathbf{v},p) &= -(\rho(\theta,\mathbf{x})\mathbf{g},\mathbf{v}) \\ b(\mathbf{u},q) &= 0 \end{aligned}\right\}, \tag{2.8.5}$$

for all $(\mathbf{v},q)\in U\times Q$.

It is well known [45, Corollary 4.1] that the choice of spaces for $\mathbf{u}$ and p affect the well-posedness of the problem. In the theory of traditional FE methods, the Lax-Milgram lemma shows stability of methods which contain bilinear functionals on the cartesian product of a Hilbert space with itself. In the mixed-method case, however, the bilinear functionals involved operate on the cartesian product of two different Hilbert spaces. The *inf-sup* condition encodes a sufficient condition for the extension of this lemma to the case of distinct Hilbert spaces. In our case, since $s(\cdot,\cdot)$ is coercive on U, and $b(\cdot,\cdot)$ does indeed satisfy the *inf-sup* condition

$$\inf_{q\in Q}\sup_{\mathbf{v}\in U}\frac{b(\mathbf{v},q)}{\|q\|_Q\|\mathbf{v}\|_U}>0,$$

(see [96] for proof of these two assertions) we satisfy the conditions of [45, Corollary 4.1], so that (2.8.5) is indeed well-posed.

Specifically, in our case we have the following result from [97, Lemma 1].

Lemma 2.4 (Well-posedness of stationary Stokes system)**.** *Let Ω be a spherical domain $\Omega = \{\mathbf{x}\in\Omega : R_1<|\mathbf{x}|<R_2\}$, and suppose that*

$$\begin{aligned} \rho(\theta,\mathbf{x})\mathbf{g}\in\left[L^2(\Omega)\right]^3, \\ \mu\in L^\infty(\Omega), \\ \mu(\theta,\mathbf{x})\geq\underline{\mu}>0. \end{aligned}$$

Then there exists a unique solution to (2.8.1), (2.8.2) *in $U\times Q$.*

2.9 Well-posedness of full problem

Finally, in view of showing well-posedness of the full Boussinesq problem, we introduce the weak form of the full system (2.5.1): for each $t \in I$, find $(\theta, \mathbf{u}, p) \in H^1(\Omega) \times U \times Q$ such that

$$\left.\begin{aligned} (\theta_t, v) + a(\theta, v) &= l(v) \\ s(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) &= -(\rho(\theta, \mathbf{x})\mathbf{g}, \mathbf{v}) \\ b(\mathbf{u}, q) &= 0 \\ \theta|_{\Gamma_D} &= g_D \\ \theta(\mathbf{x}, 0) &= \theta^0(\mathbf{x}) \end{aligned}\right\}, \tag{2.9.1}$$

for all $(v, \mathbf{v}, q) \in H^1(\Omega) \times U \times Q$, where we note the implicit dependence of the bilinear form $a(\cdot, \cdot)$ upon the convection variable $\mathbf{u}(\mathbf{x}, t)$.

Once again, [97, Theorem 3] shows the well-posedness of this system on a spherical domain, under certain conditions. We use the notation $\mathrm{cl}(\Omega)$ to denote the closure of Ω.

Lemma 2.5. *Suppose* $\mu : cl(\Omega) \times \mathbb{R} \to (0, +\infty)$ *and*

$$f \in L^\infty(0, T; L^\infty(\Omega)), \qquad \theta_0 \in L^\infty(\Omega),$$
$$g_D \in H^1(0, T; H^{\frac{1}{2}}(\Gamma)) \cap L^\infty(0, T; L^\infty(\Gamma)).$$

Then there exists a solution $(\theta, \mathbf{u}, p)$ *of* (2.9.1)*,*

$$\mathbf{u} \in L^\infty(0, T; \left[H^1(\Omega)\right]^3), \qquad p \in L^\infty(0, T; L^2(\Omega)),$$
$$\theta \in L^2(0, T; H^1(\Omega)) \cap L^\infty(0, T; L^\infty(\Omega)),$$

and, if

$$\mathbf{u} \in L^\infty(0, T; \left[W^{1,\infty}(\Omega)\right]^3),$$

then the solution is unique.

Since we have stronger conditions in place, we can apply this to our situation, along with a standard treatment of the Neumann boundary conditions, and claim that (2.9.1) is well-posed.

Chapter 3

Numerical models and methods

In this chapter, we introduce the symmetric interior penalty discontinuous Galerkin method for the energy equation, along with the finite element method for the Stokes system, before combining these into a single solution scheme. This includes discussion of the merits and drawbacks of such a scheme.

We then introduce some of the related aspects of solving the numerical problem, with details on the structure of the associated linear systems, and their handling and solution. Finally we discuss the topic of adaptive mesh refinement, and its role in enabling the solution of large problems.

3.1 Why dG?

The finite element method is a well-established numerical framework for the approximate solution of partial differential equations, just like those encountered in (2.5.1). It converts an infinite-dimensional problem to one posed in a finite-dimensional space, allowing for a numerical solution on a computer or computer cluster. The method, in its most classic flavour, is characterised by first decomposing the domain of interest into a finite number of non-overlapping cells which cover the domain. This so-called meshing of the domain is then used to define a space of admissible solutions that are (mapped) polynomial over each cell. Next, a linear system is constructed with reference to the partial differential equation of interest and to the

space of admissible solutions. Finally, this linear system is solved by a direct or iterative method to arrive at an approximate solution from the admissible space.

Over the preceding decades, a large number of techniques have been developed under the umbrella of the 'finite element method'. These are usually developed to exhibit certain desirable qualities, which are of benefit when seeking solutions to a given class of problems. One such technique, which will be introduced in the next section, is the *discontinuous Galerkin method*. This removes the constraint of continuity across mesh cell interfaces as is usually employed in the finite element method, and instead allows the solution to be *discontinuous* over the mesh interfaces. Instead, continuity is imposed weakly within the discrete weak form: different dG approaches are obtained depending on how this is done [8].

Before introducing in detail the particular discontinuous Galerkin (dG) method considered here by means of its application to the convection-diffusion equation (2.7.2)–(2.7.5), we discuss some of the merits and disadvantages of the method.

The discontinuous Galerkin method was originally introduced [11, 32, 7, 111] to capitalise on the flexibility afforded by decoupling neighbouring cells. Firstly, the mesh requirements can be relaxed from those found in conforming methods. In particular, the presence of 'hanging nodes' (nodes occurring when a cell on one side of a mesh interface is subdivided into smaller cells by splitting of such interface, but the neighbouring cell is not) is handled naturally, without technical requirements or restrictions. This has the advantage of seamless interpretation of stable numerical fluxes, enhancing the stability of the method. Secondly, the choice of polynomial approximation space is increased: cells may vary in polynomial degree across the mesh (e.g., through p-refinement, where p is the polynomial order), and if desired the polynomial space can be defined with respect to the physical frame of reference, rather than requiring mapping back and forth between the physical frame and a reference frame. This also enables the extension to meshes containing general polygons, which is of particular benefit when considering problems on intricate or heterogeneous domains. For further details of the use of discontinuous methods and polygons in an *hp* setting, see [26] and the references therein.

The main benefits of interest exploited in the current work are: ease in mesh adaptivity, and the stability of the method when modelling convection-dominated problems.

It is well known within the numerical modelling community that the standard finite element method (just like the centred finite difference method) suffers from the phenomenon of spurious oscillations when solving convection-diffusion problems in the convection-dominated regime. This was initially treated with the addition of artificial diffusion [109], before the technique was refined to add diffusion only in the direction of the streamlines [59, 65]. Following this, the method known as streamline upwind Petrov-Galerkin (SUPG) [58] enhanced the capability to solve the convection-dominated problem. Since then, numerous techniques have been proposed to stabilise FE, such as artificial viscosity and entropy viscosity [47], to mention just two.

On the other hand, the discontinuous Galerkin method, with carefully chosen numerical fluxes, is not affected by the oscillatory behaviour at boundary layers or shocks. This means no additional stabilisation term is required on top of the natural stabilising effect embedded in the numerical fluxes.

In the high-order method setting, a benefit of the discontinuous Galerkin (dG) method is the locality of its stencils. Degrees of freedom (DoFs) within one cell communicate directly with two classes of DoFs: those within the same cell, and those on the faces of neighbouring cells. This reduces the distance across the mesh which is closely coupled in the linear system, and thus eases the iterative solution process.

In the same way, at least in three dimensions, the use of dG also translates into a matrix with fewer non-zero elements per row. FEM couples cells to each of their neighbours that share at least a vertex, while dG couples only to the 6 cells (in 3 dimensions, assuming hexahedral cells are used) which share a face with the current cell. This greatly reduces the number of non-zero elements on each row of the matrix. In turn, this reduces the memory requirements when storing the sparse matrix, and is of benefit when attempting to solve the linear system.

The two main disadvantages of the dG method are its increased number of DoFs, and the expense in computing the associated linear system entries. The first comes from the fact that DoFs at vertices and on faces are no longer shared between cells. Instead, both cells sharing a face, and all cells around a vertex, contain a separate DoF. The multiplicity of DoFs at some points can increase the number of DoFs

substantially, and thus the method is often viewed as being expensive, particularly as it does not by itself reduce the discretisation error of the solution. However, recent work [26] has shown that, at least on tensor-product meshes, dG methods employing polynomials of *total degree* p are able to compete in computational efficiency with FEM for higher degrees p, while retaining the convergence properties of the dG method with tensor-polynomial basis functions. The second disadvantage occurs due to the complexity of the weak form involved with the method: a number of extra terms related to the numerical fluxes are introduced, which must be assembled. This requires the infrastructural ability to integrate over mesh faces and the programming ability and investment to implement the correct algorithm to combine all the terms.

The *a posteriori* error analysis of the method is fairly mature for a number of classes of problem. In the case of pure diffusion, there are many estimators for a selection of finite element discretisations [105, 3], and for dG methods in particular [17, 60, 56]. Stationary linear convection-diffusion equations have been analysed in [106, 68, 108, 88, 90, 35, 118]. The literature also contains many non-stationary convection-diffusion a posteriori error estimators [57, 20, 5, 6, 34, 107, 95, 40, 71] with dG methods for pure diffusion considered in [36, 42, 93], and the dG convection-diffusion case treated in [27].

The combination of these characteristics justifies the implementation and analysis of such a scheme as worthwhile, in order to develop a more rigorously supported error indicator for mesh adaptation.

We additionally note here that in the dG convection-diffusion case above, some strong assumptions are placed upon the convection field. In the case of no reaction, the convective field must have a non-positive divergence. While we solve the Boussinesq problem here, which satisfies a zero divergence condition on the velocity field, we have to overcome the difficulties raised by only solving for a divergence-free field approximately. Thus, when we come to solve the convection-diffusion equation, we do so using a convective field that is only approximately divergence-free. This means we are in need of an error estimate that is applicable to the case where the divergence of the velocity is small but positive. We note that the use of alternative mantle convection models incorporating compressible flow is becoming increasingly

widespread [113, 98, 114, 82, 99, 69]. Consequently, the development of a result handling weaker assumptions upon the convection field provides the prospect of future application to these models also.

Thus, we look to extend the current techniques to develop a new error estimate, and to then implement this as the basis of a practical error indicator for the purpose of adaptive mesh refinement.

In particular, we study the case where a discontinuous Galerkin method for the temperature equation is coupled with a conforming finite element method for the Stokes problem. This may be considered as a stepping-stone towards the use of the dG method to discretise the Stokes problem *also*. The Stokes problem would benefit particularly from the increased sparsity of the matrices generated, given that the solution of the Stokes system with strongly varying viscosity is still a very difficult problem and, in our experience, can take up to 90% of the runtime for the complete simulation. Thus, improvements in the solution of the Stokes system are bound to have the greatest effect on the runtime of the largest models we wish to consider. Nevertheless, the scope of the current work is focused more on the derivation of a rigorous error indicator to more accurately detect where to focus the given computational resources.

In the following sections we derive a dG method for the temperature equation and a FEM for the Stokes equation.

3.2 Semi-discrete dG formulation for the energy equation

We first consider a semi-discrete method for the approximation of the temperature component θ of the problem (2.7.2)–(2.7.5) whereby we discretise solely in the space variable. As such, we consider the domain Ω to be a either a polygon in 2D or a polyhedron in 3D, and define a mesh $\mathcal{T}_h$. In two dimensions, $\mathcal{T}_h$ is a collection of open, triangular or quadrilateral cells K that subdivide the domain Ω. In three dimensions, $\mathcal{T}_h$ is a collection of open, tetrahedral or hexahedral cells that similarly subdivide Ω; that is,

1. $\bigcup_{K\in\mathcal{T}_h} \mathrm{cl}\,(K) = \mathrm{cl}\,(\Omega)$, and
2. $K_i \cap K_j = \emptyset$ for all pairs of cells $K_i, K_j \in \mathcal{T}_h$, $i \neq j$,

where we use the notation $\mathrm{cl}\,(\omega)$, $\omega \subset \mathbb{R}^d$ to denote the closure of ω, with $d \in \{2, 3\}$. For each $K \in \mathcal{T}_h$, we denote the boundary of the cell by $\partial K \coloneqq \mathrm{cl}\,(K) \setminus K$. For each pair of cells $K, K' \in \mathcal{T}_h$, we say the cells are vertex-neighbours if $\mathrm{cl}\,(K) \cap \mathrm{cl}\,(K') \neq \emptyset$, and define their interface to be a face (we remark that, in 3D, faces are 2-dimensional intersections between neighbouring cells, while 1D intersections of cells are known as edges). We denote by $\mathcal{F}_h$ the collection of all such $(d-1)$-dimensional faces defined by the interfaces between cells. We also define the set of interior faces $\mathcal{F}_I$ and set of faces on the boundary $\mathcal{F}_B$. Thus we have that $\mathcal{F}_h = \mathcal{F}_I \cup \mathcal{F}_B$ and we define the boundary of the domain as $\Gamma = \bigcup_{F\in\mathcal{F}_B} F$. We also subdivide $\mathcal{F}_B$ into faces on the Dirichlet boundary $\mathcal{F}_D$ and faces on the Neumann boundary $\mathcal{F}_N$, with $\mathcal{F}_D \cup \mathcal{F}_N = \mathcal{F}_B$ and $\mathcal{F}_D \cap \mathcal{F}_N = \emptyset$.

We denote by h_F the $(d-1)$-dimensional measure of the face F, and by h_K the d-dimensional measure of the cell K.

Let $\hat{K}$ be the reference simplex in the case where the mesh $\mathcal{T}_h$ is made up of simplices, or the reference hypercube in the case $\mathcal{T}_h$ is made up of quadrilaterals or hexahedra. We say that a polynomial has *total degree* m where m is the maximum over all terms in the polynomial of the sum of the exponents of the variables in that term. Let $\mathcal{P}_k(\hat{K})$, $k \geq 0$, be the space of polynomials of *total* degree less than or equal to k in the variables $x_1, \ldots, x_d$ on the reference cell $\hat{K}$, while $\mathcal{Q}_k(\hat{K})$ is the space of polynomials of degree less than or equal to k in *each* variable $x_1, \ldots, x_d$ on $\hat{K}$.

Let $\mathcal{D}_K : \hat{K} \to K$ be a smooth map with non-singular Jacobian. Then we can define the space of piecewise-polynomial functions $X_{h,k}$ in the following way: if our mesh $\mathcal{T}_h$ is composed of simplicial cells, then

$$X_{h,k} \coloneqq \left\{ v_h \in L^2(\Omega) : v_h|_K \circ \mathcal{D}_K \in \mathcal{P}_k(\hat{K}) \;\forall\; K \in \mathcal{T}_h \right\}, \tag{3.2.1}$$

while, if the mesh is composed of quadrilateral or hexahedral cells, then

$$X_{h,k} \coloneqq \left\{ v_h \in L^2(\Omega) : v_h|_K \circ \mathcal{D}_K \in \mathcal{Q}_k(\hat{K}) \;\forall\; K \in \mathcal{T}_h \right\}. \tag{3.2.2}$$

Remark 3.1. We note here in passing that it is equally possible to apply the space $\mathcal{P}_k$ to the case of quadrilateral and hexahedral meshes. This has the added benefit of reducing the number of degrees of freedom per cell, and has been shown [26] to exhibit the same order of convergence as $\mathcal{Q}_k$. However, due to limitations in the current ability to implement these on non-fixed meshes, we do not use these here. Nonetheless, this could be a new direction of research in view of the potential complexity benefits.

Returning to the weak formulation (2.7.6), we can now split the domain into the cells of the mesh $\mathcal{T}_h$ and test against a function $v_h \in X_{h,k}$:

$$\sum_{K\in\mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon\nabla\theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta, v_h)_K \\ - \sum_{K\in\mathcal{T}_h} \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\partial K} = \sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v_h)_K,$$

where $\mathbf{n}$ is the outward normal to the cell boundary ∂K. We then split the cell-boundary terms into three sets: the set of internal faces we denote by $\mathcal{F}_I$, the faces within the Dirichlet boundary area Γ_D by $\mathcal{F}_D$, and those in the Neumann boundary areas Γ_N by $\mathcal{F}_N$:

$$\sum_{K\in\mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon\nabla\theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta, v_h)_K \\ - \sum_{K\in\mathcal{T}_h} \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\partial K\setminus\Gamma} - \sum_{K\in\mathcal{T}_h} \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\partial K\cap\Gamma_D} - \sum_{K\in\mathcal{T}_h} \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\partial K\cap\Gamma_N} \\ = \sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v_h)_K.$$

Combining boundary terms, we have

$$\sum_{K\in\mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon\nabla\theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta, v_h)_K \\ - \sum_{K\in\mathcal{T}_h} \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\partial K\setminus\Gamma} + \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\Gamma_D} + \langle \varepsilon\left(\nabla\theta\cdot\mathbf{n}\right), v_h\rangle_{\Gamma_N} \\ = \sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v_h)_K.$$

Since we know from (2.7.3) that $\varepsilon \frac{\partial \theta}{\partial \mathbf{n}} = \varepsilon \nabla \theta \cdot \mathbf{n} = g_N$ on Γ_N, we can substitute this in and move the term to the right hand side:

$$\sum_{K \in \mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon \nabla \theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x}, t) \cdot \nabla \theta + b(\mathbf{x}, t)\theta, v_h)_K \\ - \sum_{K \in \mathcal{T}_h} \langle \varepsilon (\nabla \theta \cdot \mathbf{n}), v_h \rangle_{\partial K \setminus \Gamma} + \langle \varepsilon (\nabla \theta \cdot \mathbf{n}), v_h \rangle_{\Gamma_D} \\ = \sum_{K \in \mathcal{T}_h} (f(\mathbf{x}, t), v_h)_K + \langle g_N, v_h \rangle_{\Gamma_N}.$$

We introduce the notation θ_K^+ to mean the internal trace of θ, for a given cell K, and θ_K^- the external trace. Each internal face $F \in \mathcal{F}_I$ (the set of internal faces) has two neighbouring cells, K and K', with outward normals $\mathbf{n}_K, \mathbf{n}_{K'}$ on the face F. Then the jumps over F for a scalar-valued function w and vector-valued function $\mathbf{w}$ are defined as

$$[\![w]\!]_F \coloneqq w_K^+ \mathbf{n}_K + w_{K'}^+ \mathbf{n}_{K'}, \qquad [\![\mathbf{w}]\!]_F \coloneqq \mathbf{w}_K^+ \cdot \mathbf{n}_K + \mathbf{w}_{K'}^+ \cdot \mathbf{n}_{K'}.$$

For faces on the Dirichlet portion of the boundary, we set

$$[\![w]\!]_F \coloneqq w_K^+ \mathbf{n}_K, \qquad [\![\mathbf{w}]\!]_F \coloneqq \mathbf{w}_K^+ \cdot \mathbf{n}_K,$$

while on the Neumann portion we set

$$[\![w]\!]_F \coloneqq \mathbf{0}, \qquad [\![\mathbf{w}]\!]_F \coloneqq 0.$$

We suppress the subscript when no confusion is likely. In the same way, we define the average values of w and $\mathbf{w}$ on the face $F \subset \partial K$ as

$$\{w\}_F \coloneqq \frac{1}{2}\left(w_K^+ + w_K^-\right), \qquad \{\mathbf{w}\}_F \coloneqq \frac{1}{2}\left(\mathbf{w}_K^+ + \mathbf{w}_K^-\right),$$

while on all boundary faces we define

$$\{w\}_F \coloneqq w_K^+, \qquad \{\mathbf{w}\}_F \coloneqq \mathbf{w}_K^+.$$

Again, we suppress the subscript in the notation when there is no risk of ambiguity.

Now, from elliptic regularity, $\varepsilon\nabla\theta \cdot \mathbf{n}$ is continuous on all internal faces [44, Cor 8.36]. Thus we can substitute this with the average of the two values from the two neighbouring cells, $\{\varepsilon\nabla\theta\}$, and rewrite this as

$$\sum_{K\in\mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon\nabla\theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta, v_h)_K \\ - \sum_{F\in\mathcal{F}_h} \langle \varepsilon \{\nabla\theta\}, [\![v_h]\!]\rangle_F = \sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v_h)_K + \langle g_N, v_h\rangle_{\Gamma_N}. \quad (3.2.3)$$

Rather than strongly enforcing the Dirichlet boundary conditions by modifying the test and trial spaces, we instead enforce them weakly by means of the following equality:

$$\begin{aligned}\Upsilon\langle \varepsilon \{\nabla v_h\}, [\![\theta]\!]\rangle_{\Gamma_D} + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta]\!], [\![v_h]\!]\rangle_{\Gamma_D} &= \Upsilon\langle \varepsilon (\nabla v_h\cdot\mathbf{n}), \theta\rangle_{\Gamma_D} + \frac{\sigma\varepsilon}{h_F}\langle \theta, v_h\rangle_{\Gamma_D} \\ &= \Upsilon\langle \varepsilon (\nabla v_h\cdot\mathbf{n}), g_D\rangle_{\Gamma_D} + \frac{\sigma\varepsilon}{h_F}\langle g_D, v_h\rangle_{\Gamma_D},\end{aligned} \quad (3.2.4)$$

with a *penalty parameter* σ that will be chosen later, and with Υ a value in $\{-1, 1\}$. A choice of $\Upsilon = -1$ results in the *symmetric interior penalty method* while a choice of $\Upsilon = 1$ results in the *non-symmetric interior penalty method.*

Given we are seeking a solution in $H^1(\Omega)$, we have that $[\![\theta]\!] = 0$ almost everywhere. Thus, for all $F \in \mathcal{F}_I \cup \mathcal{F}_N$,

$$\Upsilon\langle \varepsilon \{\nabla v_h\}, [\![\theta]\!]\rangle_F + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta]\!], [\![v_h]\!]\rangle_F = 0. \quad (3.2.5)$$

This allows us to symmetrise (3.2.3), and to ensure that the resulting bilinear form is coercive. Adding (3.2.4) and (3.2.5) in gives

$$\sum_{K\in\mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon\nabla\theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta, v_h)_K \\ + \sum_{F\in\mathcal{F}_h} -\langle \varepsilon \{\nabla\theta\}, [\![v_h]\!]\rangle_F + \Upsilon\langle \varepsilon \{\nabla v_h\}, [\![\theta]\!]\rangle_F + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta]\!], [\![v_h]\!]\rangle_F \\ = \sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v_h)_K + \langle g_N, v_h\rangle_{\Gamma_N} \\ + \Upsilon\langle \varepsilon (\nabla v_h\cdot\mathbf{n}), g_D\rangle_{\Gamma_D} + \frac{\sigma\varepsilon}{h_F}\langle g_D, v_h\rangle_{\Gamma_D}.$$

We also denote, for each cell K, the inflow boundary of K by

$$\partial_- K \coloneqq \{\mathbf{x} \in \partial K : \mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K(\mathbf{x}) < 0\},$$

and similarly by

$$\partial_+ K \coloneqq \{\mathbf{x} \in \partial K : \mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K(\mathbf{x}) \geq 0\},$$

the outflow boundary of K. We note that intervals along individual faces may move between $\partial_- K$ and $\partial_+ K$ at different times $t \in I$.

We wish to impose numerical fluxes on the local problem on each cell, so as to couple neighbouring cells together in a stable fashion in the convection-dominated regime. To this end, we denote the upwind-jump on each cell K across the cell boundary by

$$\lfloor \theta \rfloor_K \coloneqq \begin{cases} \theta_K^+ - \theta_K^- & \text{on } \partial_- K \backslash \Gamma, \\ \theta_K^- - \theta_K^+ & \text{on } \partial_+ K \backslash \Gamma. \end{cases}$$

We weakly impose continuity across this cell boundary portion by the following:

$$\left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K)\theta^+, v_h^+ \right\rangle_{\partial_- K} = \left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K)\theta^-, v_h^+ \right\rangle_{\partial_- K}.$$

In the region $\partial_- K \cap \Gamma_D$, we impose the Dirichlet boundary condition by replacing θ^- by g_D. Thus, for each cell K in $\mathcal{T}_h$, we have the following two equations:

$$\left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K)\theta^+, v_h^+ \right\rangle_{\partial_- K \backslash \Gamma_D} = \left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K)\theta^-, v_h^+ \right\rangle_{\partial_- K \backslash \Gamma_D}, \tag{3.2.6}$$

$$\left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n})\theta^+, v_h^+ \right\rangle_{\partial_- K \cap \Gamma_D} = \left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n})g_D, v_h^+ \right\rangle_{\partial_- K \cap \Gamma_D}. \tag{3.2.7}$$

We note that (3.2.6) is equivalent to

$$\left\langle (\mathbf{u}(\mathbf{x}, t) \cdot \mathbf{n}_K)\lfloor \theta \rfloor, v_h^+ \right\rangle_{\partial_- K \backslash \Gamma_D} = 0. \tag{3.2.8}$$

Adding both (3.2.7) and (3.2.8) into our equation gives

$$\sum_{K \in \mathcal{T}_h} (\theta_t, v_h)_K + (\varepsilon \nabla \theta, \nabla v_h)_K + (\mathbf{u}(\mathbf{x}, t) \cdot \nabla \theta + b(\mathbf{x}, t)\theta, v_h)_K$$

$$
\begin{aligned}
&+ \sum_{F\in\mathcal{F}_h} -\langle \varepsilon \{\nabla\theta\}, [\![v_h]\!]\rangle_F + \Upsilon\langle \varepsilon \{\nabla v_h\}, [\![\theta]\!]\rangle_F + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta]\!], [\![v_h]\!]\rangle_F \\
&- \sum_{K\in\mathcal{T}_h} \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n})\theta^+, v_h^+\right\rangle_{\partial_- K\cap\Gamma_D} + \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n}_K)\lfloor\theta\rfloor, v_h^+\right\rangle_{\partial_- K\setminus\Gamma_D} \\
&= \sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v_h)_K + \langle g_N, v_h\rangle_{\Gamma_N} + \Upsilon\langle \varepsilon\nabla v_h\cdot\mathbf{n}, g_D\rangle_{\Gamma_D} + \frac{\sigma\varepsilon}{h_F}\langle g_D, v_h\rangle_{\Gamma_D} \\
&\qquad - \sum_{K\in\mathcal{T}_h} \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n})g_D, v_h^+\right\rangle_{\partial_- K\cap\Gamma_D}.
\end{aligned}
$$

We can now define the bilinear form $a_{\mathrm{reac},h}(\theta, v)$ and linear functional $l_h(v)$ by

$$
\begin{aligned}
a_{\mathrm{reac},h}(\theta, v) \coloneqq &\sum_{K\in\mathcal{T}_h} (\varepsilon\nabla\theta, \nabla v)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + b(\mathbf{x},t)\theta, v)_K \\
&+ \sum_{F\in\mathcal{F}_h} -\langle \varepsilon \{\nabla\theta\}, [\![v]\!]\rangle_F + \Upsilon\langle \varepsilon \{\nabla v\}, [\![\theta]\!]\rangle_F + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta]\!], [\![v]\!]\rangle_F \\
&+ \sum_{K\in\mathcal{T}_h} \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n})\theta^+, v^+\right\rangle_{\partial_- K\cap\Gamma_D} + \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n}_K)\lfloor\theta\rfloor, v^+\right\rangle_{\partial_- K\setminus\Gamma_D},
\end{aligned}
$$

$$
\begin{aligned}
l_h(v) \coloneqq &\sum_{K\in\mathcal{T}_h} (f(\mathbf{x},t), v)_K + \langle g_N, v\rangle_{\Gamma_N} + \Upsilon\langle \varepsilon\nabla v\cdot\mathbf{n}, g_D\rangle_{\Gamma_D} + \frac{\sigma\varepsilon}{h_F}\langle g_D, v\rangle_{\Gamma_D} \\
&\qquad - \sum_{K\in\mathcal{T}_h} \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n})g_D, v^+\right\rangle_{\partial_- K\cap\Gamma_D}.
\end{aligned}
$$

Additionally we can define the bilinear form $a_h(\theta, v)$ by

$$
\begin{aligned}
a_h(\theta, v) \coloneqq &\sum_{K\in\mathcal{T}_h} (\varepsilon\nabla\theta, \nabla v)_K + (\mathbf{u}(\mathbf{x},t)\cdot\nabla\theta, v)_K \\
&+ \sum_{F\in\mathcal{F}_h} -\langle \varepsilon \{\nabla\theta\}, [\![v]\!]\rangle_F + \Upsilon\langle \varepsilon \{\nabla v\}, [\![\theta]\!]\rangle_F + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta]\!], [\![v]\!]\rangle_F \\
&- \sum_{K\in\mathcal{T}_h} \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n})\theta^+, v^+\right\rangle_{\partial_- K\cap\Gamma_D} + \left\langle (\mathbf{u}(\mathbf{x},t)\cdot\mathbf{n}_K)\lfloor\theta\rfloor, v^+\right\rangle_{\partial_- K\setminus\Gamma_D}.
\end{aligned}
$$

Thus, the semi-discrete discontinuous Galerkin method for the non-stationary convection-diffusion-reaction equation reads:

Let $V_h \coloneqq X_{h,k}$. For each $t \in I$, find $\theta_h(t) \in V_h$ such that

$$\sum_{K \in \mathcal{T}_h} \left(\theta_{ht}, v_h\right)_K + a_{\text{reac},h}\left(\theta_h, v_h\right) = l_h\left(v_h\right) \tag{3.2.9}$$

for all $v_h \in V_h$, with $\theta_h(0) = \theta_{0,h}$ the L^2-projection of θ_0 into V_h.

Similarly the semi-discrete discontinuous Galerkin method for the non-stationary convection-diffusion equation reads as follows:

Let $V_h \coloneqq X_{h,k}$. For each $t \in I$, find $\theta_h(t) \in V_h$ such that

$$\sum_{K \in \mathcal{T}_h} \left(\theta_{ht}, v_h\right)_K + a_h\left(\theta_h, v_h\right) = l_h\left(v_h\right) \tag{3.2.10}$$

for all $v_h \in V_h$, with $\theta_h(0) = \theta_{0,h}$ the L^2-projection of θ_0 into V_h.

3.3 Fully discrete dG formulation for the energy equation

The semi-discrete problem as formulated above is still an infinite-dimensional problem in the time variable. In order to fully reduce the solution of the PDE (2.7.5) to a finite-dimensional problem, we discretise also in time in the following manner.

Let $N > 0$ be a positive integer, and let $t^0 = 0, t^1, t^2, \ldots, t^N = T$ be a strictly increasing sequence of values in the interval $I = [0, T]$. Then we discretise the time interval I into N subintervals I_n, $n \in \{1, \ldots, N\}$, with each subinterval defined by $I_n \coloneqq [t^{n-1}, t^n]$. We denote the timestep length $\tau^n = t^n - t^{n-1}$.

At each timestep t^n, we define a triangulation $\mathcal{T}_h^n$ in the identical fashion to the previous section. We denote by $\mathcal{F}_h^n$ the collection of all $(d-1)$-dimensional faces defined by the interfaces between cells. We also define the set of interior faces $\mathcal{F}_I^n$ and set of faces on the boundary $\mathcal{F}_B^n$. Thus we have that $\mathcal{F}_h^n = \mathcal{F}_I^n \cup \mathcal{F}_B^n$. We also subdivide $\mathcal{F}_B^n$ into faces on the Dirichlet boundary $\mathcal{F}_D^n$ and faces on the Neumann boundary $\mathcal{F}_N^n$, with $\mathcal{F}_D^n \cup \mathcal{F}_N^n = \mathcal{F}_B^n$ and $\mathcal{F}_D^n \cap \mathcal{F}_N^n = \emptyset$.

For each $\mathcal{T}_h^n$ we can define the space of piecewise-polynomial functions $X_{h,k}^n$ in the obvious way: if our mesh $\mathcal{T}_h^n$ is composed of simplicial cells, then

$$X_{h,k}^n \coloneqq \left\{ v_h \in L^2(\Omega) : v_h|_K \circ \mathcal{D}_K \in \mathcal{P}_k \ \forall \ K \in \mathcal{T}_h^n \right\}, \tag{3.3.1}$$

while if the mesh is composed of quadrilateral or hexahedral cells, then

$$X_{h,k}^n \coloneqq \left\{ v_h \in L^2(\Omega) : v_h|_K \circ \mathcal{D}_K \in \mathcal{Q}_k \ \forall \ K \in \mathcal{T}_h^n \right\}. \tag{3.3.2}$$

We then say for $n = 0, \ldots, N$ that $V_h^n \coloneqq X_{h,k}^n$.

Taking the semi-discrete form from (3.2.9) we can discretise the time derivative by a timestepping scheme – here we choose the backward Euler method. The approximation in the backward Euler method is

$$w_t(\mathbf{x}, t^n) \approx \frac{w(\mathbf{x}, t^n) - w(\mathbf{x}, t^{n-1})}{\tau^n}.$$

We can then use this approximation to state the fully discrete discontinuous Galerkin method for the convection-diffusion problem: for $n = 0, \ldots, N$, find $\theta_h^n \in V_h^n$ such that

$$\sum_{K \in \mathcal{T}_h^n} \left(\frac{\theta_h^n - \theta_h^{n-1}}{\tau^n}, v_h \right)_K + a_h\left(\theta_h^n, v_h\right) = l_h\left(v_h\right), \tag{3.3.3}$$

for all $v_h \in V_h^n$, with $\theta_h^0 = \theta_{0,h}$ the L^2-projection of θ_0 into V_h^0.

The sequence $\theta_h^n(\mathbf{x})$, $n = 0, \ldots, N$ is then an approximation to the true solution $\theta(\mathbf{x}, t)$.

Similarly, the fully discrete discontinuous Galerkin method for the convection-diffusion-reaction problem is: for $n = 0, \ldots, N$, find $\theta_h^n \in V_h^n$ such that

$$\sum_{K \in \mathcal{T}_h^n} \left(\frac{\theta_h^n - \theta_h^{n-1}}{\tau^n}, v_h \right)_K + a_{\mathrm{reac},h}\left(\theta_h^n, v_h\right) = l_h\left(v_h\right),$$

for all $v_h \in V_h^n$, with $\theta_h^0 = \theta_{0,h}$ the L^2-projection of θ_0 into V_h^0.

3.4 Finite element method for the Stokes system

Having introduced the dG method for the convection-diffusion-reaction equation, we now concentrate on the FE for the associated stationary Stokes system in the Boussinesq system on a thick-shell domain (2.5.1).

We recall the following two bilinear forms from (2.8.4):

$$\begin{aligned} s\left(\mathbf{u},\mathbf{v}\right) &= \left(2\mu\left(\theta,\mathbf{x}\right)\kappa\left(\mathbf{u}\right),\kappa\left(\mathbf{v}\right)\right), \\ b\left(\mathbf{v},p\right) &= -\left(\nabla\cdot\mathbf{v},p\right), \end{aligned}$$

and the weak formulation (2.8.5): find $\mathbf{u}\in U$, $p\in Q$ such that

$$\left.\begin{aligned} s\left(\mathbf{u},\mathbf{v}\right)+b\left(\mathbf{v},p\right) &= -\left(\rho(\theta,\mathbf{x})\mathbf{g},\mathbf{v}\right) \\ b\left(\mathbf{u},q\right) &= 0 \end{aligned}\right\} \tag{3.4.1}$$

for all $(\mathbf{v},q)\in U\times Q$, where

$$\begin{aligned} W &\coloneqq \left\{\mathbf{w}\in\left[H^1(\Omega)\right]^3 : \mathbf{w}\cdot\mathbf{n}=0 \text{ on } \Gamma\right\}, \\ U &\coloneqq \left\{\mathbf{w}\in W : (\mathbf{w},\mathbf{v}^{(i)})=0 \text{ for } i=1,2,3\right\}, \\ Q &\coloneqq \left\{q\in L^2(\Omega) : (q,1)=0\right\}. \end{aligned}$$

For each $t\in I$, we define the FEM for the stationary Stokes problem (3.4.1) in the following manner. Let $\mathcal{T}_h$ be a mesh of the domain Ω as defined in Section 3.2.

We define the space of piecewise-polynomial functions $X_{h,k}$ in the following way, identically to as in Section 3.3: if our mesh $\mathcal{T}_h$ is composed of simplicial cells, then

$$X_{h,k} \coloneqq \left\{v_h\in L^2(\Omega) : v_h|_K\circ\mathcal{D}_K\in\mathcal{P}_k\ \forall\ K\in\mathcal{T}_h\right\},$$

while if the mesh is composed of quadrilateral or hexahedral cells, then

$$X_{h,k} \coloneqq \left\{v_h\in L^2(\Omega) : v_h|_K\circ\mathcal{D}_K\in\mathcal{Q}_k\ \forall\ K\in\mathcal{T}_h\right\}.$$

We then can introduce the following two spaces for $k \geq 2$:

$$U_{h,k} \coloneqq \left\{ \mathbf{v} \in \left[X_{h,k} \right]^d : \mathbf{v} \in \left[C^0(\Omega) \right]^d \right\},$$
$$Q_{h,k-1} \coloneqq \left\{ q \in X_{h,k-1} : q \in C^0(\Omega) \right\}.$$

The pairing of these two spaces, $U_{h,k} \times Q_{h,k-1}$ gives rise to the Taylor-Hood element (see e.g., [22]).

Defining the discrete versions of the bilinear forms $s(\cdot,\cdot)$ and $b(\cdot,\cdot)$,

$$s_h(\mathbf{u},\mathbf{v}) \coloneqq \sum_{K \in \mathcal{T}_h} \left(2\mu(\theta,\mathbf{x})\,\kappa(\mathbf{u}), \kappa(\mathbf{v}) \right)_K,$$
$$b_h(\mathbf{v},p) \coloneqq -\sum_{K \in \mathcal{T}_h} \left(\nabla \cdot \mathbf{v}, p \right)_K,$$

we state the FEM for the Stokes problem using the Taylor-Hood element as the following:

find $(\mathbf{u}_h, p_h) \in U_{h,k} \times Q_{h,k-1}$ such that

$$\left.\begin{aligned} s_h(\mathbf{u}_h,\mathbf{v}_h) + b_h(\mathbf{v}_h,p_h) &= -\left(\rho(\theta,\mathbf{x})\mathbf{g},\mathbf{v}_h\right) \\ b_h(\mathbf{u}_h,q_h) &= 0 \end{aligned}\right\}, \tag{3.4.2}$$

for all $(\mathbf{v}_h, q_h) \in U_{h,k} \times Q_{h,k-1}$.

The well-posedness of this formulation is guaranteed, conditional upon the fulfillment of the *discrete inf-sup condition* by the chosen mixed finite element pairing, that is, upon the existence of a constant $\beta^* > 0$ such that

$$\inf_{q_h \in Q_{h,k-1}} \sup_{\mathbf{v}_h \in U_{h,k}} \frac{b_h(\mathbf{v}_h,q_h)}{\|\mathbf{v}_h\|_U \|q_h\|_Q} \geq \beta^*.$$

We refer the reader to [45] and [22] for proof of the sufficiency of the condition in guaranteeing well-posedness, and to [104], which builds upon [19], that show the Taylor-Hood mixed finite element does indeed satisfy this condition.

3.5 The full coupled FE/dG system

Having introduced the numerical schemes for the convection-diffusion (2.7.7) and Stokes problems (2.8.5) in turn, we are now ready to set out the complete process for approximating the full system (2.9.1).

Since (2.7.7) is non-stationary (it contains the time-derivate term u_t), while (2.8.5) is stationary (its time-dependence comes purely from the time-dependence of its parameter values), we employ a scheme that alternates between the numerical solution of (3.3.3) and (3.4.2) in the following manner.

Given an initial condition on the temperature, $\theta_h^0 = \theta_{0,h}$, we use this to solve (3.4.2) for $(\mathbf{u}_h^0, p_h^0)$, with θ_h^0 used to evaluate $\mu(\theta_h, \mathbf{x})$ and $\rho(\theta_h, \mathbf{x})$. Having established the initial convection field in this way, this is then used when timestepping forward: at each timestep t^n, we solve the convection-diffusion problem (3.3.3) for θ_h^n with the *previous convection field* $\mathbf{u}^{n-1}$ used to evaluate the term $\mathbf{u} \cdot \nabla\theta$ in the bilinear form a_h. We are then in turn able to employ θ_h^n in solving (3.4.2) for $\mathbf{u}_h^n$ and p_h^n. This scheme is presented in Algorithm 3.1.

Algorithm 3.1 Calculate $(\theta_h^n, \mathbf{u}_h^n, p_h^n)$ for $n = 0, \ldots, N$

$\theta_h^0 \leftarrow \theta_{0,h}$.
Solve (3.4.2) with θ_h^0 for $\mathbf{u}_h^0, p_h^0$.
for $n = 1, \ldots, N$ **do**
    Solve (3.3.3) with $\mathbf{u}_h^{n-1}$ for θ_h^n.
    Solve (3.4.2) with θ_h^n for $\mathbf{u}_h^n, p_h^n$.
end for

3.6 Linear systems, their preconditioning, and solution

Finite Element Methods reduce the original (linearised) infinite-dimensional problem to a finite-dimensional linear system. Implementation of these methods, therefore, combines the assembly of the required matrices and vectors to frame the problem as a linear system, with the solution of this resulting system.

In order to quickly simulate these systems, we therefore need to be able to both assemble and solve quickly. It is often beneficial to design algorithms that take both these requirements into account simultaneously.

We consider the solution stage first, as it is often the most crucial, consuming the majority of the computational time in a simulation.

Linear systems resulting from finite element approximations are typically characterised by their large size, and sparse, highly-structured nature. This lends them to solution by iterative methods, since sparse direct methods cannot scale suitably for use in the largest scenarios (typically, when tackling practically relevant 3D problems).

Typically, iterative methods are known to converge at a rate dependent on the properties of the matrix being inverted [46]. Often, this property is the condition number of the matrix (the ratio of the largest and smallest eigenvalues) or the clustering properties of the eigenvalues. Hence it is often worth investing in preconditioning, a technique to 'improve' the properties of the matrix being inverted, such that convergence of the iterative technique is accelerated. The ability to correctly precondition matrices is crucial in the largest, or worst-behaved, problems. Recent work in this mathematical area include e.g., [86, 87] for coupled flow of magma and mantle.

Preconditioners usually work best when the properties of the original matrix are taken into account. For example, if the original matrix is block-structured, or diagonally-dominant, then the best preconditioners will take account of, and make use of, this structure to treat the matrix in an improved way. Essentially, knowledge of the matrix, or the original problem itself, is used to give the solver the best chance of solving in fewer iterations. We employ an FGMRES solver for both the Stokes system and temperature system, since this allows us to modify the preconditioning matrices within the iteration if desired. We precondition with a Schur-based preconditioner to solve the Stokes system, and a Jacobi preconditioner to solve for the temperature.

The assembly process typically should work to ensure that as much structure is preserved as is reasonably possible. Thus, assembly should ensure that blocks in the coupling between components of the solution fields are reflected as blocks in the resulting matrix.

Alongside this, the assembly of the matrix needs to be as fast as possible. Parallelism of this task is generally possible, since it involves computation of integrals over cells and (possibly) faces. This is, typically, a fully asynchronous process. In Chapter 5 we describe some of the algorithms needed to parallelise this assembly in our implementation in deal.II. This makes heavy use of the `WorkStream` functionality [103] as well as the ability to decompose the domain into discrete subdomains, with each parallel process taking responsibility for contributions from all cells and faces in a given subdomain.

3.7 Adaptive mesh refinement

A dominant aim of this project is to derive a rigorous *a posteriori* error indicator for use in adaptive mesh refinement (AMR).

Adaptivity is the technique of changing the structure of the simulation locally between timesteps, or iteratively on a single timestep, in order to provide more or less accuracy at a given location. The three most common techniques are:

- adapting the mesh locally by adding cells or coarsening adjacent cells into larger cells;
- modifying the degree of the polynomial approximation space locally;
- keeping the number of cells fixed but modifying the positions of nodes.

These are referred to as h-, p- and r-refinement, respectively. We consider here only h-refinement.

The use of h-refinement has a long history in the finite element community (e.g., [10, 4]), and is useful for targeting computational effort at areas of the domain that would most benefit from it. These are usually areas containing singularities, boundary layers, interior layers, and fast-varying coefficients. Increasing the spatial resolution in these areas allows improved approximation of the solution, but at the cost of additional computation. Conversely, other areas may be coarsened, ensuring these areas do not use so much computational effort, but at the cost of diminished

accuracy. The task of an adaptivity indicator is, thus, simple and at the same time incredibly complex: it should rank areas that would most benefit from extra refinement, and rank those whose coarsening will least damage the accuracy of the overall simulation.

The main difficulty, of course, is that we do not know the exact solution of our simulation *a priori* and, thus, cannot know the exact error at any point. Instead, we employ *a posteriori* bounds on the error, which allow us to bound some quantity of error based purely on knowledge of the partial differential equation and the approximate solution. Based upon this bound, an error indicator may be selected for the purposes of directing the adaptivity.

Another complication in providing such an indicator is the fact that of course we may not in general have full knowledge of the effect of a given adaptation. Without completing the calculation with a given adaptation, we are in general unable to say how greatly this will affect the accuracy, since we do not know the resulting approximate solution until it is calculated. While the repeated comparison of alternatively adapted meshes may be considered in the case of a stationary problem, it is largely uncountenanceable in the case of a time-dependent simulation, since adaptation at a single timestep may have strong effects on the accuracy later in the simulation, and would, thus, require multiple simulations through the full time interval being simulated. Doing so in order to decide upon the best locations to refine is obviously infeasible, and largely defeats the aim of the adaptivity, which is to reduce the computational effort. Thus, the error indicator is unlikely to know the full effect of a given refinement or coarsening, especially in a time-dependent problem.

In Chapter 4 we derive a new *a posteriori* error bound for the dG discretisation of the temperature equation in a rigorous manner. This allows us to then produce an error indicator for use in driving the adaptivity algorithm.

Chapter 4

An *a posteriori* error bound

We wish to derive an *a posteriori* error bound for the discontinuous Galerkin method applied to the non-stationary convection-diffusion problem (2.7.1)–(2.7.4), for use within an adaptive finite element solution of the full coupled system (2.5.1), as described in Algorithm 3.1. In such a scheme, we have that the convection term $\mathbf{u}$ is a finite element function in U_h approximately solving the Stokes system (3.4.2).

The approach we take here, similarly to [27], is to first derive an *a posteriori* error bound for the stationary problem, before using the elliptic reconstruction framework [72] to extend this to the non-stationary problem.

Little previous work has been done on the *a posteriori* analysis of the stationary convection-diffusion problem without a reaction term, except where severe restrictions are placed on the divergence of the convection. In order to use the Lax-Milgram Lemma to prove existence and uniqueness of the solution, coercivity and continuity of the bilinear form need to be established. The standard assumptions to ensure coercivity are then:

$$\nabla \cdot \mathbf{u} \in L^{\infty}(\Omega),$$

and there exists a constant $\gamma_0 > 0$ such that

$$-\frac{1}{2}\nabla \cdot \mathbf{u}(\mathbf{x}, t) + b(\mathbf{x}, t) > \gamma_0. \tag{4.0.1}$$

The usefulness of this assumption (4.0.1) is evident when considering that, in the case of zero boundary conditions, we may split the convection and reaction terms of the weak formulation into an *anti-symmetric* part and a *symmetric* part in the following way:

$$(\mathbf{u}\cdot\nabla\theta + b\theta, v) = \frac{1}{2}(\mathbf{u}\cdot\nabla\theta, v) - \frac{1}{2}(\mathbf{u}\cdot\nabla v, \theta) + \left(\left(b - \frac{1}{2}\nabla\cdot\mathbf{u}\right)\theta, v\right).$$

The assumption of (4.0.1) then is seen to be an assumption that the symmetric part is positive, bounded uniformly below by a positive constant. Coercivity of the bilinear form can then be trivially established by replacing v with θ, which cancels out the anti-symmetric part, and leaves only the positive symmetric part.

However, this condition cannot always be assumed to hold for general $\mathbf{u}$ and b. In particular, in the case of vanishing reaction, this demands the troublesome requirement that $-\frac{1}{2}\nabla\cdot\mathbf{u} > \gamma_0 > 0$ everywhere. Since in our setup $\mathbf{u}$ is the numerical solution of a Stokes problem, the divergence is only approximately zero, and thus we cannot in general enforce the condition of strictly negative divergence.

One approach to handling this issue is to add in an artificial reaction term characterised by the *reaction coefficient* δ_0, with $\delta_0 > \frac{1}{2}\nabla\cdot\mathbf{u}$, so that we can satisfy (4.0.1) and thus have coercivity. This can be unsatisfactory since, while we know $\nabla\cdot\mathbf{u} \in [L^\infty(\Omega)]^d$, we demand that δ_0 must be at least as large as $\nabla\cdot\mathbf{u}$, and δ_0 ultimately ends up inside an exponential factor in the *a posteriori* error bound for the non-stationary convection-diffusion problem.

An alternative approach, proposed in [9, 41], is to use an exponential-fitting technique, testing against a modified test function to prove coercivity in a modified norm. However, this alone is not enough to guarantee coercivity in the modified norm without still assuming $\nabla\cdot\mathbf{u} \leq 0$.

We proceed by combining the two approaches: the exponential fitting technique modifies the norm, and the effective reaction term, which is then supplemented by an additional reaction term, ensures coercivity. As we shall see, in this way a minimal amount of artificial reaction is introduced in all regimes. The benefit of combining these two approaches is that they can work together complementarily to give the strongest result. By modifying the norm by an exponential-fitting technique, we

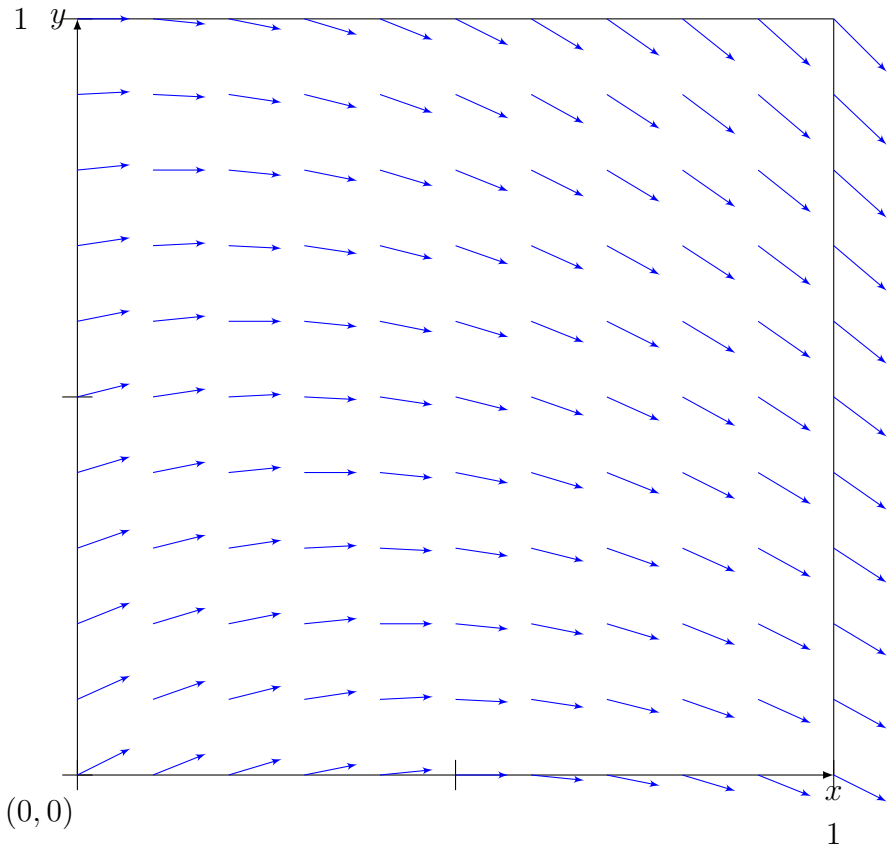


FIGURE 4.1: Flow field diagram for the example of negative-divergence flow in a unit box.

are able to enlarge the set of convection fields under which no additional reaction is required to give coercivity. However, for convection fields where this is not sufficient, we still add enough reaction locally to ensure coercivity. In this manner, we reduce the additional reaction that must be added. This is important to minimise, since our later results depend upon this additional reaction in an exponential fashion.

We introduce here two 2-dimensional examples to highlight the theoretical development. In the first, we consider a box $[0,1]^2$ with a flow field $\left(1, \frac{1}{2} - \frac{1}{2}y - x\right)^\intercal$ across it, as illustrated in Figure 4.1. In this case, $\nabla \cdot \mathbf{u} = -\frac{1}{2}$, and so we should have little difficulty in deriving a bound as shown in [90].

In the second example, we consider an annular domain, inner radius R_0, outer radius R_1, centred at the origin, with free-slip boundaries (that is, $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ). Suppose that we only approximately impose a circular flow $\mathbf{u} \approx (y, -x)^\intercal$, be it through solution of a Stokes system or some other means. Given that we are using

an approximate convection field, we can expect the divergence of the approximate field to be only approximately zero. In this case, there will be areas of positive divergence. Successive cycles of flow around the annulus may cause the same material to flow through this same area of positive divergence multiple times, compounding the associated error at each cycle. Thus, to bound such a case, we can expect to require an exponential term in our error bound.

Our strategy over the remainder of the chapter is the following: after proving an *a posteriori* error bound on the stationary convection-diffusion-reaction equation in a modified norm, we reframe the non-stationary convection-diffusion equation (crucially, with no reaction) as a non-stationary convection-diffusion-reaction equation by means of the observation that we may rewrite the equation

$$\theta_t - \varepsilon\Delta\theta + \mathbf{u}\cdot\nabla\theta = f,$$

as

$$\theta_t - \varepsilon\Delta\theta + \mathbf{u}\cdot\nabla\theta + \delta\theta = f + \delta\theta.$$

Then, using the elliptic reconstruction framework of [72], and a Gronwall inequality, we bound the error for the non-stationary convection-diffusion problem, converting the reaction term into an exponential factor in the final error bound.

We note here that this error bound takes the view that the convection field is imposed upon the temperature solution without any dependence upon the underlying temperature solution. The *a posteriori* analysis of the fully coupled system of Stokes flow and temperature equation is beyond the scope of this work, and is, to the author's knowledge, an unresolved area of study.

4.1 Problem definitions

In view of proving the *a posteriori* error bound for the non-stationary problem, we first consider the *stationary* convection-diffusion-reaction problem

$$-\varepsilon\Delta\theta + \mathbf{u}(\mathbf{x})\cdot\nabla\theta + \delta(\mathbf{x})\theta = f(\mathbf{x}) \qquad \text{on } \Omega, \tag{4.1.1}$$

$$\theta = 0 \qquad \text{on } \Gamma_D, \tag{4.1.2}$$

$$\varepsilon \frac{\partial \theta}{\partial \mathbf{n}} = g_N(\mathbf{x}) \qquad \text{on } \Gamma_N, \tag{4.1.3}$$

with $\delta \in L^2(\Omega)$, where we focus on the case of zero Dirichlet values since this problem can always be reduced to such (cf. [44]). Here we assume ε is a positive constant, $\mathbf{u}$ belongs to a Stokes finite element space, and thus $\mathbf{u} \in [W^{1,\infty}(\Omega)]^d$, with bounded divergence, and $f \in L^2(\Omega)$. We require $g_D \in H^{\frac{1}{2}}(\Gamma)$, $g_N \in H^{\frac{3}{2}}(\Gamma)$, and apply the same conditions to the sets Γ_D and Γ_N as in Chapter 3.

Considering the convection-diffusion-reaction problem (4.1.1)–(4.1.3), we restate the bilinear form $a_{\text{reac}}(\cdot,\cdot)$ from (2.7.8), with b replaced by δ:

$$a_{\text{reac}}(\theta, v) = (\varepsilon \nabla \theta, \nabla v) + (\mathbf{u}(\mathbf{x},t) \cdot \nabla \theta, v) + (\delta \theta, v).$$

The weak formulation for the problem including reaction δ then reads: find $\theta \in H^1_D(\Omega)$ such that

$$a_{\text{reac}}(\theta, v) = l(v), \tag{4.1.4}$$

for all $v \in H^1(\Omega)$.

Let $\Pi : V_h + H^1(\Omega) \to X_{h,1}$ be the L^2-projection, with $X_{h,1}$ defined as in (3.3.1)–(3.3.2). For $w_h, v_h \in V_h + H^1(\Omega)$, we (re)define the bilinear form $a_{\text{reac},h}$ with added reaction δ:

$$\begin{aligned} a_{\text{reac},h}(w_h, v_h) \coloneqq & \sum_{K \in \mathcal{T}_h} (\varepsilon \nabla_h w_h, \nabla_h v_h)_K + (\mathbf{u} \cdot \nabla_h w_h + \delta w_h, v_h)_K \\ & - \sum_{F \in \mathcal{F}_h} \Big(\langle \{\varepsilon \nabla \Pi w_h\}, [\![v_h]\!] \rangle_F + \langle \{\varepsilon \nabla \Pi v_h\}, [\![w_h]\!] \rangle_F - \frac{\varepsilon \sigma}{h_F} \langle [\![w_h]\!], [\![v_h]\!] \rangle_F \Big) \\ & \qquad - \sum_{K \in \mathcal{T}_h} \Big(\langle \mathbf{u} \cdot \mathbf{n} w_h, v_h \rangle_{\partial_- K \cap \Gamma_D} + \langle \mathbf{u} \cdot \mathbf{n}_K \lfloor w_h \rfloor, v_h \rangle_{\partial_- K \setminus \Gamma_D} \Big), \end{aligned}$$

Note that here and in the following we use the *symmetric* interior penalty method, with the parameter $\Upsilon = -1$, and that we use the *piecewise gradient* ∇_h on each cell. We note that these bilinear forms coincide with the previous definitions of a_h and $a_{\text{reac},h}$ on $V_h \times V_h$. Also note that we assume, without loss of generality, that $\sigma \geq 1$.

Arguing in an identical manner as in the derivation of (3.2.10), the discontinuous Galerkin method for the convection-diffusion-reaction problem (4.1.1)–(4.1.3) reads: find $\theta_h \in V_h$ such that

$$a_{\mathrm{reac},h}\left(\theta_h, v_h\right) = l_h\left(v_h\right) \tag{4.1.5}$$

for all $v_h \in V_h$.

We also note here that for $\theta, v \in H_D^1(\Omega)$,

$$a_{\mathrm{reac},h}\left(\theta, v\right) = a_{\mathrm{reac}}\left(\theta, v\right). \tag{4.1.6}$$

Remark 4.1. We define here some useful shorthand notation for certain three-dimensional vector fields. In three dimensions, the curl operator is defined to be

$$\mathbf{curl}\boldsymbol{\Phi} = \begin{pmatrix} \frac{\partial \boldsymbol{\Phi}_3}{\partial y} - \frac{\partial \boldsymbol{\Phi}_2}{\partial z} \\ \frac{\partial \boldsymbol{\Phi}_1}{\partial z} - \frac{\partial \boldsymbol{\Phi}_3}{\partial x} \\ \frac{\partial \boldsymbol{\Phi}_2}{\partial x} - \frac{\partial \boldsymbol{\Phi}_1}{\partial y} \end{pmatrix},$$

for a vector-valued function $\boldsymbol{\Phi}$.

In the case of two-dimensional flow, it is helpful to reconsider this as a three-dimensional flow with a zero z-direction component, e.g.

$$\mathbf{u} \coloneqq \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ 0 \end{pmatrix}.$$

We observe that, for a function of the form

$$\boldsymbol{\Psi} \coloneqq \begin{pmatrix} 0 \\ 0 \\ g(x,y) \end{pmatrix},$$

where $g(x,y)$ is constant in the z-direction, we have

$$\mathbf{curl}\boldsymbol{\Psi} = \begin{pmatrix} \frac{\partial g}{\partial y} \\ -\frac{\partial g}{\partial x} \\ 0 \end{pmatrix},$$

i.e. the curl is a two-dimensional flow in the plane. Since this property will be of use, this motivates the following shorthand: for a vector-valued function $\boldsymbol{\Psi}$ of the form

$$\boldsymbol{\Psi} \coloneqq \begin{pmatrix} 0 \\ 0 \\ g(x,y) \end{pmatrix},$$

where $g(x,y)$ is a scalar-valued function with no dependence on z, we use the shorthand $\boldsymbol{\Psi} = g(x)$, which allows us to write $\mathbf{curl}\, g(x)$ instead of $\mathbf{curl}\boldsymbol{\Psi}$.

4.2 Exponential fitting

Since we assume $\mathbf{u} \in [W^{1,\infty}(\Omega)]^d \subset [L^2(\Omega)]^d$, this has a Helmholtz decomposition as guaranteed by the following result (e.g., [101, 45]).

Lemma 4.2 (Helmholtz decomposition)**.** *Let $d \in \{2,3\}$. Every $\mathbf{u} \in [L^2(\Omega)]^d$ has a decomposition of the form*

$$\mathbf{u} = \nabla\eta + \boldsymbol{curl}\boldsymbol{\phi},$$

with $\eta \in H^1(\Omega)$ and $\boldsymbol{\phi} \in [H^1(\Omega)]^3$. In the case of $d = 2$, we use the view of the convection field as a three-dimensional vector field with zero z-direction component, as in Remark 4.1.

Remark 4.3. Since η is the solution of the equation $\Delta\eta = \nabla \cdot \mathbf{u}$ on a smooth, or convex polygonal domain, we have that $\eta \in H^2(\Omega) \subset W^{1,\infty}(\Omega)$ and $\mathbf{curl}\boldsymbol{\phi} = \mathbf{u} - \nabla\eta \in L^\infty(\Omega)^d$.

Remark 4.4. Additionally, since we have assumed that $\mathbf{u} \cdot \mathbf{n} = 0$ on Γ_N, we have $\nabla\eta \cdot \mathbf{n} = 0$ on Γ_N (cf. [45, Thm 3.2]).

We now define the *weighting function*

$$\psi \coloneqq \exp(-\alpha\eta), \tag{4.2.1}$$

with $\alpha \in \mathbb{R}^+$ a constant to be determined later, so that

$$\nabla\psi = -\alpha\psi\nabla\eta.$$

Since $\eta \in W^{1,\infty}(\Omega)$ (cf. Remark 4.3), we have that $\psi \in W^{1,\infty}(\Omega)$. Thus $\psi v \in H^1(\Omega)$ for all $v \in H^1(\Omega)$, and $\psi w \in H^1_D(\Omega)$ for all $w \in H^1_D(\Omega)$.

With this weighting function, we are able to define the ψ-weighted L^p-norm $\|\cdot\|_{\psi,\omega,p}$ as

$$\|v\|^p_{\psi,\omega,p} \coloneqq \int_\omega \psi v^p \,\mathrm{d}x,$$

where we suppress the ω subscript if $\omega = \Omega$, and suppress the p subscript if $p = 2$.

In the case of $p = \infty$, we use the definition $\|v\|_{\psi,\omega,\infty} \coloneqq \sup_\omega \left|\sqrt{\psi} v\right|$.

For $\theta, v \in H^1(\Omega)$, using ψv as test function in the definition of a_{reac} and applying the product rule, yields

$$a_{\mathrm{reac}}(\theta, \psi v) = (\varepsilon \nabla\theta, \psi\nabla v) + ((\mathbf{u} - \alpha\varepsilon\nabla\eta)\cdot\nabla\theta, \psi v) + (\delta\theta, \psi v).$$

Integration by parts on the second term, and using (4.2.1) and Remark 4.4, reveals

$$\begin{aligned}((\mathbf{u} - \alpha\varepsilon\nabla\eta)\,\theta, \psi\nabla v) + ((\mathbf{u} - \alpha\varepsilon\nabla\eta)\,\psi v, \nabla\theta) \\ = ((\alpha\nabla\eta - \nabla)\cdot(\mathbf{u} - \alpha\varepsilon\nabla\eta)\,\theta, \psi v) + \langle(\mathbf{u} - \alpha\varepsilon\nabla\eta)\cdot\mathbf{n}\theta, \psi v\rangle_{\Gamma_D}, \quad (4.2.2)\end{aligned}$$

where we abuse the 'dot' notation to simultaneously denote both the vector dot-product and part of the divergence operator: that is, we write

$$(\alpha\nabla\eta - \nabla)\cdot\mathbf{u} = \alpha\nabla\eta\cdot\mathbf{u} - \nabla\cdot\mathbf{u}.$$

Equation (4.2.2) allows us to write

$$\begin{aligned}a_{\mathrm{reac}}(\theta, \psi v) = (\varepsilon\nabla\theta, \psi\nabla v) + ((\delta + (\alpha\nabla\eta - \nabla)\cdot(\mathbf{u} - \alpha\varepsilon\nabla\eta))\,\theta, \psi v) \\ - ((\mathbf{u} - \alpha\varepsilon\nabla\eta)\,\theta, \psi\nabla v) + \langle(\mathbf{u} - \alpha\varepsilon\nabla\eta)\cdot\mathbf{n}\theta, \psi v\rangle_{\Gamma_D}, \quad (4.2.3)\end{aligned}$$

and, thus,

$$\begin{aligned}a_{\mathrm{reac}}(\theta, \psi\theta) = (\varepsilon\nabla\theta, \psi\nabla\theta) + \left(\left(\delta + \frac{1}{2}(\alpha\nabla\eta - \nabla)\cdot(\mathbf{u} - \alpha\varepsilon\nabla\eta)\right)\theta, \psi\theta\right) \\ + \frac{1}{2}\langle(\mathbf{u} - \alpha\varepsilon\nabla\eta)\cdot\mathbf{n}\theta, \psi\theta\rangle_{\Gamma_D}. \quad (4.2.4)\end{aligned}$$

Similarly, we can write, for $w_h, v_h \in V_h$,

$$\begin{aligned}
a_{\mathrm{reac},h}\left(w_h, \psi v_h\right) &= \sum_{K\in\mathcal{T}_h} \left(\varepsilon\nabla_h w_h, \psi\nabla_h v_h\right)_K + \left((\mathbf{u}-\alpha\varepsilon\nabla\eta)\cdot\nabla_h w_h + \delta w_h, \psi v_h\right)_K \\
&\quad - \sum_{F\in\mathcal{F}_h} \left(\left\langle \left\{\varepsilon\nabla\Pi w_h\right\}, \left[\!\left[\psi v_h\right]\!\right]\right\rangle_F + \left\langle \left\{\varepsilon\nabla\Pi(\psi v_h)\right\}, \left[\!\left[w_h\right]\!\right]\right\rangle_F - \frac{\varepsilon\sigma}{h_F}\left\langle \left[\!\left[w_h\right]\!\right], \left[\!\left[\psi v_h\right]\!\right]\right\rangle_F \right) \\
&- \sum_{K\in\mathcal{T}_h} \left(\left\langle (\mathbf{u}-\alpha\varepsilon\nabla\eta)\cdot\mathbf{n} w_h, \psi v_h\right\rangle_{\partial_- K\cap\Gamma_D} + \left\langle (\mathbf{u}-\alpha\varepsilon\nabla\eta)\cdot\mathbf{n}_K \lfloor\psi v_h\rfloor, w_h\right\rangle_{\partial_- K\setminus\Gamma_D} \right),
\end{aligned}$$

and, using integration by parts and the cell-wise analogue of (4.2.2), we get

$$\begin{aligned}
&a_{\mathrm{reac},h}\left(w_h, \psi w_h\right) \\
&\quad = \sum_{K\in\mathcal{T}_h} \left(\varepsilon\nabla_h w_h, \psi\nabla_h w_h\right)_K + \left(\left(\delta + \frac{1}{2}\left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u}-\alpha\varepsilon\nabla\eta\right)\right) w_h, \psi w_h \right)_K \\
&\quad - \sum_{F\in\mathcal{F}_h} \left(\left\langle \left\{\varepsilon\nabla\Pi w_h\right\}, \left[\!\left[\psi w_h\right]\!\right]\right\rangle_F + \left\langle \left\{\varepsilon\nabla\Pi(\psi w_h)\right\}, \left[\!\left[w_h\right]\!\right]\right\rangle_F - \frac{\varepsilon\sigma}{h_F}\left\langle \left[\!\left[w_h\right]\!\right], \left[\!\left[\psi w_h\right]\!\right]\right\rangle_F \right) \\
&+ \sum_{K\in\mathcal{T}_h} \left(\left\langle (\mathbf{u}-\alpha\varepsilon\nabla\eta)\cdot\mathbf{n} w_h, \psi w_h\right\rangle_{\partial_+ K\cap\Gamma_D} + \left\langle (\mathbf{u}-\alpha\varepsilon\nabla\eta)\cdot\mathbf{n}_K \lfloor\psi w_h\rfloor, w_h\right\rangle_{\partial_+ K\setminus\Gamma_D} \right).
\end{aligned}$$

We introduce the helpful notation

$$\mathcal{L} \coloneqq \delta + \frac{1}{2}\left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u}-\alpha\varepsilon\nabla\eta\right),$$

and, for future reference,

$$\mathcal{M} \coloneqq \delta + \left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u}-\alpha\varepsilon\nabla\eta\right). \tag{4.2.5}$$

Then, if δ is large enough that $\mathcal{L} \geq 0$, we may define the ψ-weighted norm $|||v_h|||_\psi$:

$$|||v_h|||_\psi^2 \coloneqq \sum_{K\in\mathcal{T}_h} \varepsilon \left\|\nabla v_h\right\|_{\psi,K}^2 + \sum_{K\in\mathcal{T}_h} \left\|\sqrt{\mathcal{L}} v_h\right\|_{\psi,K}^2 + \sum_{F\in\mathcal{F}_h} \frac{\sigma\varepsilon}{h_F} \left\|\left[\!\left[v_h\right]\!\right]\right\|_{\psi,F}^2. \tag{4.2.6}$$

We note that, in the case of a divergence-free convection field, we may allow $\eta = 0$, in which case $\mathcal{L} = 0$ if we also choose $\delta = 0$. See Section 4.8 for a discussion of this case: in the following analysis, for simplicity of presentation, we assume $\mathcal{L} \neq 0$. All the results follow analogously in the $\mathcal{L} = 0$ case, however.

For $\mathbf{w} \in [L^2(\Omega)]^d$, we further define the semi-norm

$$|\mathbf{w}|_{\psi,\star} \coloneqq \sup_{v \in H^1_D(\Omega)\setminus\{0\}} \frac{\int_\Omega \mathbf{w}\psi \cdot \nabla v \, \mathrm{d}x}{|||v|||_\psi}.$$

Finally, we now define

$$|v_h|^2_{\psi,A} \coloneqq |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, v_h|^2_{\psi,\star} + \sum_{F\in\mathcal{F}_h} \frac{h_F\|\mathbf{u} - \alpha\varepsilon\nabla\eta\|^2_{F,\infty}}{\varepsilon} \|[\![v_h]\!]\|^2_{\psi,F}\,. \tag{4.2.7}$$

These definitions will be used to bound the convective derivative by the semi-norm $|(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, v_h|^2_{\psi,\star}$ and the jump terms $h_F\|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{F,\infty}\varepsilon^{-1}\,\|[\![v_h]\!]\|^2_{\psi,F}$, analogously to [108] and the subsequent [90]. Here we note that

$$h_F\|\mathbf{u} - \alpha\varepsilon\nabla\eta\|^2_{F,\infty}\varepsilon^{-1} = \sigma^{-1}Pe_L^2\frac{\sigma\varepsilon}{h_F},$$

where Pe_L is the modified local mesh Péclet number and $\frac{\sigma\varepsilon}{h_F}$ is the penalty term.

We end this section with the following observation, which will be useful later.

Remark 4.5. For any vector field $\mathbf{b}$ and scalar function w, by the Cauchy-Schwarz inequality,

$$\begin{aligned}
|\mathbf{b}w|_{\psi,\star} &\coloneqq \sup_{v\in H^1_D(\Omega)\setminus\{0\}} \frac{\int_\Omega \mathbf{b}\psi w\cdot\nabla v \, \mathrm{d}x}{|||v|||_\psi} \\
&\leq \sup_{v\in H^1_D(\Omega)\setminus\{0\}} \frac{\|\mathbf{b}w\|_\psi \, \|\nabla v\|_\psi \, \mathrm{d}x}{|||v|||_\psi} \\
&\leq \sup_{v\in H^1_D(\Omega)\setminus\{0\}} \frac{\left(\sum_{K\in\mathcal{T}_h}\|\mathbf{b}w\|^2_{\psi,K}\right)^{\frac{1}{2}}\left(\sum_{K\in\mathcal{T}_h}\|\nabla v\|^2_{\psi,K}\right)^{\frac{1}{2}}}{\sqrt{\varepsilon}\left(\sum_{K\in\mathcal{T}_h}\|\nabla v\|^2_{\psi,K}\right)^{\frac{1}{2}}} \\
&= \frac{1}{\sqrt{\varepsilon}}\left(\sum_{K\in\mathcal{T}_h}\|\mathbf{b}w\|^2_{\psi,K}\right)^{\frac{1}{2}} \\
&\leq \frac{1}{\sqrt{\varepsilon}}\left(\sum_{K\in\mathcal{T}_h}\left(\|\mathbf{b}\|^2_{\psi,K,\infty}\|w\|^2_K\right)\right)^{\frac{1}{2}}.
\end{aligned} \tag{4.2.8}$$

4.2.1 Coercivity and continuity results

It is trivial to observe the following from (4.2.4):

Lemma 4.6 (Coercivity of $a_{\mathrm{reac}}(\cdot,\psi\cdot)$ over $H^1_D(\Omega)$)**.** *For $w \in H^1_D(\Omega)$,*

$$a_{reac}(w,\psi w) = |||w|||^2_{\psi}.$$

In what follows, we use the notation $\lesssim$ to denote "less than, up to a constant independent of h and ε".

Lemma 4.7 (Continuity of $a_{\mathrm{reac}}(\cdot,\psi\cdot)$ over $H^1_D(\Omega)$)**.** *Under the assumption that, for $\mathbf{x} \in \Omega$,*

$$\delta(\mathbf{x}) \geq \max\left\{0, -2\left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)(\mathbf{x})\right\}, \tag{4.2.9}$$

we have that for $w, v \in H^1_D(\Omega)$,

$$(\varepsilon\nabla w, \psi\nabla v) + \left(\left(\delta + \left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)\right)w, \psi v\right) \lesssim |||w|||_{\psi}|||v|||_{\psi}, \tag{4.2.10}$$

and

$$a_{reac}(w,\psi v) \lesssim \left(|||w|||_{\psi} + |\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)w|_{\psi,\star}\right)|||v|||_{\psi}.$$

Proof. The first assertion (4.2.10) follows from the assumption (4.2.9). Applying (4.2.10) to (4.2.3), we have

$$\begin{aligned}
a_{\mathrm{reac}}(w,\psi v) =& (\varepsilon\nabla w, \psi\nabla v) + \left(\left(\delta + \left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)\right)w, \psi v\right) \\
&- \left(\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)w, \psi\nabla v\right) + \left(\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)\cdot\mathbf{n}w, \psi v\right)_{\Gamma_D} \\
\lesssim& (\varepsilon\nabla w, \psi\nabla v) + \left(\left(\delta + \left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)\right)w, \psi v\right) \\
&- \left(\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)w, \psi\nabla v\right) \\
\lesssim& \left(|||w|||_{\psi} + |\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)w|_{\psi,\star}\right)|||v|||_{\psi}.
\end{aligned}$$

□

Motivated by this continuity result, we choose δ as in (4.2.9), and then have the following result.

Lemma 4.8 (Continuity of $a_{\text{reac}}(\cdot, \psi\cdot)$ over $(V_h + H^1_D(\Omega)) \times H^1_D(\Omega)$)**.** *It is easy to verify that, for $w_h \in V_h + H^1_D(\Omega)$, $v \in H^1_D(\Omega)$,*

$$a_{reac}(w_h, \psi v) \lesssim (|||w_h|||_\psi + |w_h|_{\psi,A})\, |||v|||_\psi .$$

Having chosen δ, we are able to characterise the behaviour of the weight ψ and the term $\mathcal{L}$ based upon the underlying flow pattern.

Since η is the solution of the equation

$$\Delta\eta = \nabla \cdot \mathbf{u},$$

a flow field of purely negative-divergence will lead to a large weighting, while a purely positive-divergence field will have a reduced weighting. A divergence-free field has weighting $\psi = 1$. In a similar way, $\frac{1}{2}(\alpha\nabla\eta - \nabla) \cdot (\mathbf{u} - \alpha\varepsilon\nabla\eta)$ is generally negative when $\nabla \cdot \mathbf{u}$ is positive, and vice-versa. By our choice of δ, this means that $\mathcal{L}$ is always non-negative, with a value of zero for divergence-free flow, and a positive value for all other divergence values, in line with the absolute size of the divergence.

In this way, the ψ-weighted dG norm does the following: in areas of zero divergence, we recover the unweighted dG norm. Areas of negative divergence are emphasised, since both ψ and $\mathcal{L}$ will be large there. Areas of positive divergence have reduced H^1- and edge jump-terms, with an L^2 term that is also negatively affected by the weighting, but is emphasised by the $\mathcal{L}$ value.

Thus the ψ-weighted norm reduces the prominence of areas of positive divergence. This reduces the area highlighted to be problematic in the example on an annular domain at the beginning of this chapter.

4.3 Estimator notation and definitions

In this section, we introduce all the notation needed to state the *a posteriori* error estimator.

For each cell K and each edge F we define the following patches, using the notation $\mathrm{cl}\,(K)$ to denote the closure of a cell.

$$\begin{aligned}
\omega_K &\coloneqq \{K' \in \mathcal{T}_h : K' \text{ shares a face with } K\}\,,\\
\omega_F &\coloneqq \{K' \in \mathcal{T}_h : F \subset \partial K'\}\,,\\
\tilde{\omega}_K &\coloneqq \{K' \in \mathcal{T}_h : \mathrm{cl}\,(K') \cap \mathrm{cl}\,(K) \neq \emptyset\}\,,\\
\tilde{\omega}_F &\coloneqq \{K' \in \mathcal{T}_h : \mathrm{cl}\,(K') \cap \mathrm{cl}\,(F) \neq \emptyset\}\,.
\end{aligned}$$

For each facet (either a cell or an edge) ω and for each function $\phi \in L^\infty(\omega)$ defined on it, we define the following notation for the supremum and infimum of the absolute value over the facet ω:

$$\overline{\phi}_\omega \coloneqq \sup_\omega |\phi|, \qquad\qquad \underline{\phi}_\omega \coloneqq \inf_\omega |\phi|.$$

Similarly, for each vector function $\boldsymbol{\phi} \in [L^\infty(\omega)]^d$ defined on it, we define the following notation for the supremum and infimum of the norm of the vector over the facet ω:

$$\overline{\boldsymbol{\phi}}_\omega \coloneqq \sup_\omega \|\boldsymbol{\phi}\|\,, \qquad\qquad \underline{\boldsymbol{\phi}}_\omega \coloneqq \inf_\omega \|\boldsymbol{\phi}\|\,.$$

On each cell, we define the shorthand

$$\lambda_K \coloneqq \begin{cases} \varepsilon^{-\frac{1}{2}} & \text{if } \underline{\psi}_K = \overline{\psi}_K = 1,\\ \max\left\{ \frac{\overline{\nabla\psi}_K}{\sqrt{\underline{\mathcal{L}}_K}}, \frac{\overline{\psi}_K}{\sqrt{\varepsilon}} \right\} & \textit{otherwise.} \end{cases}$$

Then, for each cell K and each edge F, we also define the local weighting functions

$$\begin{aligned}
\rho_K &\coloneqq \frac{1}{\sqrt{\underline{\psi}_K}} \min\left\{ \frac{\overline{\psi}_K}{\sqrt{\underline{\mathcal{L}}_K}}, h_K \lambda_K \right\}, & \rho_{\omega_F} &\coloneqq \min_{K' \in \omega_F} \left\{ \frac{h_{K'}}{\underline{\psi}_{K'}} \lambda_K^2 \right\},\\
\varrho_K &\coloneqq \frac{\lambda_K^2}{\underline{\psi}_K}, & \varrho_{\omega_F} &\coloneqq \max_{K' \in \omega_F} \varrho_{K'}.
\end{aligned} \tag{4.3.1}$$

Note that for $F \in \mathcal{F}_h$ and $K \subset \omega_F$, we have

$$\varrho_K^{-1} \geq \varrho_{\omega_F}^{-1}. \tag{4.3.2}$$

Let now θ_h be the dG approximation obtained from (4.1.5). For each element $K \in \mathcal{T}_h$, we introduce a local error indicator ζ_K which is given by the sum of the three

terms

$$\zeta_K^2 = \zeta_{R_K}^2 + \zeta_{E_K}^2 + \zeta_{J_K}^2,$$

to be defined below.

The first term ζ_{R_K} is the *interior residual* defined by

$$\zeta_{R_K}^2 = \rho_K^2 \left\| f + \varepsilon\Delta\theta_h - \mathbf{u}\cdot\nabla\theta_h - \delta\theta_h \right\|_K^2 .$$

The second term ζ_{E_K} is the *edge residual* defined by

$$\zeta_{E_K}^2 = \frac{1}{2}\sum_{F\in\partial K\setminus\Gamma} \rho_{\omega_F} \left\| [\![\varepsilon\nabla\theta_h]\!] \right\|_F^2 .$$

The last term ζ_{J_K} measures the *edge jumps* of the approximate solution θ_h and is defined by

$$\zeta_{J_K}^2 = \sum_{F\in\partial K} \left(\frac{\sigma\varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F}\sigma\varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 + h_F \|\mathcal{L}\|_{\psi,\tilde{\omega}_F,\infty} + \frac{\overline{\psi}_{\tilde{\omega}_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{\tilde{\omega}_F,\infty}^2 \right) \left\| [\![\theta_h]\!] \right\|_F^2 .$$

We then define the a-posteriori error estimator by

$$\zeta := \left(\sum_{K\in\mathcal{T}_h} \zeta_K^2 \right)^{\frac{1}{2}} . \tag{4.3.3}$$

4.4 Bounding the stationary convection-diffusion-reaction problem

We claim the following bound on the stationary convection-diffusion-reaction problem.

Theorem 4.9. *Let θ be the solution of* (4.1.1)–(4.1.3) *and let $\theta_h \in V_h$ be the solution to the discontinuous Galerkin problem* (4.1.5)*. Let the error estimator ζ be defined*

as in (4.3.3)*. Then we have the a posteriori error bound*

$$|||\theta - \theta_h|||_\psi + |\theta - \theta_h|_{\psi,A} \lesssim \zeta.$$

In the remainder of Section 4.4, we present a proof of Theorem 4.9. We begin with some results that will be necessary in our proof.

We begin with the well-known *inverse inequality* (e.g., [92, 110] for the proof on d-simplices – the result also holds for quadrilaterals and hexahedra). Let $p \geq 1$ be a fixed integer. For any polynomial $v \in \mathcal{P}_p(\omega)$, we have

$$\|v\|_{\partial\omega}^2 \lesssim h_K^{-1} \|v\|_\omega^2 . \tag{4.4.1}$$

Theorem 4.10 (Karakachian-Pascal operator)**.** *Let $V_h^c \coloneqq V_h \cap H_D^1(\Omega)$, the conforming subspace of V_h which satisfies the Dirichlet boundary condition* (4.1.2)*. For any $v_h \in V_h$ there exists a function $C_h(v_h) \in V_h^c$, satisfying*

$$\sum_{K\in\mathcal{T}_h} \|v_h - C_h(v_h)\|_K^2 \lesssim \sum_{F\in\mathcal{F}_h} h_F \|[\![v_h]\!]\|_F^2 ,$$
$$\sum_{K\in\mathcal{T}_h} \|\nabla (v_h - C_h(v_h))\|_K^2 \lesssim \sum_{F\in\mathcal{F}_h} h_F^{-1} \|[\![v_h]\!]\|_F^2 .$$

We refer to $C_h : V_h \to V_h^c$ as the Karakashian-Pascal approximation operator (cf. [61] for the constructive proof of a unique such operator).

We can extend this result to show that, for any positive weight $\phi \in L^\infty(\Omega)$, the following approximation result holds.

Lemma 4.11. *For $v_h \in V_h$,*

$$\sum_{K\in\mathcal{T}_h} \|\phi (v_h - C_h(v_h))\|_K^2 \lesssim \sum_{F\in\mathcal{F}_h} \|\phi\|_{\tilde{\omega}_F,\infty}^2 h_F \|[\![v_h]\!]\|_F^2 ,$$
$$\sum_{K\in\mathcal{T}_h} \|\phi\nabla (v_h - C_h(v_h))\|_K^2 \lesssim \sum_{F\in\mathcal{F}_h} \|\phi\|_{\tilde{\omega}_F,\infty}^2 h_F^{-1} \|[\![v_h]\!]\|_F^2 .$$

In particular, setting $\phi \coloneqq \xi\sqrt{\psi}$*, for a positive function* $\xi \in L^\infty(\Omega)$*, we have: for* $v_h \in V_h$*,*

$$\sum_{K\in\mathcal{T}_h} \|\xi\,(v_h - C_h(v_h))\|^2_{\psi,K} \lesssim \sum_{F\in\mathcal{F}_h} \|\xi\|^2_{\psi,\tilde{\omega}_F,\infty} h_F \,\|[\![v_h]\!]\|^2_F\,,$$
$$\sum_{K\in\mathcal{T}_h} \|\xi\nabla\,(v_h - C_h(v_h))\|^2_{\psi,K} \lesssim \sum_{F\in\mathcal{F}_h} \|\xi\|^2_{\psi,\tilde{\omega}_F,\infty} h_F^{-1} \,\|[\![v_h]\!]\|^2_F\,.$$

Proof. Defining a local analogue of the canonical Karakashian-Pascal operator for each cell, which coincides with the latter on each cell, we can prove a local inequality in the spirit of the Karakashian-Pascal inequality using just the patch of edges touching each cell by at least a vertex. Weighting by ϕ, we pull out the maximum per cell. Summing over all cells, we convert this to a sum over edges by weighting each edge F with the maximum over all cells K such that $F \subset \tilde{\omega}_K$, that is, over all cells in $\tilde{\omega}_F$.

The second part is shown by replacing ϕ by $\xi\sqrt{\psi}$. □

Let I be the identity operator. The L^2-projection $\Pi : V_h + H^1_D(\Omega) \to X_{h,1}$ (the space of linear polynomials as defined by (3.2.1)–(3.2.2)) is defined, for each $w \in V_h + H^1_D(\Omega)$, as the unique $w_h = \Pi w$ such that

$$(w, v_h) = (\Pi w, v_h) \qquad \forall v_h \in X_{h,1}.$$

The L^2-projection satisfies the following:

1. the stability property
$$\|\Pi v\|_\omega \le \|v\|_\omega\,;$$

2. the local estimates
$$\begin{aligned} \rho_K^{-1}\,\|(I-\Pi)\,(\psi v)\|_K &\lesssim |||v|||_{\psi,K}, \\ \rho_{\omega_F}^{-\frac{1}{2}}\,\|(I-\Pi)\,(\psi v)\|_F &\lesssim |||v|||_{\psi,\omega_F}, \end{aligned} \tag{4.4.2}$$
for any $v \in H^1_D(\Omega)$, and any $K, F \subset \mathcal{T}_h$;

3. the global estimates

$$\begin{aligned}\left(\sum_{K\in\mathcal{T}_h}\rho_K^{-2}\left\|(I-\Pi)(\psi v)\right\|_K^2\right)^{\frac{1}{2}} &\lesssim |||v|||_\psi,\\ \left(\sum_{F\in\mathcal{F}_h}\rho_{\omega_F}^{-1}\left\|(I-\Pi)(\psi v)\right\|_F^2\right)^{\frac{1}{2}} &\lesssim |||v|||_\psi,\end{aligned} \tag{4.4.3}$$

for any $v \in H_D^1(\Omega)$.

These bounds on the L^2-projector (see Appendix A for proofs) are based on the following result from [85, 3.5.22]: let $0 \le l \le k$. For any bounded, open set ω with diameter h, and $v \in H^{l+1}(\omega)$,

$$\|v-\Pi v\|_\omega + h\left\|(v-\Pi v)\right\|_{H^1(\omega)} \lesssim h^{l+1}|v|_{H^{l+1}(\omega)}, \tag{4.4.4}$$

and the resulting bound:

$$\varrho_K^{-1}\left\|\nabla(\psi v)\right\|_K^2 \lesssim |||v|||_{\psi,K}^2, \tag{4.4.5}$$

along with two trace inequalities [2]: for $v_h \in V_h$, and for any cell K and edge $F \subset \partial K$,

$$\|v_h\|_F^2 \lesssim h_K^{-1}\|v_h\|_K^2 + \|v_h\|_K\|\nabla v_h\|_K$$

and

$$\|v_h\|_F^2 \lesssim h_K^{-1}\|v_h\|_K^2 + h_K\|\nabla v_h\|_K^2.$$

4.4.1 An inf-sup result

The following inf-sup result will be used to bound the conforming error in Lemma 4.16.

Lemma 4.12. *There is a constant $C > 0$ such that*

$$\inf_{\theta\in H_D^1(\Omega)\setminus\{0\}}\sup_{v\in H_D^1(\Omega)\setminus\{0\}}\frac{a_{reac}(\theta,\psi v)}{\left(|||\theta|||_\psi + |(\mathbf{u}-\alpha\varepsilon\nabla\eta)\,\theta|_{\psi,\star}\right)|||v|||_\psi} \ge C > 0.$$

Proof. Let $w \in H^1_D(\Omega)$ and $\Lambda \in (0,1)$. Then there exists $w_\Lambda \in H^1_D(\Omega)$ such that

$$|||w_\Lambda|||_\psi = 1, \qquad -\int_\Omega (\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w\psi \cdot \nabla w_\Lambda \,\mathrm{d}x \geq \Lambda |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w|_{\psi,\star}.$$

From (4.2.3) we have

$$\begin{aligned} a_{\mathrm{reac}}(w, \psi w_\Lambda) = &\int_\Omega \varepsilon\psi\nabla w \cdot \nabla w_\Lambda \,\mathrm{d}x + \int_\Omega (\delta + (\alpha\nabla\eta - \nabla)\cdot(\mathbf{u} - \alpha\varepsilon\nabla\eta))\, \psi w w_\Lambda \,\mathrm{d}x \\ &- \int_\Omega (\mathbf{u} - \alpha\varepsilon\nabla\eta)\, \psi w \cdot \nabla w_\Lambda \,\mathrm{d}x. \end{aligned}$$

Then, by Lemma 4.7, we obtain

$$\begin{aligned} a_{\mathrm{reac}}(w, \psi w_\Lambda) &\geq \Lambda |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w|_{\psi,\star} - C_1 |||w|||_\psi |||w_\Lambda|||_\psi \\ &= \Lambda |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w|_{\psi,\star} - C_1 |||w|||_\psi, \end{aligned}$$

for some positive constant C_1.

Define $v_\Lambda = w + \frac{|||w|||_\psi}{1+C_1} w_\Lambda$. Obviously, $|||v_\Lambda|||_\psi \leq \left(1 + \frac{1}{1+C_1}\right) |||w|||_\psi$.

So, using Lemma 4.6,

$$\begin{aligned} \sup_{v \in H^1_D(\Omega)\setminus\{0\}} \frac{a_{\mathrm{reac}}(w, \psi v)}{|||v|||_\psi} \geq &\frac{a_{\mathrm{reac}}(w, \psi v_\Lambda)}{|||v_\Lambda|||_\psi} \\ = &\frac{a_{\mathrm{reac}}(w, \psi w) + \frac{|||w|||_\psi}{1+C_1} a_{\mathrm{reac}}(w, \psi w_\Lambda)}{|||v_\Lambda|||_\psi} \\ \geq &\frac{|||w|||_\psi^2 + \frac{|||w|||_\psi}{1+C_1} \left(\Lambda |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w|_{\psi,\star} - C_1 |||w|||_\psi\right)}{\left(1 + \frac{1}{1+C_1}\right) |||w|||_\psi} \\ = &\frac{|||w|||_\psi + \Lambda |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w|_{\psi,\star}}{2 + C_1}. \end{aligned}$$

Since $\Lambda \in (0,1)$ and $w \in H^1_D(\Omega)$ are arbitrary,

$$\inf_{w \in H^1_D(\Omega)\setminus\{0\}} \sup_{v \in H^1_D(\Omega)\setminus\{0\}} \frac{a_{\mathrm{reac}}(w, \psi v)}{\left(|||w|||_\psi + |(\mathbf{u} - \alpha\varepsilon\nabla\eta)\, w|_{\psi,\star}\right) |||v|||_\psi} \geq \frac{1}{2 + C_1} > 0,$$

and the result follows. □

4.4.2 A decomposition of functions from V_h

Following [55, 56] we decompose the discontinuous Galerkin solution into a conforming part and a non-conforming remainder. That is, we write

$$\theta_h = \theta_h^c + \theta_h^d,$$

where $\theta_h^c = C_h(\theta_h) \in V_h^c \coloneqq V_h \cap H_D^1(\Omega)$, with C_h the Karakashian-Pascal approximation operator from Theorem 4.10. The remainder is given by $\theta_h^d \coloneqq \theta_h - \theta_h^c$. We make the observation that $\left[\!\left[\theta_h^d\right]\!\right]_F = \left[\!\left[\theta_h\right]\!\right]_F$ on any edge F. By the triangle inequality we obtain

$$|||\theta - \theta_h|||_\psi + |\theta - \theta_h|_{\psi,A} \leq |||\theta - \theta_h^c|||_\psi + |\theta - \theta_h^c|_{\psi,A} + |||\theta_h^d|||_\psi + |\theta_h^d|_{\psi,A}. \quad (4.4.6)$$

We can now prove that both the nonconforming term θ_h^d and the continuous error $\theta - \theta_h^c$ can each be bounded by the error estimator.

4.4.3 Bounding the nonconforming terms

Lemma 4.13. *There holds*

$$|||\theta_h^d|||_\psi^2 + |\theta_h^d|_{\psi,A}^2 \lesssim \sum_{F\in\mathcal{F}_h} \left(\overline{\psi}_F \frac{\sigma\varepsilon}{h_F} + h_F \|\mathcal{L}\|_{\psi,\tilde{\omega}_F,\infty} + \frac{\overline{\psi}_{\tilde{\omega}_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{\tilde{\omega}_F,\infty}^2 \right) \left\|\left[\!\left[\theta_h\right]\!\right]\right\|_F^2 .$$

Proof. Since $\left[\!\left[\theta_h^d\right]\!\right] = \left[\!\left[\theta_h\right]\!\right]$, we have

$$\begin{aligned} |||\theta_h^d|||_\psi^2 + |\theta_h^d|_{\psi,A}^2 = &\sum_{K\in\mathcal{T}_h} \left(\varepsilon \left\|\nabla\theta_h^d\right\|_{\psi,K}^2 + \left\|\sqrt{\mathcal{L}}\theta_h^d\right\|_{\psi,K}^2 \right) + \left|\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)\theta_h^d\right|_{\psi,\star}^2 \\ &+ \sum_{F\in\mathcal{F}_h} \left(\frac{\sigma\varepsilon}{h_F} + \frac{h_F\|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{F,\infty}^2}{\varepsilon} \right) \left\|\left[\!\left[\theta_h\right]\!\right]\right\|_{\psi,F}^2 . \end{aligned}$$

Lemma 4.11 yields

$$\sum_{K\in\mathcal{T}_h} \varepsilon \left\|\nabla\theta_h^d\right\|_{\psi,K}^2 \lesssim \sigma^{-1} \sum_{F\in\mathcal{F}_h} \frac{\sigma\varepsilon}{h_F}\overline{\psi}_F \left\|[\![\theta_h]\!]\right\|_F^2,$$

and

$$\sum_{K\in\mathcal{T}_h} \left\|\sqrt{\mathcal{L}}\theta_h^d\right\|_{\psi,K}^2 \lesssim \sum_{F\in\mathcal{F}_h} h_F \|\mathcal{L}\|_{\psi,\tilde{\omega}_F,\infty} \left|\left|\left|[\![\theta_h]\!]\right|\right|\right|_F^2.$$

To estimate $|(\mathbf{u}-\alpha\varepsilon\nabla\eta)\,\theta_h^d|_{\psi,\star}$, we apply Lemma 4.11 once more, with the bound (4.2.8), and obtain

$$\begin{aligned}|(\mathbf{u}-\alpha\varepsilon\nabla\eta)\,\theta_h^d|_{\psi,\star}^2 \leq& \frac{1}{\varepsilon}\sum_{K\in\mathcal{T}_h} \|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{\psi,K,\infty}^2 \left\|\theta_h^d\right\|_K^2 \\ \lesssim& \sum_{F\in\mathcal{F}_h} \frac{h_F\overline{\psi}_{\tilde{\omega}_F}\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{\tilde{\omega}_F,\infty}^2}{\varepsilon} \left|\left|\left|[\![\theta_h]\!]\right|\right|\right|_F^2.\end{aligned}$$

Finally,

$$\begin{aligned}&\sum_{F\in\mathcal{F}_h}\left(\frac{\sigma\varepsilon}{h_F}+\frac{h_F\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{F,\infty}^2}{\varepsilon}\right)\left\|[\![\theta_h]\!]\right\|_{\psi,F}^2 \\ &\qquad\leq \sum_{F\in\mathcal{F}_h}\overline{\psi}_F\left(\frac{\sigma\varepsilon}{h_F}+\frac{h_F\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{F,\infty}^2}{\varepsilon}\right)\left\|[\![\theta_h]\!]\right\|_F^2.\end{aligned}$$

Collecting together these bounds and noting that $\overline{\psi}_F \leq \overline{\psi}_{\tilde{\omega}_F}$ yields the result. □

4.4.4 Bounding the conforming terms

The following two lemmas are intermediate results in view of bounding the conforming error, in Lemma 4.16.

Lemma 4.14. *For any $v \in H_D^1(\Omega)$, we have*

$$\begin{aligned}&(f,(I-\Pi)(\psi v)) - a_{reac,h}(\theta_h,(I-\Pi)(\psi v)) \\ &\qquad\lesssim \left(\sum_{K\in\mathcal{T}_h}\left(\zeta_{R_K}^2+\zeta_{E_K}^2\right)\right.\end{aligned}$$

$$+\sum_{F\in\mathcal{F}_h}\left(\varrho_{\omega_F}\frac{\sigma^2\varepsilon^2}{h_F}+\rho_{\omega_F}\|\mathbf{u}\|_{F,\infty}^2\right)\|[\![\theta_h]\!]\|_F^2\Bigg)^{\frac{1}{2}}|||v|||_\psi.$$

Proof. Set

$$T=(f,(I-\Pi)(\psi v))-a_{\mathrm{reac},h}(\theta_h,(I-\Pi)(\psi v)).$$

Then, using $(I-\Pi)\Pi=0$, and after integration by parts,

$$\begin{aligned}T=&\sum_{K\in\mathcal{T}_h}(f+\varepsilon\Delta\theta_h-\mathbf{u}\cdot\nabla\theta_h-\delta\theta_h,(I-\Pi)(\psi v))_K\\&-\sum_{K\in\mathcal{T}_h}\langle\varepsilon\nabla\theta_h\cdot\mathbf{n}_K,(I-\Pi)(\psi v)\rangle_{\partial K}\\&+\sum_{F\in\mathcal{F}_h}\langle\{\varepsilon\nabla\theta_h\},[\![(I-\Pi)(\psi v)]\!]\rangle_F\\&-\sum_{F\in\mathcal{F}_h}\frac{\sigma\varepsilon}{h_F}\langle[\![\theta_h]\!],[\![\Pi(\psi v)]\!]\rangle_F\\&+\sum_{K\in\mathcal{T}_h}\langle\mathbf{u}\cdot\mathbf{n}_K\theta_h,(I-\Pi)(\psi v)\rangle_{\partial_-K\cap\Gamma_D}\\&+\sum_{K\in\mathcal{T}_h}\langle\mathbf{u}\cdot\mathbf{n}_K\lfloor\theta_h\rfloor,(I-\Pi)(\psi v)\rangle_{\partial_-K\setminus\Gamma_D}\\=&T_1+T_2+T_3+T_4+T_5+T_6.\end{aligned}$$

By the Cauchy-Schwarz inequality and (4.4.3),

$$T_1\lesssim\left(\sum_{K\in\mathcal{T}_h}\zeta_{R_K}^2\right)^{\frac{1}{2}}\left(\sum_{K\in\mathcal{T}_h}\rho_K^{-2}\|(I-\Pi)(\psi v)\|_K^2\right)^{\frac{1}{2}}\lesssim\left(\sum_{K\in\mathcal{T}_h}\zeta_{R_K}^2\right)^{\frac{1}{2}}|||v|||_\psi.$$

T_2+T_3 can be written in terms of jumps and averages as

$$T_2+T_3=-\sum_{F\in\mathcal{F}_h}\langle[\![\varepsilon\nabla\theta_h]\!],\{(I-\Pi)(\psi v)\}\rangle_F+\sum_{F\in\mathcal{F}_N}\langle[\![\varepsilon\nabla\theta_h]\!],\{(I-\Pi)(\psi v)\}\rangle_F.$$

The Cauchy-Schwarz inequality and (4.4.3) yield

$$T_2+T_3\lesssim\left(\sum_{F\in\mathcal{F}_h}\rho_{\omega_F}\|[\![\varepsilon\nabla\theta_h]\!]\|_F^2\right)^{\frac{1}{2}}\left(\sum_{F\in\mathcal{F}_h}\rho_{\omega_F}^{-1}\|(I-\Pi)(\psi v)\|_F^2\right)^{\frac{1}{2}}$$

$$\lesssim \left(\sum_{F \in \mathcal{F}_h} \rho_{\omega_F} \left\| [\![\varepsilon \nabla \theta_h]\!] \right\|_F^2 \right)^{\frac{1}{2}} |||v|||_\psi$$
$$\lesssim \left(\sum_{K \in \mathcal{T}_h} \zeta_{E_K} \right)^{\frac{1}{2}} |||v|||_\psi .$$

To bound T_4, we use the interpolant G_h introduced by Clément [28] which is continuous and has the following property:

$$h_K^{-1} \left\| (G_h - I)\, v \right\| \lesssim \left\| \nabla v \right\| . \tag{4.4.7}$$

By applying in turn the Cauchy-Schwarz inequality, (4.3.2), the inverse inequality (4.4.1), triangle inequality, Clément bound (4.4.7), L^2-bound (4.4.2), and (4.4.5), we can show

$$\begin{aligned}
T_4 &= - \sum_{F \in \mathcal{F}_h} \frac{\sigma \varepsilon}{h_F} \langle [\![\theta_h]\!] , [\![\Pi\, (\psi v)]\!] \rangle_F \\
&\lesssim \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma^2 \varepsilon^2}{h_F} \left\| [\![\theta_h]\!] \right\|_F^2 \right)^{\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F}^{-1} h_F^{-1} \left\| [\![(G_h - \Pi)\, (\psi v)]\!] \right\|_F^2 \right)^{\frac{1}{2}} \\
&\lesssim \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma^2 \varepsilon^2}{h_F} \left\| [\![\theta_h]\!] \right\|_F^2 \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \varrho_K^{-1} h_K^{-2} \left\| (G_h - \Pi)\, (\psi v) \right\|_K^2 \right)^{\frac{1}{2}} \\
&\lesssim \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma^2 \varepsilon^2}{h_F} \left\| [\![\theta_h]\!] \right\|_F^2 \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \varrho_K^{-1} h_K^{-2} \left(\left\| (G_h - I)\, (\psi v) \right\|_K^2 \right. \right. \\
&\qquad \left. \left. + \left\| (I - \Pi)\, (\psi v) \right\|_K^2 \right) \right)^{\frac{1}{2}} \\
&\lesssim \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma^2 \varepsilon^2}{h_F} \left\| [\![\theta_h]\!] \right\|_F^2 \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \varrho_K^{-1} \left\| \nabla\, (\psi v) \right\|_K^2 \right)^{\frac{1}{2}} \\
&\lesssim \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma^2 \varepsilon^2}{h_F} \left\| [\![\theta_h]\!] \right\|_F^2 \right)^{\frac{1}{2}} |||v|||_\psi .
\end{aligned}$$

To bound the final terms $T_5 + T_6$, we again use the Cauchy-Schwarz inequality and (4.4.3):

$$\begin{aligned}
T_5 + T_6 &= \sum_{K\in\mathcal{T}_h} \langle \mathbf{u}\cdot\mathbf{n}_K\theta_h, (I-\Pi)(\psi v)\rangle_{\partial_- K\cap\Gamma_D} \\
&\quad + \sum_{K\in\mathcal{T}_h} \langle \mathbf{u}\cdot\mathbf{n}_K\lfloor\theta_h\rfloor, (I-\Pi)(\psi v)\rangle_{\partial_- K\setminus\Gamma_D} \\
&= \sum_{F\in\mathcal{F}_h} \langle [\![\mathbf{u}\theta_h]\!], (I-\Pi)(\psi v)\rangle_F \\
&\lesssim \left(\sum_{F\in\mathcal{F}_h} \rho_{\omega_F} \|[\![\mathbf{u}\theta_h]\!]\|_F^2\right)^{\frac{1}{2}} \left(\sum_{F\in\mathcal{F}_h} \rho_{\omega_F}^{-1} \|(I-\Pi)(\psi v)\|_F^2\right)^{\frac{1}{2}} \\
&\lesssim \left(\sum_{F\in\mathcal{F}_h} \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \|[\![\theta_h]\!]\|_F^2\right)^{\frac{1}{2}} |||v|||_\psi .
\end{aligned}$$

. □

Lemma 4.15. *For any $v \in H_D^1(\Omega)$, the following bound holds:*

$$\begin{aligned}
a_{reac,h}\left(\theta_h^d, \psi v\right) \lesssim \Bigg(\sum_{F\in\mathcal{F}_h} \Bigg(\frac{\sigma\varepsilon}{h_F}\left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F}\varepsilon + \frac{\overline{\psi}_F\alpha^2\varepsilon\overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}}\right) + h_F\|\mathcal{M}\|_{\psi,\tilde{\omega}_F,\infty} \\
+ \frac{h_F}{\varepsilon}\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{\psi,\tilde{\omega}_F,\infty}^2\Bigg)\|[\![\theta_h]\!]\|_F^2\Bigg)^{\frac{1}{2}} |||v|||_\psi ,
\end{aligned}$$

with $\mathcal{M}$ defined as in (4.2.5).

Proof. Recalling the definition of $a_{\mathrm{reac},h}$:

$$\begin{aligned}
a_{\mathrm{reac},h}\left(\theta_h^d, \psi v\right) &= \sum_{K\in\mathcal{T}_h} \left(\varepsilon\nabla_h\theta_h^d, \nabla_h(\psi v)\right)_K - \left(\theta_h^d, \nabla_h\cdot(\mathbf{u}\psi v)\right)_K + \left(\delta\theta_h^d, \psi v\right)_K \\
&\quad - \sum_{F\in\mathcal{F}_h} \left\langle \{\varepsilon\nabla\Pi(\psi v)\}, [\![\theta_h^d]\!]\right\rangle_F \\
&= \sum_{K\in\mathcal{T}_h} \left(\varepsilon\psi\nabla\theta_h^d, \nabla v\right)_K - \left((\mathbf{u}-\alpha\varepsilon\nabla\eta)\psi\theta_h^d, \nabla v\right)_K \\
&\quad + \sum_{K\in\mathcal{T}_h} \left(\mathcal{M}\theta_h^d, \psi v\right)_K
\end{aligned}$$

$$
\begin{aligned}
&- \sum_{F\in\mathcal{F}_h} \left\langle \{\varepsilon\nabla\Pi\left(\psi v\right)\}, \left[\!\left[\theta_h^d\right]\!\right] \right\rangle_F - \sum_{K\in\mathcal{T}_h} \alpha\varepsilon \left\langle \nabla\eta\cdot\mathbf{n}_K\theta_h^d, \psi v\right\rangle_{\partial K} \\
=&S_1+S_2+S_3+S_4+S_5,
\end{aligned}
$$

by the product rule, and integration by parts. By the Cauchy-Schwarz inequality and Lemma 4.11,

$$
\begin{aligned}
S_1 &\leq \left(\sum_{K\in\mathcal{T}_h}\int_K \varepsilon\psi\left|\nabla\theta_h^d\right|^2 \mathrm{d}x\right)^{\frac{1}{2}} \left(\sum_{K\in\mathcal{T}_h}\int_K \varepsilon\psi\left(\nabla v\right)^2 \mathrm{d}x\right)^{\frac{1}{2}} \\
&\leq \left(\sum_{K\in\mathcal{T}_h}\int_K \varepsilon\psi\left|\nabla\theta_h^d\right|^2 \mathrm{d}x\right)^{\frac{1}{2}} |||v|||_\psi \\
&\lesssim \sigma^{-\frac{1}{2}} \left(\sum_{F\in\mathcal{F}_h} \overline{\psi}_{\omega_F}\frac{\sigma\varepsilon}{h_F}\left\|\left[\!\left[\theta_h\right]\!\right]\right\|_F^2\right)^{\frac{1}{2}} |||v|||_\psi.
\end{aligned}
$$

Using the definition of the semi-norm $|\cdot|_{\psi,\star}$, Lemma 4.11 and (4.2.8),

$$
\begin{aligned}
S_2 \leq& |\left(\mathbf{u}-\alpha\varepsilon\nabla\eta\right)\theta_h^d|_{\psi,\star}|||v|||_\psi \\
\leq& \frac{1}{\sqrt{\varepsilon}}\left(\sum_{K\in\mathcal{T}_h}\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{\psi,K,\infty}^2\left\|\theta_h^d\right\|_K^2\right)^{\frac{1}{2}} |||v|||_\psi \\
\lesssim& \left(\sum_{F\in\mathcal{F}_h}\frac{h_F}{\varepsilon}\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{\psi,\tilde{\omega}_F,\infty}^2\left\|\left[\!\left[\theta_h\right]\!\right]\right\|_F^2\right)^{\frac{1}{2}} |||v|||_\psi.
\end{aligned}
$$

By the Cauchy-Schwarz inequality and Lemma 4.11, and since $\mathcal{M}+\delta=2\mathcal{L}$,

$$
\begin{aligned}
S_3 &\leq \left(\sum_{K\in\mathcal{T}_h}\int_K \psi\mathcal{M}\left(\theta_h^d\right)^2 \mathrm{d}x\right)^{\frac{1}{2}} \left(\sum_{K\in\mathcal{T}_h}\int_K \psi\mathcal{M}v^2 \mathrm{d}x\right)^{\frac{1}{2}} \\
&\leq \left(\sum_{K\in\mathcal{T}_h}\int_K \psi\mathcal{M}\left(\theta_h^d\right)^2 \mathrm{d}x\right)^{\frac{1}{2}} \left(\sum_{K\in\mathcal{T}_h}\int_K \psi\left(\mathcal{M}+\delta\right)v^2 \mathrm{d}x\right)^{\frac{1}{2}} \\
&\lesssim \left(\sum_{K\in\mathcal{T}_h}\int_K \psi\mathcal{M}\left(\theta_h^d\right)^2 \mathrm{d}x\right)^{\frac{1}{2}} |||v|||_\psi
\end{aligned}
$$

$$\lesssim \left(\sum_{F \in \mathcal{F}_h} h_F \|\mathcal{M}\|^2_{\psi,\tilde{\omega}_F,\infty} \|[\![\theta_h]\!]\|^2_F \right)^{\frac{1}{2}} |||v|||_\psi.$$

By the Cauchy-Schwarz inequality, the stability of the L^2-projection, an inverse inequality, the triangle inequality (splitting Π into $\Pi - I + I$), the linearity of the gradient operator, and (4.4.4), we have

$$\begin{aligned}
S_4 &\leq \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \int_F \varrho_{\omega_F} \frac{\sigma\varepsilon^2}{h_F} [\![\theta_h]\!]^2 \, \mathrm{d}s \right)^{\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F}^{-1} h_F \int_F \{\nabla \Pi (\psi v)\}^2 \, \mathrm{d}s \right)^{\frac{1}{2}} \\
&\lesssim \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma\varepsilon^2}{h_F} \|[\![\theta_h]\!]\|^2_F \right)^{\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F}^{-1} h_F \int_F \{\nabla (\psi v)\}^2 \, \mathrm{d}s \right)^{\frac{1}{2}} \\
&\lesssim \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma\varepsilon^2}{h_F} \|[\![\theta_h]\!]\|^2_F \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \varrho_K^{-1} \|\nabla (\psi v)\|^2_K \right)^{\frac{1}{2}} \\
&\lesssim \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \varrho_{\omega_F} \frac{\sigma\varepsilon^2}{h_F} \|[\![\theta_h]\!]\|^2_F \right)^{\frac{1}{2}} |||v|||_\psi.
\end{aligned}$$

Finally, by the Cauchy-Schwarz inequality,

$$\begin{aligned}
S_5 &= - \sum_{K \in \mathcal{T}_h} \int_{\partial K} \alpha\varepsilon \nabla\eta \cdot \mathbf{n}_K \psi \theta_h^d v \, \mathrm{d}s \\
&\leq \sum_{F \in \mathcal{F}_h} \int_F \alpha\varepsilon \nabla\eta \cdot [\![\theta_h^d]\!] \, \psi v \, \mathrm{d}s \\
&\leq \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \frac{\sigma\alpha^2\varepsilon^2 \overline{\nabla\eta}^2_F}{h_F \underline{\mathcal{L}}_{\omega_F}} \|[\![\theta_h]\!]\|^2_{\psi,F} \right)^{\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \int_F h_F \underline{\mathcal{L}}_{\omega_F} \psi v^2 \, \mathrm{d}s \right)^{\frac{1}{2}} \\
&\leq \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \frac{\sigma\alpha^2\varepsilon^2 \overline{\nabla\eta}^2_F}{h_F \underline{\mathcal{L}}_{\omega_F}} \|[\![\theta_h]\!]\|^2_{\psi,F} \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \int_K \underline{\mathcal{L}}_K \psi v^2 \, \mathrm{d}x \right)^{\frac{1}{2}} \\
&\leq \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \frac{\sigma\alpha^2\varepsilon^2 \overline{\nabla\eta}^2_F}{h_F \underline{\mathcal{L}}_{\omega_F}} \|[\![\theta_h]\!]\|^2_{\psi,F} \right)^{\frac{1}{2}} \left(\sum_{K \in \mathcal{T}_h} \left\| \sqrt{\mathcal{L}} v \right\|^2_{\psi,K} \right)^{\frac{1}{2}}
\end{aligned}$$

$$\leq \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \frac{\sigma \alpha^2 \varepsilon^2 \overline{\nabla \eta}_F^2}{h_F \underline{\mathcal{L}}_{\omega_F}} \|[\![\theta_h]\!]\|_{\psi,F}^2 \right)^{\frac{1}{2}} |||v|||_\psi$$

$$\leq \sigma^{-\frac{1}{2}} \left(\sum_{F \in \mathcal{F}_h} \frac{\overline{\psi}_F \sigma \alpha^2 \varepsilon^2 \overline{\nabla \eta}_F^2}{h_F \underline{\mathcal{L}}_{\omega_F}} \|[\![\theta_h]\!]\|_F^2 \right)^{\frac{1}{2}} |||v|||_\psi.$$

We note that, of course, in the case of a constant η, such that $\nabla \eta = 0$, we have no S_5 term, and thus in the following results the term $\frac{\overline{\psi}_F \sigma \alpha^2 \varepsilon^2 \overline{\nabla \eta}_F^2}{h_F \underline{\mathcal{L}}_{\omega_F}}$ should then not appear, being treated as 0 rather than $\frac{0}{\underline{\delta}_{\omega_F}}$, which may not be defined in the case $\underline{\delta}_{\omega_F} = 0$. □

We can now state and prove the following bound on the conforming error $\theta - \theta_h^c$:

Lemma 4.16. *There holds:*

$$\begin{aligned}
&|||\theta - \theta_h^c|||_\psi + |\theta - \theta_h^c|_{\psi,A} \\
&\quad \lesssim \Bigg(\sum_{K \in \mathcal{T}_h} \left(\zeta_{R_K}^2 + \zeta_{E_K}^2 \right) \\
&\qquad + \sum_{F \in \mathcal{F}_h} \left(\frac{\sigma \varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F} \sigma \varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla \eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \right. \\
&\qquad\qquad \left. + h_F \|\mathcal{M}\|_{\psi,\tilde{\omega}_F,\infty} + \frac{h_F}{\varepsilon} \|\mathbf{u} - \alpha \varepsilon \nabla \eta\|_{\psi,\tilde{\omega}_F,\infty}^2 \right) \|[\![\theta_h]\!]\|_F^2 \Bigg)^{\frac{1}{2}}.
\end{aligned}$$

Proof. Note that $|\theta - \theta_h^c|_{\psi,A} = |(\mathbf{u} - \alpha \varepsilon \nabla \eta)(\theta - \theta_h^c)|_{\psi,\star}$, cf. (4.2.7). Then, the inf-sup Lemma 4.12 yields:

$$|||\theta - \theta_h^c|||_\psi + |(\mathbf{u} - \alpha \varepsilon \nabla \eta)(\theta - \theta_h^c)|_{\psi,\star} \lesssim \sup_{v \in H_D^1(\Omega) \setminus \{0\}} \frac{a_{\mathrm{reac}}(\theta - \theta_h^c, \psi v)}{|||v|||_\psi},$$

for any $v \in H_D^1(\Omega)$, since $\psi \in W^{1,\infty}(\Omega)$, we have that $\psi v \in H_D^1(\Omega)$. The properties (4.1.6), (4.1.4), and (4.1.5), along with the bilinearity of a_{reac} and $a_{\mathrm{reac},h}$, allow us to

conclude that, for any $v \in H^1_D(\Omega)$,

$$
\begin{aligned}
&a_{\mathrm{reac}}\left(\theta - \theta_h^c, \psi v\right) \\
&= a_{\mathrm{reac}}\left(\theta, \psi v\right) - a_{\mathrm{reac}}\left(\theta_h^c, \psi v\right) \\
&= a_{\mathrm{reac}}\left(\theta, \psi v\right) - a_{\mathrm{reac},h}\left(\theta_h^c, \psi v\right) \\
&= a_{\mathrm{reac}}\left(\theta, \psi v\right) - a_{\mathrm{reac},h}\left(\theta_h, \psi v\right) + a_{\mathrm{reac},h}\left(\theta_h^d, \psi v\right) \\
&= \left(f, \psi v\right) - a_{\mathrm{reac},h}\left(\theta_h, \psi v\right) + a_{\mathrm{reac},h}\left(\theta_h^d, \psi v\right) \\
&= \left(f, \left(I - \Pi\right)\left(\psi v\right)\right) + \left(f, \Pi\left(\psi v\right)\right) - a_{\mathrm{reac},h}\left(\theta_h, \psi v\right) + a_{\mathrm{reac},h}\left(\theta_h^d, \psi v\right) \\
&= \left(f, \left(I - \Pi\right)\left(\psi v\right)\right) + a_{\mathrm{reac},h}\left(\theta_h, \Pi\left(\psi v\right)\right) - a_{\mathrm{reac},h}\left(\theta_h, \psi v\right) + a_{\mathrm{reac},h}\left(\theta_h^d, \psi v\right) \\
&= \left(f, \left(I - \Pi\right)\left(\psi v\right)\right) - a_{\mathrm{reac},h}\left(\theta_h, \left(I - \Pi\right)\left(\psi v\right)\right) + a_{\mathrm{reac},h}\left(\theta_h^d, \psi v\right)
\end{aligned}
$$

Then Lemmas 4.14 and 4.15 already give the result.

□

4.4.5 Completing the bound on the stationary problem

By combining (4.4.6) with Lemmas 4.13 and 4.16, and noting that $\mathcal{M} \lesssim \mathcal{L}$, we have the following bound on the error of the problem (4.1.5).

Theorem 4.17. *Let θ be the solution of* (4.1.1)–(4.1.3) *and θ_h its discontinuous Galerkin approximation, the solution of* (4.1.5)*. Then, the following bound holds:*

$$
\begin{aligned}
&|||\theta - \theta_h|||_\psi + |\theta - \theta_h|_{\psi,A} \\
&\qquad \lesssim \left(\sum_{K \in \mathcal{T}_h} \left(\zeta_{R_K}^2 + \zeta_{E_K}^2\right) \right. \\
&\qquad\quad + \sum_{F \in \mathcal{F}_h} \left(\frac{\sigma\varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F}\sigma\varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F}\|\mathbf{u}\|_{F,\infty}^2 \right. \\
&\qquad\qquad \left. \left. + h_F \|\mathcal{L}\|_{\psi,\tilde{\omega}_F,\infty} + \frac{\overline{\psi}_{\tilde{\omega}_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{\tilde{\omega}_F,\infty}^2 \right) \|[\![\theta_h]\!]\|_F^2 \right)^{\frac{1}{2}}.
\end{aligned}
$$

4.5 An a posteriori error bound for the non-stationary problem

Having shown a bound on the stationary convection-diffusion-reaction problem, we can now use this to tackle the non-stationary convection-diffusion (crucially, with no reaction) problem, using the observation noted previously, that we can rewrite the equation

$$\theta_t - \varepsilon\Delta\theta + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta = f(\mathbf{x},t),$$

in the form of a convection-diffusion-reaction problem simply as

$$\theta_t - \varepsilon\Delta\theta + \mathbf{u}(\mathbf{x},t)\cdot\nabla\theta + \delta\theta = f(\mathbf{x},t) + \delta\theta.$$

Thus, let θ_h be the solution of the semi-discrete problem

$$\sum_{K\in\mathcal{T}_h} \left(\theta_{ht}, v_h\right)_K + a_h\left(\theta_h, v_h\right) = \left(f, v_h\right), \tag{4.5.1}$$

for all $v_h \in V_h$, where, for $w_h, v_h \in V_h + H^1(\Omega)$, we (re)define the bilinear form a_h by

$$\begin{aligned} a_h\left(w_h, v_h\right) \coloneqq & \sum_{K\in\mathcal{T}_h} \left(\varepsilon\nabla_h w_h, \nabla_h v_h\right)_K + \left(\mathbf{u}\cdot\nabla_h w_h, v_h\right)_K \\ & - \sum_{F\in\mathcal{F}_h} \left(\left\langle \varepsilon\left\{\nabla\Pi w_h\right\}, \left[\!\left[v_h\right]\!\right]\right\rangle_F + \left\langle \varepsilon\left\{\nabla\Pi v_h\right\}, \left[\!\left[w_h\right]\!\right]\right\rangle_F - \frac{\sigma\varepsilon}{h_F}\left\langle \left[\!\left[w_h\right]\!\right], \left[\!\left[v_h\right]\!\right]\right\rangle_F \right) \\ & \qquad - \sum_{K\in\mathcal{T}_h} \left(\left\langle \mathbf{u}\cdot\mathbf{n} w_h, v_h\right\rangle_{\partial_- K\cap\Gamma_D} + \left\langle \mathbf{u}\cdot\mathbf{n}_K \lfloor w_h\rfloor, v_h\right\rangle_{\partial_- K\setminus\Gamma_D} \right). \end{aligned}$$

Then θ_h is also the solution to

$$\sum_{K\in\mathcal{T}_h} \left(\theta_{ht}, v_h\right)_K + a_{\mathrm{reac},h}\left(\theta_h, v_h\right) = \left(f + \delta\theta_h, v_h\right),$$

for all $v_h \in V_h$. Thus, we have reframed our convection-diffusion problem in terms of a convection-diffusion-reaction problem, allowing us to use the stationary bound derived in the previous section.

With this observation in place, we turn to the elliptic reconstruction framework of [72] to provide an *a posteriori* error bound on the numerical solution to the non-stationary solution.

4.5.1 Elliptic reconstruction

We define the semi-discrete elliptic reconstruction $w_e \in H_D^1(\Omega)$ to be the solution of

$$a_{\text{reac}}(w_e, v) = (f + \delta\theta_h - \theta_{ht}, v) \quad \forall v \in H_D^1(\Omega). \tag{4.5.2}$$

This means that the elliptic reconstruction w_e is the exact solution to the parabolic PDE whose dG approximation is θ_h [72].

We define the dG version of this, which is to find $w_{e,h} \in V_h$ for all $t \in I$, such that at each time $t \in I$,

$$a_{\text{reac},h}(w_{e,h}, v_h) = (f + \delta\theta_h - \theta_{ht}, v_h) \quad \forall v_h \in V_h.$$

Then since $a_{\text{reac},h}(\cdot, \psi\cdot)$ is coercive over V_h, we have that $w_{e,h} = \theta_h$.

Since w_e and $w_{e,h}$ solve a convection-diffusion-reaction problem and its dG approximation, we can therefore apply the bound in Theorem 4.9 to show that

$$\begin{aligned} &|||w_e - \theta_h|||_\psi + |w_e - \theta_h|_{\psi,A} \\ &\qquad \lesssim \sum_{K\in\mathcal{T}_h} \left(\rho_K^2 \left\|f - \theta_{ht} + \varepsilon\Delta\theta_h - \mathbf{u}\cdot\nabla\theta_h\right\|_K^2 + \zeta_{E_K}^2\right) + \sum_{F\in\mathcal{F}_h} \zeta_{J_K}^2. \end{aligned} \tag{4.5.3}$$

4.5.2 A semi-discrete bound

We use the notation $e = \theta - \theta_h$, and introduce the splitting:

$$e = \rho + \pi \quad \text{with} \quad \rho \coloneqq \theta - w_e, \quad \pi \coloneqq w_e - \theta_h,$$

along with the extra notation $e^c \coloneqq \theta - \theta_h^c$ and $\pi^c \coloneqq w_e - \theta_h^c$.

We introduce the following error estimator terms:

$$\begin{aligned}
\tilde{\zeta}_{S_1}^2 &\coloneqq \sum_{K\in\mathcal{T}_h} \rho_K^2 \left\| f - \theta_{ht} + \varepsilon\Delta\theta_h - \mathbf{u}\cdot\nabla\theta_h \right\|_K^2 \\
&\quad + \sum_{F\in\mathcal{F}_I} \rho_{\omega_F} \left\| [\![\varepsilon\nabla\theta_h]\!] \right\|_F^2 \\
&\quad + \sum_{F\in\mathcal{F}_h} \left(\frac{\sigma\varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F}\sigma\varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \right. \\
&\qquad\qquad \left. + h_F \|\mathcal{L}\|_{\psi,\tilde{\omega}_F,\infty} + \frac{\overline{\psi}_{\tilde{\omega}_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{\tilde{\omega}_F,\infty}^2 \right) \left\| [\![\theta_h]\!] \right\|_F^2 , \\
\tilde{\zeta}_{S_2}^2 &\coloneqq \sum_{F\in\mathcal{F}_h} \min\left\{ \|\mathcal{L}^{-\frac{1}{2}}\|_{\psi,\tilde{\omega}_F,\infty}^2, \frac{\overline{\psi}_{\tilde{\omega}_F}}{\varepsilon} \right\} h_F \left\| [\![\theta_{ht}]\!] \right\|_F^2 , \\
\tilde{\zeta}_{S_3}^2 &\coloneqq \sum_{F\in\mathcal{F}_h} \overline{\psi}_{\tilde{\omega}_F} h_F \left\| [\![\theta_h]\!] \right\|_F^2 .
\end{aligned}$$

We have that θ satisfies

$$(\theta_t, \psi v) + a_{\text{reac}}(\theta, \psi v) = (f + \delta\theta, \psi v) \quad \forall v \in H_D^1(\Omega),$$

so by rearrangement and recalling (4.5.2) we can show that

$$(e_t, \psi v) + a_{\text{reac}}(\rho, \psi v) = (\delta e, \psi v) \quad \forall v \in H_D^1(\Omega).$$

Testing with $v = e^c$, and noting that $e = e^c - \theta_h^d$ and $\rho = e^c - \pi^c$, gives

$$({e^c}_t, \psi e^c) + a_{\text{reac}}(e^c, \psi e^c) = \left(\theta_{ht}^d, \psi e^c\right) + a_{\text{reac}}(\pi^c, \psi e^c) + (\delta e, \psi e^c).$$

In the following, we note that in the case of constant η and $\delta = 0$, we have zero $\mathcal{L}$. In this case, the result carries through in the natural way, resulting in a bound on the quantity

$$\|e\|_{\psi,L^\infty(0,t;L^2(\Omega))}^2 + \int_0^t |||e|||_\psi^2 \, \mathrm{d}s,$$

with the $|||\cdot|||_\psi$ norm containing only an H^1 term.

By the Cauchy-Schwarz inequality, Poincare-Friedrichs inequality, and the coercivity and continuity of $a_{\text{reac}}(\cdot,\cdot)$ (cf. Lemma 4.6 and Lemma 4.8 respectively),

$$\left(\|e^c\|_\psi^2\right)_t + |||e^c|||_\psi^2 \lesssim \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}|||e^c|||_\psi + \left(|||\pi^c|||_\psi + |\pi^c|_{\psi,A}\right)|||e^c|||_\psi + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi \left\|\sqrt{\mathcal{L}}e^c\right\|_\psi.$$

Using Young's inequality, we arrive to

$$\begin{aligned}\left(\|e^c\|_\psi^2\right)_t + |||e^c|||_\psi^2 &\lesssim \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}|||e^c|||_\psi \\ &\quad + \frac{1}{2}\left(|||\pi^c|||_\psi + |\pi^c|_{\psi,A}\right)^2 + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi \left\|\sqrt{\mathcal{L}}e^c\right\|_\psi \\ &\lesssim \left(|||\pi^c|||_\psi + |\pi^c|_{\psi,A}\right)^2 + \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}^2 \\ &\quad + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi^2 + \left\|\sqrt{\mathcal{L}}e^c\right\|_\psi^2 \\ &\lesssim \left(|||\pi^c|||_\psi + |\pi^c|_{\psi,A}\right)^2 + \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}^2 \\ &\quad + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi^2.\end{aligned}$$

By the triangle inequality,

$$\left(\|e\|_\psi^2\right)_t + |||e|||_\psi^2 \lesssim \left(|||\pi|||_\psi + |\pi|_{\psi,A}\right)^2 + \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}^2 + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi^2 + \left(\|\theta_h^d\|_\psi^2\right)_t + |||\theta_h^d|||_\psi^2 + |\theta_h^d|_{\psi,A}^2.$$

By applying a Gronwall inequality [37, Appendix B, p.624] we have that for $t \in I$,

$$\begin{aligned}&\|e\|_{\psi,L^\infty(0,t;L^2(\Omega))}^2 + \int_0^t |||e|||_\psi^2 \, \mathrm{d}s \\ &\lesssim \exp\left(\int_0^t \max_\Omega \frac{\delta^2}{\mathcal{L}}(s)\, \mathrm{d}s\right)\left(\|e(0)\|_\psi^2 + \int_0^t \left(|||\pi|||_\psi + |\pi|_{\psi,A}\right)^2 \, \mathrm{d}s\right.\end{aligned}$$

$$+\int_0^t \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}^2 + \left\|\theta_h^d\right\|^2_{\psi,L^\infty(0,t;L^2(\Omega))} + |||\theta_h^d|||^2_\psi + |\theta_h^d|^2_{\psi,A}\ \mathrm{d}s\Bigg).$$

We can then use (4.5.3), Theorem 4.9, Lemma 4.11 and Lemma 4.13 to show that:

$$\begin{aligned}
&\|e\|^2_{\psi,L^\infty(0,t;L^2(\Omega))} + \int_0^t |||e|||^2_\psi\ \mathrm{d}s \\
&\lesssim \exp\left(\int_0^t \max_\Omega \frac{\delta^2}{\mathcal{L}}(s)\ \mathrm{d}s\right)\Bigg(\|e(0)\|^2_\psi + \int_0^t \tilde{\zeta}^2_{S_1}\ \mathrm{d}s \\
&\qquad + \int_0^t \sum_{F\in\mathcal{F}_h} \min\left\{\|\mathcal{L}^{-\frac{1}{2}}\|^2_{\psi,\tilde{\omega}_F,\infty}, \frac{\overline{\psi}_{\tilde{\omega}_F}}{\varepsilon}\right\} h_F\ \|[\![\theta_{ht}]\!]\|^2_F\ \mathrm{d}s \\
&\qquad + \max_{0\le s\le t}\sum_{F\in\mathcal{F}_h}\overline{\psi}_{\tilde{\omega}_F} h_F\ \|[\![\theta_h(s)]\!]\|^2_F\Bigg).
\end{aligned}$$

Thus, we have the following theorem.

Theorem 4.18 (An a posteriori error bound on the semi-discrete convection-diffusion problem)**.** *Let $e = \theta - \theta_h$ be the difference between the solution θ of the equation* (2.7.1) *and its semi-discrete dG approximate solution satisfying* (4.5.1)*. Then we have the a posteriori error bound*

$$\begin{aligned}
&\|e\|^2_{\psi,L^\infty(0,t;L^2(\Omega))} + \int_0^t |||e|||^2_\psi\ \mathrm{d}s \\
&\lesssim \exp\left(\int_0^t \max_\Omega \frac{\delta^2}{\mathcal{L}}(s)\ \mathrm{d}s\right)\left(\|e(0)\|^2_\psi + \int_0^t \tilde{\zeta}^2_{S_1} + \tilde{\zeta}^2_{S_2}\ \mathrm{d}s + \max_{0\le s\le t}\tilde{\zeta}^2_{S_3}\right).
\end{aligned}$$

4.6 A bound on the discrete problem

We can now discuss the analogous bound for the fully discrete problem.

Let θ_h^i, $i = 0,\ldots,N$ be the solution to

$$\sum_{K\in\mathcal{T}_h^n}\left(\frac{\theta_h^n - \theta_h^{n-1}}{\tau^n}, v_h^n\right)_K + a_{\mathrm{reac},h}\left(\theta_h^n, v_h^n\right) = \left(f^n + \delta^n\theta_h^n, v_h^n\right) \quad \forall v_h^n \in V_h^n. \tag{4.6.1}$$

We define $A^n \in V_h^n$, $n \geq 1$ as the solution of

$$a_{\mathrm{reac},h}\left(\theta_h^n, v_h^n\right) = \left(A^n, v_h^n\right) \quad \forall v_h^n \in V_h^n,$$

and then define the elliptic reconstruction $w^n \in H_D^1(\Omega)$ as the solution of

$$a_{\mathrm{reac}}\left(w^n, v\right) = \left(A^n, v\right) \quad \forall v \in H_D^1(\Omega),$$

where

$$A^n = \Pi^n\left(f^n + \delta^n \theta_h^n\right) - \left(\theta_h^n - \Pi^n \theta_h^{n-1}\right)/\tau^n, \tag{4.6.2}$$

where Π^n is the L^2-projection into $X_{h,1}^n$.

We decompose the dG solution θ_h^n at each timestep into its conforming and nonconforming parts, $\theta_h^{n,c} \in H_D^1(\Omega) \cap V_h^n$ and $\theta_h^{n,d} \in V_h^n$, respectively.

In order to make sense of time integrals, we define $\theta_h(t)$ to be the linear interpolant at intermediate times, that is,

$$\theta_h(t) \coloneqq \ell_n(t)\theta_h^n + \ell_{n-1}(t)\theta_h^{n-1},$$

on the interval $[t^{n-1}, t^n]$, where ℓ_n is the standard linear Lagrange basis function on $[t^{n-1}, t^n]$. Additionally we extend the definition of $\pi^n \coloneqq w^n - \theta_h^n$.

Defining

$$\beta^n \coloneqq \delta^n - \delta + \alpha^n \mathbf{u}^n \cdot \nabla \eta^n - \alpha \mathbf{u} \cdot \nabla \eta - \left(\nabla \cdot \mathbf{u}^n - \nabla \cdot \mathbf{u}\right),$$

we can define the following estimator terms for $n \geq 1$:

$$\begin{aligned}
\zeta_{S_1,n}^2 \coloneqq & \sum_{K \in \mathcal{T}_h^n} \rho_K^2 \left\| A^n + \varepsilon \Delta \theta_h^n - \mathbf{u}^n \cdot \nabla \theta_h^n - \delta^n \theta_h^n \right\|_K^2 \\
& + \sum_{F \in \mathcal{F}_I^n} \rho_{\omega_F} \left\| [\![\varepsilon \nabla \theta_h^n]\!] \right\|_F^2 \\
& + \sum_{F \in \mathcal{F}_h^n} \left(\frac{\sigma \varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F} \sigma \varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla \eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \right. \\
& \qquad \left. + h_F \|\mathcal{L}\|_{\psi, \tilde{\omega}_F, \infty} + \frac{\overline{\psi}_{\tilde{\omega}_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha \varepsilon \nabla \eta\|_{\tilde{\omega}_F, \infty}^2 \right) \left\| [\![\theta_h^n]\!] \right\|_F^2,
\end{aligned}$$

$$
\begin{aligned}
\zeta_{S_2,n}^2 &\coloneqq \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \rho_K^2 \left\| (I-\Pi^n)\left(f^n + \delta^n\theta_h^n + \frac{\theta_h^{n-1}}{\tau^n}\right)\right\|_K^2,\\
\zeta_{S_3,n}^2 &\coloneqq \sum_{F\in\mathcal{F}_h^n} \overline{\psi}_{\tilde{\omega}_F} h_F \left\|[\![\theta_h^n]\!]\right\|_F^2,\\
\zeta_{S_4,n}^2 &\coloneqq \sum_{F\in\mathcal{F}_h^{n-1}\cup\mathcal{F}_h^n} \min\left\{ \|\mathcal{L}^{-\frac{1}{2}}\|_{\psi,\tilde{\omega}_F,\infty}^2, \frac{\overline{\psi}_{\tilde{\omega}_F}}{\varepsilon}\right\} h_F \left\| \left[\!\!\left[\frac{\theta_h^n - \theta_h^{n-1}}{\tau^n}\right]\!\!\right]\right\|_F^2,\\
\zeta_{T_1,n}^2 &\coloneqq \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \varepsilon^{-1}\left\|\ell_n\left(\mathbf{u}^n-\mathbf{u}\right)\theta_h^n + \ell_{n-1}\left(\mathbf{u}^{n-1}-\mathbf{u}\right)\theta_h^{n-1}\right\|_{\psi,K}^2,\\
\zeta_{T_2,n}^2 &\coloneqq \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \Big\|\min\left\{\mathcal{L}^{-\frac{1}{2}}, \varepsilon^{-\frac{1}{2}}\right\}\left(f - f^n + \delta\theta_h - \delta^n\theta_h^n + \ell_{n-1}\left(A^n - A^{n-1}\right)\right.\\
&\qquad \left. + \ell_n\beta^n\theta_h^n + \ell_{n-1}\beta^{n-1}\theta_h^{n-1}\right)\Big\|_{\psi,K}^2.
\end{aligned}
$$

By rearrangement we can show that for $v\in H_D^1(\Omega)$ and $t\in(t^{n-1},t^n]$,

$$
\begin{aligned}
(e_t,\psi v) + a_{\mathrm{reac}}(e,\psi v) &= (\theta_t,\psi v) - (\theta_{ht},\psi v) + a_{\mathrm{reac}}(\theta,\psi v) - a_{\mathrm{reac}}(\theta_h,\psi v)\\
&= (f - f^n + \delta\theta - \delta^n\theta_h^n,\psi v) + (f^n + \delta^n\theta_h^n - \theta_{ht} - A^n,\psi v)\\
&\quad + a_{\mathrm{reac}}(\pi^n,\psi v) + a_{\mathrm{reac}}(\theta_h^n,\psi v) - a_{\mathrm{reac}}(\theta_h,\psi v)\\
&= (f^n + \delta^n\theta_h^n - \theta_{ht} - A^n,\psi v)\\
&\quad + \left(f - f^n + \delta\theta_h - \delta^n\theta_h^n + \ell_{n-1}\left(A^n - A^{n-1}\right),\psi v\right)\\
&\quad + \ell_n a_{\mathrm{reac}}(\theta_h^n,\psi v) + \ell_{n-1}a_{\mathrm{reac}}\left(\theta_h^{n-1},\psi v\right) - a_{\mathrm{reac}}(\theta_h,\psi v)\\
&\quad + \ell_n a_{\mathrm{reac}}(\pi^n,\psi v) + \ell_{n-1}a_{\mathrm{reac}}\left(\pi^{n-1},\psi v\right) + (\delta e,\psi v).
\end{aligned}
\tag{4.6.3}
$$

By using (4.6.2) and the property (4.4.2) we have

$$
\begin{aligned}
(f^n + \delta^n\theta_h^n - \theta_{ht} - A^n,\psi v) &= (f^n + \delta^n\theta_h^n - \theta_{ht} - A^n,(I-\Pi^n)(\psi v))\\
&\lesssim \zeta_{S_2,n}|||v|||_\psi.
\end{aligned}
$$

We can bound the next four terms thus:

$$
\begin{aligned}
&\left(f - f^n + \delta\theta_h - \delta^n\theta_h^n + \ell_{n-1}\left(A^n - A^{n-1}\right),\psi v\right)\\
&\qquad + \ell_n a_{\mathrm{reac}}(\theta_h^n,\psi v) + \ell_{n-1}a_{\mathrm{reac}}\left(\theta_h^{n-1},\psi v\right) - a_{\mathrm{reac}}(\theta_h,\psi v)
\end{aligned}
$$

$$
\begin{aligned}
&= \left(f - f^n + \delta\theta_h - \delta^n\theta_h^n + \ell_{n-1}\left(A^n - A^{n-1}\right), \psi v\right) \\
&\qquad + \ell_n(\beta^n\theta_h^n, \psi v) + \ell_{n-1}\left(\beta^{n-1}\theta_h^{n-1}, \psi v\right) \\
&\quad - \left(\ell_n\left(\mathbf{u}^n - \mathbf{u}\right)\theta_h^n + \ell_{n-1}\left(\mathbf{u}^{n-1} - \mathbf{u}\right)\theta_h^{n-1}, \psi\nabla v\right) \\
&\qquad\qquad \lesssim \zeta_{T_2,n}|||v|||_\psi + \zeta_{T_1,n}|||v|||_\psi.
\end{aligned}
$$

In a similar fashion to the semi-discrete case, by Lemma 4.8 we have

$$
\begin{aligned}
&\ell_n a_{\mathrm{reac}}\left(\pi^n, \psi v\right) + \ell_{n-1} a_{\mathrm{reac}}\left(\pi^{n-1}, \psi v\right) \\
&\lesssim \ell_n^2\left(|||\pi^n|||_\psi + |\pi^n|_{\psi,A}\right)^2 + \ell_{n-1}^2\left(|||\pi^{n-1}|||_\psi + |\pi^{n-1}|_{\psi,A}\right)^2 + |||v|||_\psi^2 \\
&\lesssim \ell_n^2\zeta_{S_1,n}^2 + \ell_{n-1}^2\zeta_{S_1,n-1}^2 + |||v|||_\psi^2.
\end{aligned}
$$

Returning to (4.6.3), and testing with $v = e^c$ we have, via Young's inequality,

$$
\begin{aligned}
\left(e_t, \psi e^c\right) + a_{\mathrm{reac}}\left(e, \psi e^c\right) \lesssim\ & \ell_n^2\zeta_{S_1,n}^2 + \ell_{n-1}^2\zeta_{S_1,n-1}^2 + \zeta_{S_2,n}^2 \\
&+ \zeta_{T_1,n}^2 + \zeta_{T_2,n}^2 \\
&+ |||e^c|||_\psi^2 + (\delta e, \psi e^c),
\end{aligned}
\tag{4.6.4}
$$

thus

$$
\begin{aligned}
\left(\|e^c\|_\psi^2\right)_t + |||e^c|||_\psi^2 &\lesssim \ell_n^2\zeta_{S_1,n}^2 + \ell_{n-1}^2\zeta_{S_1,n-1}^2 + \zeta_{S_2,n}^2 + \zeta_{T_1,n}^2 + \zeta_{T_2,n}^2 \\
&\quad + \min\left\{\left\|\frac{1}{\sqrt{\mathcal{L}}}\left(\theta_h^d\right)_t\right\|_\psi, \left\|\frac{1}{\sqrt{\varepsilon}}\left(\theta_h^d\right)_t\right\|_\psi\right\}^2 + \left(|||\theta_h^d|||_\psi + |\theta_h^d|_{\psi,A}\right)^2 \\
&\quad + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi^2 \\
&\lesssim \ell_n^2\zeta_{S_1,n}^2 + \ell_{n-1}^2\zeta_{S_1,n-1}^2 + \zeta_{S_2,n}^2 + \zeta_{T_1,n}^2 + \zeta_{T_2,n}^2 \\
&\quad + \zeta_{S_4,n}^2 + \zeta_{S_1,n}^2 \\
&\quad + \left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi^2.
\end{aligned}
\tag{4.6.5}
$$

Then, by a completely analogous argument to the semi-discrete case, we have the following theorem.

Theorem 4.19 (An a posteriori error bound on the fully discrete convection-diffusion problem)**.** *Let $e = \theta - \theta_h$ be the difference between the solution θ of the equation* (2.7.1) *and its dG approximate solution satisfying* (4.6.1)*. Then we have the a posteriori error bound*

$$\begin{aligned}
&\|e\|^2_{\psi,L^\infty(0,T;L^2(\Omega))} + \int_0^t |||e|||^2_\psi \,\mathrm{d}s \\
&\lesssim \exp\left(\int_0^t \max_\Omega \frac{\delta^2}{\mathcal{L}}(s)\,\mathrm{d}s\right) \\
&\qquad \left(\|e(0)\|^2_\psi + \sum_{n=1}^N \int_{t^{n-1}}^{t^n} \zeta^2_{S_1,n} + \zeta^2_{S_1,n-1} + \zeta^2_{S_2,n} + \zeta^2_{S_4,n}\,\mathrm{d}s \right. \\
&\qquad\qquad \left. + \sum_{n=1}^N \int_{t^{n-1}}^{t^n} \zeta^2_{T_1,n} + \zeta^2_{T_2,n}\,\mathrm{d}s + \max_{0\le n\le N} \zeta^2_{S_3,n}\right).
\end{aligned} \tag{4.6.6}$$

It is worthwhile here highlighting the effect that the use of the Gronwall inequality may have upon the sharpness of the resulting bound, and upon the efficacy of the resulting error indicator as an adaptivity indicator. By splitting the $\left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi$ term apart, we lose the local dependence of the inequality upon $\frac{\delta}{\sqrt{\mathcal{L}}}$. This reduces the sharpness of the bound in some cases. This in turn means that the possibility exists that the resulting error indicator may not generate local values that directly correspond in order to the local contribution to the true error. However, we still consider the error indicator to, in practical usage, be a good choice. Consider again (4.6.5): this is an *a priori* bound on the error before the time integration step. On the right hand side are a number of terms, including the term $\left\|\frac{\delta}{\sqrt{\mathcal{L}}}e\right\|_\psi$. Unless this is the dominant term locally, then most of the information is encoded in the remaining terms on the right hand side. In this case, these terms will act as a good adaptivity indicator. In other cases, it is possible that this term will be dominant, and thus the adaptivity indicator will not act in an optimal manner. In this case, it may instead rank cells in an order different to their local contribution to error.

4.7 Parameter choices

We now comment upon the choice of α and δ, which are as yet unchosen.

We know from Lemma 4.7 that we require

$$\delta(\mathbf{x}) \geq \max\left\{0, -2\left(\alpha\nabla\eta - \nabla\right) \cdot \left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right)(\mathbf{x})\right\}, \tag{4.2.9}$$

to assert continuity. Since (4.6.6) contains an exponential term of $\max_\Omega\left(\frac{\delta^2}{\mathcal{L}}\right)$, it is of paramount importance to reduce the value of δ wherever possible. Thus, choosing δ to be equal to this quantity is a sensible manner in which to ensure continuity while also minimising the use of added reaction, and thus minimising the exponential term.

The correct choice of α is less clear. Dimensional analysis shows that α must have units of seconds per metre, but further analysis and calibration of the choice of value has not been undertaken. However, two main concerns should guide further work upon this subject.

Firstly, as above, we wish to reduce the required δ needed wherever possible. In some circumstances, a judicious choice of the value of α may lead to the method requiring no δ anywhere, in which case no exponential term will be incurred (see Section 4.8).

Secondly, the choice of α affects the weight ψ, and thus the weighted norm upon which we provide an error bound. It also affects the value of $\mathcal{L}$. Through these quantities, an injudicious choice of α may have the undesirable effect of weighting the norm in too extreme a way, such that the derived error bound is not useful for our purposes. If a very large value of α is used, such that the weight $\psi = \exp(-\alpha\eta)$ is very small in most areas, and a larger value in only a small area, then the resulting norm informs us little about the global behaviour of the solution.

It is, thus, clear that the optimal choice of α is a non-trivial problem, and is dependent upon the desired weighted norm which we wish to bound, and upon the behaviour of the given convection field.

However, it is also clear that the validity of the results of this chapter rely purely on a choice of constant $\alpha \in \mathbb{R}^+$, and δ satisfying (4.2.9).

4.8 Relation to existing results

Given the bound (4.6.6), we can now relate this to existing results, and show it to be an extension of the existing literature.

Firstly, we consider the first example given in this chapter, with an imposed negative-divergence flow field $\left(1, \frac{1}{2} - \frac{1}{2}y - x\right)^\intercal$ on the unit box, illustrated in Figure 4.1. Since this flow is characterised (using the shorthand for z-independent vector fields as defined in Remark 4.1) by

$$\mathbf{u} = \nabla\left(x - \frac{y^2}{4}\right) + \mathbf{curl}\left(-x + \frac{x^2}{2} + y\right),$$

we have that

$$(\alpha\nabla\eta - \nabla)\cdot(\mathbf{u} - \alpha\varepsilon\nabla\eta) \geq 1 - \frac{3}{2}\varepsilon,$$

on the box domain $[0,1]^2$. Thus, we do not require to add an artificial reaction term, for a small ε, instead deriving an error bound in a weighted dG norm, with

$$\psi = \exp\left(-\alpha\left(x - \frac{y^2}{4}\right)\right),$$

which we may view as an alternative bound to that proven in [27].

More strongly, if we are able to say that the field is exactly the curl of another field, i.e., $\nabla\eta = 0$, then, since we can choose our constant, we choose $\eta = 0$, and thus we have $\psi = 1$. So, we recover the unweighted norms, with the L^2 term equal to δ, which we may choose to be zero if we wish. Thus for the case where the convection is exactly a curl (i.e. it is divergence-free), the equation can be written in divergence form as

$$-\varepsilon\Delta\theta + \nabla\cdot(\mathbf{u}(\mathbf{x})\theta) = f(\mathbf{x}).$$

Returning to (4.6.4), we can see that setting $\delta = 0$ here removes the need for the use of the Gronwall result, and the resulting addition of an exponential term, and so we recover the bound of [27] for this case.

As a general conclusion, the above analysis improves upon and refines known results, while offering the possibility of reduced dependence upon the "worst case" Gronwall constant for a number of relevant scenarios.

4.9 Conclusions

In this chapter, we have presented the derivation of a new error estimate for the convection-diffusion problem, under very general assumptions upon the convection field. By assuming only that $\mathbf{u} \in [W^{1,\infty}(\Omega)]^d$, with bounded divergence, we are unable to guarantee the validity of the usual assumption that there exists a constant $\gamma_0 > 0$ such that

$$-\frac{1}{2}\nabla \cdot \mathbf{u}(\mathbf{x}, t) + b(\mathbf{x}, t) > \gamma_0.$$

Instead, we use an exponential-fitting technique, combined with some added reaction term, to prove an *a posteriori* error bound in a modified norm. The combination of the two techniques gives us the ability to prove a bound on all such flows, while minimising the incurred penalty of a Gronwall-type exponential factor where possible. This means that for some flows we are able to recover previously known results, while we also prove bounds upon problems with a broader class of convection fields.

The flexibility of this combined approach is a novel and useful tool for the study of geodynamic flows, in particular mantle convection. The complexity and nonlinearity of these systems mean that *a priori* knowledge of the flow characteristics is often extremely limited. The ability to handle a wide class of convection fields with a single technique allows the use of this method in simulations, especially as an *a posteriori* adaptivity indicator, without overly restricting the (*a priori* unknown) permissible flows.

The implementation of the calculation of the error bound, and the implementation of a simplified version for use as an adaptivity indicator, are presented in the following chapters. Chapter 5 presents an implementation of the full error bound and indicator within a small code. This allows the exploration of the behaviour of the bound and indicator. Chapter 6 presents the implementation of the adaptivity indicator within a larger, community-maintained code, which enables the exploration of the behaviour, and utility, of the adaptivity indicator under a wide range of conditions that are of interest to geodynamists.

Chapter 5

Parallel implementation

In this chapter, we discuss the implementation of a mantle convection simulation finite element code to numerically approximate the system (2.5.1) using the solution process outlined in Algorithm 3.1, with a focus on the parallelisable nature of the implementation.

In particular, this includes an explanation of the implementation of the dG and FE methods in parallel, as well as the implementation of an adaptivity routine using a cellwise error indicator based on the error estimate derived in Chapter 4. This is used to direct the finite computational resources in such a manner as to maximise the accuracy of the simulation while minimising the computational cost. Finally, it includes details of the implementation of the full error estimate from Chapter 4, in order to explore the behaviour of this bound under certain conditions.

We use the deal.II library [15] to build the dG/FE implementation in parallel. In choosing a library to build upon, we consider as essential the accessibility of the codebase and its possible modification. We require the ability to use dG and FE methods simultaneously, including mixed methods for the Stokes element. In addition, we wish to be able to implement a scheme making use of multiple processors, with the ability to scale to large numbers of processors in a reasonable manner. In order to be able to assess the behaviour of a new adaptivity indicator, we require the ability to calculate such, and to be able to use this to drive an h-adaptive mesh refinement strategy. Finally, a well-documented, widely-tested and well-maintained

library is essential, in order to have a ready understanding of the library and its modification, and reasonable trust in the correctness of the results.

The library chosen for our implementation is the deal.II library, which is built in the C++ language. It interfaces to the `p4est` library [24] for its distributed-mesh capabilities, and the Trilinos [53, 54] and PETSc [14, 13, 12] libraries for solvers and distributed linear algebra. It utilises the MPI (Message Passing Interface) framework within these and within the main library to facilitate parallelism across multiple nodes, and the Threading Building Blocks library for parallelising many operations across cores within a node.

The deal.II library satisfies the above requirements, and so is an ideal fit for such a project. In addition, it contains the step-32 tutorial code, which models mantle convection. This is also the basis for a much more advanced mantle simulator, built using deal.II, called ASPECT, which uses enhanced techniques to simulate mantle convection in a modular, extensible fashion, with a focus upon ensuring the code is usable by geophysicists in research. This is another strength of the choice of deal.II as our underlying library for this application, and in fact, Chapter 6 deals with a further implementation of some of the results of Chapter 4 into the ASPECT codebase.

For the purposes of this current chapter, we present here a basic summary of the characteristics of the original step-32 tutorial code. step-32 implements a parallel Boussinesq system approximation, with the three primary variables of temperature, velocity and pressure each discretised by the FE method. It follows a variant of Algorithm 3.1, which is identical for our purposes, and uses an entropy viscosity method to stabilise the convection-dominated temperature equation.

The core modifications to this code here are an implementation of the dG method for the temperature variable, and an implementation of the *a posteriori* error bound derived in Chapter 4, along with the use of an adaptivity indicator based upon the *a posteriori* error bound. This is to explore the behaviour of the error bound and its handling of non-zero divergence velocity fields, and the behaviour of the adaptivity indicator, as well as to demonstrate the usage of the dG method to compute it.

5.1 A bird's-eye view of the distributed finite element infrastructure of deal.II

We now introduce the basic infrastructure and associated terminology for the deal.II library and its functionality in providing for distributed processing: the `typewriter` font denotes deal.II classes or functions.

A `parallel::distributed::Triangulation` class provides the underlying framework of the mesh. This stores information about the cells and edges of the mesh, without exposing the internal information of how this is stored and computed to the end-user. Instead, it allows access to *iterators* over the cells and edges. In partnership with the `p4est` library, it uses a Z space-filling curve algorithm [81] to order the cells, and to split these cells' ownership between the processors. Thus exactly one process owns each cell, and so, from the processor's point of view, we can refer to the set of *locally-owned cells*. In addition to this set, each processor also has access to a single layer of *ghost* cells around the cells it owns. In this way, every edge of a locally-owned cell will either (a) be a boundary edge; (b) have a locally-owned neighbour behind it; (c) have a ghost cell behind it. The combined set of cells which are locally-owned or ghost cells of a given processor are referred to as *locally-relevant* cells.

This framework for the triangulation allows the library to distribute ownership of the degrees of freedom (DoFs) of a finite element space defined over the triangulation, by means of a `DoFHandler` class. All DoFs associated to a given cell or edge are owned by the processor claiming ownership of the cell or edge, and are given a local numbering. DoFs on edges at the boundaries between processors are owned by just one of the processors, decided by an ordering. Each processor has access to the DoFs associated to all *locally-relevant* cells. This requires communication between processors, which, on the largest machines, can be a bottleneck in the solution process, but is infinitely preferable to communication of the data for the full problem.

With the DoFs distributed between the processors, we are then able to define *distributed vectors*, which hold the value of each DoF. A distributed vector contains values on only *locally-owned* or *locally-relevant* DoFs, with the choice between these

two options determined by the user, based on the purpose of the vector in the computation.

Finally, there exists functionality to communicate changes to the mesh across processors. In particular, the effects of local refinement and coarsening are handled such that the representation of the mesh is consistent across processors after these operations. In addition, vectors defining finite element fields across the domain are interpolated after these operations, to ensure the result approximates the original field in the new finite element space. Lastly, either during or after these adaptivity operations, the `repartition` function allows the number of cells owned per processor to be approximately equilibrated, with the information held in the finite element vectors communicated as appropriate to ensure data on each processor matches the ownership patterns of the new configuration. This load-balancing operation is a crucial step in the algorithm to ensure processors share work evenly amongst themselves, and is handled by the deal.II library.

Through the above framework, the full problem is able to be distributed across many processors. Each processor owns only a portion of the data associated to a physical portion of the computational domain, and has knowledge of an area slightly larger than the owned area, just enough to facilitate the assembly and solution process.

5.2 An introduction to step-32

The main outcome of this chapter is a modified code based upon step-32, which implements a combination of conforming FE and dG methods to build a mantle simulator with a rigourously derived error estimator for the convection-diffusion equation, and a subsequent error indicator for use in driving the h-adaptivity of the scheme. It is built to be parallelised in the sense explained above, with a focus on calculating as much as possible in a local manner. This code can be viewed as a stepping-stone to a fully-dG code with an associated error bound and error indicator that would take into account the full nonlinearity of the interdependence between the Stokes and convection-diffusion parts. While some of the largest benefits of the dG method would be realised by implementing the Stokes system in dG, the current code explores the ability of the error estimate, derived in Chapter 4, to

handle convection fields with small, but potentially positive, divergence. This is a necessary bridging step in our understanding towards a fully non-linear *a posteriori* estimate upon the error of the full system, and an error indicator that is able to direct h-refinement taking into account the full problem.

The basic outline of the modified code is presented in Algorithm 5.1, although the final form is, ultimately, significantly modified in order to implement the full error bound calculation (see Section 5.6).

Algorithm 5.1

while stopping criterion is not satisfied **do**
    Assemble dG system for temperature
    Solve temperature system
    Assemble FEM system for Stokes
    Solve Stokes system
    Calculate error bound, and indicators for adaptivity
    Apply adaptivity algorithm to mesh
    Step forwards in time
end while

In the following, we present details of the implementation of several of these steps.

5.3 Assembly and solution for Stokes

The Stokes equations are solved using a Taylor-Hood mixed finite element scheme, which is implemented in the deal.II code for step-32 [67, 66], upon which this code is built. The implementation described below is largely unchanged from this original implementation, and the description here is provided for completeness and for comparison against the dG implementation used for the convection-diffusion equation.

The system of equations (3.4.2) for the discretised Stokes system is written in matrix form as:

find vector $\mathbf{X} = \begin{pmatrix} U \\ P \end{pmatrix} \in \mathbb{R}^{m\times 1}$ satisfying

$$S\mathbf{X} = \begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} G_U \\ 0 \end{pmatrix} = \mathbf{G},$$

where the matrix block A corresponds to the bilinear form $s_h(\cdot,\cdot)$ and the block B corresponds to $b_h(\cdot,\cdot)$.

Due to the simplicity of the FEM scheme, this is relatively easy to assemble. All terms are defined as integrals over the cells of the scheme, with continuity over edges guaranteed by the choice of the finite element scheme. Assembly of these terms is implemented by looping over all degrees of freedom associated to each cell, and all quadrature points on each cell, weighting each appropriately to form a numerical integration scheme that evaluates the integral.

Efficient solution of this system is an area of ongoing research. Preconditioning of this system is extremely difficult in many cases beyond the simple isoviscous setting. The implemented scheme leverages the inherent block structure of the problem, and utilises preconditioners on the separate blocks within a Schur complement solution scheme. Details of this process are given in [66, 67].

5.4 Assembly of the discontinuous Galerkin terms

A feature of the dG method is the large number of terms in the bilinear form $a_h(\cdot,\cdot)$ of equation (3.3.3). This complexity translates into a complicated assembly procedure for the matrix M corresponding to this bilinear form.

We begin by splitting the bilinear forms on the left-hand side of (3.3.3) into four subforms in the following manner:

$$a_h(\theta_h^n, v_h) = a_{h,M}(\theta_h^n, v_h) + a_{h,S}(\theta_h^n, v_h) + a_{h,C}(\theta_h^n, v_h) + a_{h,O}(\theta_h^n, v_h),$$

where

$$\begin{aligned}
a_{h,M}(\theta_h^n, v_h) &\coloneqq \frac{1}{\tau^n} \sum_{K\in\mathcal{T}_h} (\theta_h^n, v)_K, \\
a_{h,S}(\theta_h^n, v_h) &\coloneqq \sum_{K\in\mathcal{T}_h} (\varepsilon\nabla\theta_h^n, \nabla v)_K, \\
a_{h,C}(\theta_h^n, v_h) &\coloneqq \sum_{K\in\mathcal{T}_h} (\mathbf{u}\cdot\nabla\theta_h^n, v)_K,
\end{aligned}$$

$$a_{h,O}\left(\theta_h^n, v_h\right) \coloneqq \sum_{F\in\mathcal{F}_h} -\langle \varepsilon\left\{\nabla\theta_h^n\right\}, [\![v_h]\!]\rangle_F + \Upsilon\langle \varepsilon\left\{\nabla v_h\right\}, [\![\theta_h^n]\!]\rangle_F + \frac{\sigma\varepsilon}{h_F}\langle [\![\theta_h^n]\!], [\![v_h]\!]\rangle_F$$
$$- \sum_{K\in\mathcal{T}_h} \left(\left\langle (\mathbf{u}\cdot\mathbf{n})(\theta_h^n)^+, v_h^+ \right\rangle_{\partial_- K\cap\Gamma_D} + \left\langle (\mathbf{u}\cdot\mathbf{n}_K)\lfloor\theta_h^n\rfloor, v_h^+ \right\rangle_{\partial_- K\setminus\Gamma_D} \right).$$

This can be viewed as being equivalent to decomposing the matrix M into its four submatrices, namely

$$M = M_1 + M_2 + M_3 + M_4.$$

We have that M_1, M_2 and M_3 are sums of integrals over cells, while M_4 consists of sums of integrals over edges.

The first three matrices are relatively easy to implement, even in a distributed setting, since cell integrals are an inherently asynchronous operation, allowing computation in parallel in a natural manner. Since all cells will be owned by exactly one processor, it is enough to allow each processor to assemble the terms for each cell it owns, before using this local matrix to update the global matrix.

The final matrix, M_4, is significantly more complicated to assemble, which we shall now explore.

5.4.1 Edge ownership algorithm

Since M_4 consists of integrals over edges, we require an algorithm to determine the iteration process over all edges in the mesh.

Due to the data structures used in deal.II, edges are accessed through their bordering cells. Since all internal edges are shared by two cells, an iteration over the cells of the triangulation will entail a double-iteration over all internal edges. There are two strategies to cope with this double-iteration: the first is to use both occurrences of access to the same edge, and to assemble only half the integral term from each side of the edge. This is a fairly natural way to build jumps across edges – summing from each side in turn, with the outward-pointing normal from each cell, will yield the jump as required. However, on an adaptively refined mesh this causes some issues in the presence of hanging nodes. Since an edge on a refined neighbour corresponds

to only part of an edge on a coarse cell, the approach of integrating from both sides is less natural.

Instead, we take a second approach, whereby all edge integrals are computed from one side. This requires an algorithm to determine which cell should take ownership of the integration task on each edge, to ensure that edges are treated just once. The algorithm presented in Figure 5.1 handles this choice.

To explain this algorithm, it is necessary to understand the way cells are stored over differing refinement levels and differing processors.

The triangulation is based upon a *coarse triangulation*. This has no hanging nodes and all cells are on the same base 'level'. The full triangulation is built up by refinement from this coarse level. Since refinement happens by splitting each quadrilateral into four in an isotropic manner (or each hexahedra into 8 sub-hexahedra), we have a forest-of-quadtrees (respectively, octrees) structure to the mesh. In the case of a single processor, a cell's *level* is defined as the number of refinements that are required to travel from the coarse mesh to the cell. Within the set of all cells at a given level, a unique *index* is assigned to each cell, thus providing a unique identity for the cell by means of the (level, index) pair. In contrast, on a distributed mesh, the triple (subdomain id, level, index) is required to uniquely identify a cell, since the pairing (level, index) is not unique over the whole mesh, but only within a single processor.

The algorithm for choosing ownership of edges is then as follows. Iterating over all cells, we inspect each edge of that cell in turn. For any edge F, if F is on the boundary then it borders just the current cell, which takes ownership of that edge. Otherwise, we have an interior edge and we have the following options:

1. the edge contains a hanging node, as the neighbouring cell is more refined;
2. the edge forms only part of the neighbour's edge, as the current cell is more refined;
3. the edge forms the whole of both the current cell's edge and neighbour's edge, with both in the same subdomain;

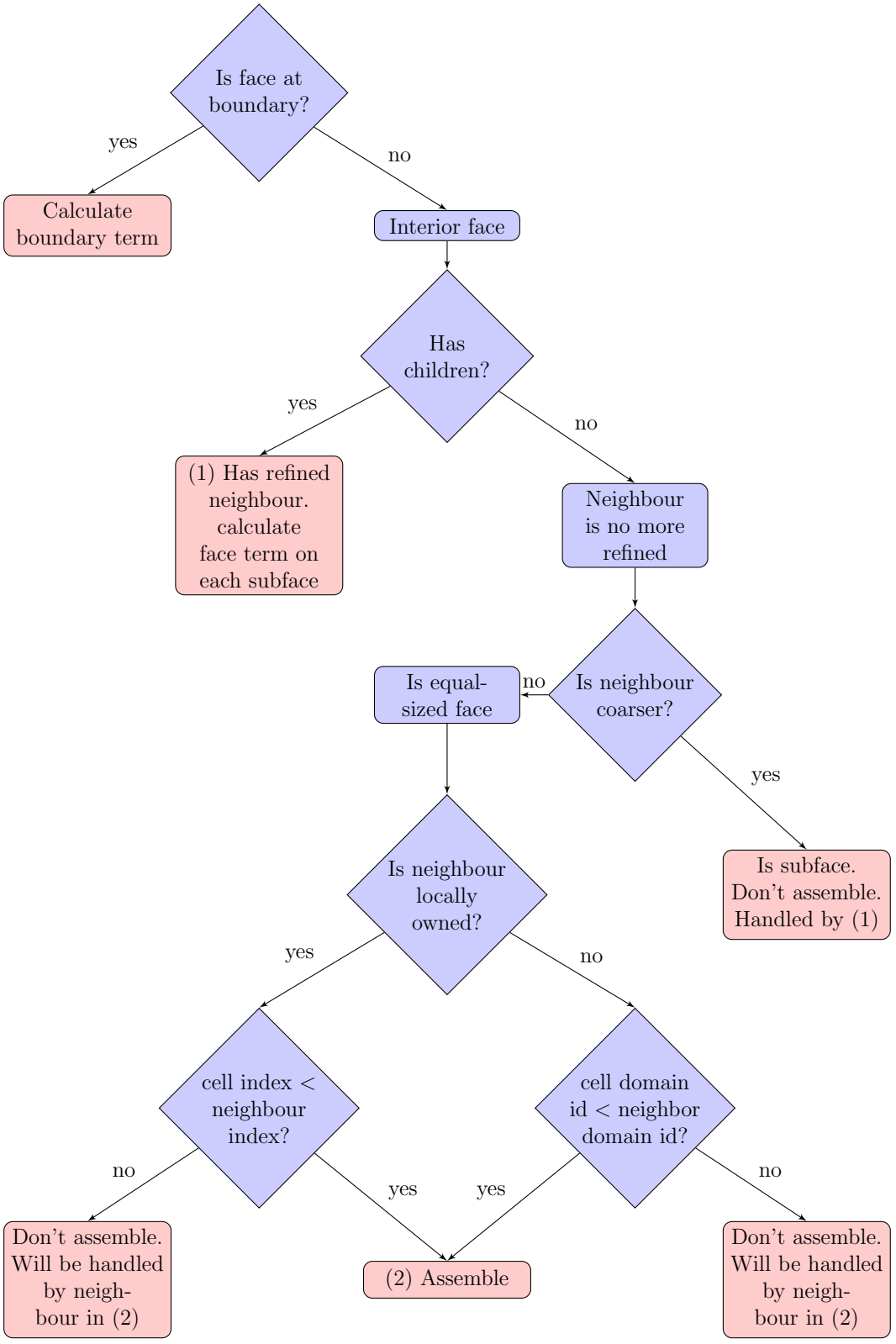


FIGURE 5.1: An algorithm for determining the ownership of edges in a distributed mesh.

4. the edge forms the whole of both the current cell's edge and neighbour's edge, and the current cell and neighbour are in distinct subdomains.

In the first case, the current cell takes ownership of the edge, while in the second it ignores the edge. In this manner, all edges between cells on differing levels will be owned by the larger cell.

In the third case, we still need a deterministic choice on which cell should take ownership. Since the two cells are both on the same subdomain, then they must have differing indices: the lower index is to take ownership.

In the final case where the two cells are not on the same subdomain, they may have identical indices and levels. Thus, we instead compare the subdomain ID of each subdomain, allowing the cell on the subdomain with lower ID to form the integral over the edge.

5.5 WorkStream

We describe very briefly here the use of the `WorkStream` class [103] in parallelising the assembly of the matrices and vectors across cores. The `WorkStream` class utilises the fact that assembly of matrices and vectors involves iterations over cells and edges, with local contributions from each cell and each edge then combined into a global object. `WorkStream` enables deal.II to share the task of assembling local contributions from cells and edges owned by a given processor between as many processors as are available in the shared-memory setting of this processor. It also implements a graph colouring strategy to add local contributions to the global objects in an efficient and scalable manner. To the end user, this process is nearly transparent, requiring only the use of scratch and data objects to package the necessary data. This is particularly useful in the case of complicated coefficients and equations, which reduce the ability to parallelise via GPUs, which do not handle complicated or branching code efficiently.

5.6 Error indicator and bound

In order to drive an h-adaptivity algorithm, we need some criteria to decide which cells should be refined or coarsened at a given timestep. We follow the standard approach of using a cell-wise *error indicator* in the following manner.

Typically, adaptive mesh simulations utilise the logical loop in Algorithm 5.2. That is, after solving the problem on a given mesh and time step, the solution is used to generate error indicators for each cell or edge. These indicators are used to mark cells or edges for refinement (or coarsening), after which the marked cells are refined (resp. coarsened), to generate a new mesh. This new mesh is then used to solve the problem, and the cycle repeats until some given stopping criterion is satisfied.

Algorithm 5.2

while stopping criteria is not satisfied **do**
  Solve system
  Calculate indicators
  Mark cells for refinement (or coarsening)
  Refine (resp. coarsen) marked cells to generate new mesh
end while

In order to validate the theory of the *a posteriori error estimate* proven in (4.6.6), we would like to be able to compute the estimate, or its approximation. This is in addition to the error indicator which drives the adaptivity scheme.

Let us recall the various terms in the error estimate (4.6.6) that will need to be computed for each timestep $n \in \{0, \dots, N\}$. The spatial estimator terms are:

$$\begin{aligned}\zeta_{S_1,n}^2 = &\sum_{K\in\mathcal{T}_h^n} \rho_K^2 \left\| A^n + \varepsilon\Delta\theta_h^n - \mathbf{u}^n\cdot\nabla\theta_h^n - \delta^n\theta_h^n \right\|_K^2 \\ &+ \sum_{F\in\mathcal{F}_I^n} \rho_{\omega_F} \left\| [\![\varepsilon\nabla\theta_h^n]\!] \right\|_F^2 \\ &+ \sum_{F\in\mathcal{F}_h^n} \left(\frac{\sigma\varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F}\sigma\varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \right. \\ &\qquad \left. + h_F \|\mathcal{L}\|_{\psi,\tilde{\omega}_F,\infty} + \frac{\overline{\psi}_{\tilde{\omega}_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{\tilde{\omega}_F,\infty}^2 \right) \left\| [\![\theta_h^n]\!] \right\|_F^2 ,\end{aligned}$$

$$\zeta_{S_2,n}^2 = \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \rho_K^2 \left\| (I-\Pi^n)\left(f^n + \delta^n\theta_h^n + \frac{\theta_h^{n-1}}{\tau^n}\right)\right\|_K^2 ,$$

$$\zeta_{S_3,n}^2 = \sum_{F\in\mathcal{F}_h^n} \overline{\psi}_{\tilde{\omega}_F} h_F \left\| [\![\theta_h^n]\!]\right\|_F^2 ,$$

$$\zeta_{S_4,n}^2 = \sum_{F\in\mathcal{F}_h^{n-1}\cup\mathcal{F}_h^n} \min\left\{ \|\mathcal{L}^{-\frac{1}{2}}\|_{\psi,\tilde{\omega}_F,\infty}^2, \frac{\overline{\psi}_{\tilde{\omega}_F}}{\varepsilon}\right\} h_F \left\| \left[\!\!\left[\frac{\theta_h^n - \theta_h^{n-1}}{\tau^n}\right]\!\!\right]\right\|_F^2 .$$

The temporal estimator terms, which will be integrated over time, are:

$$\zeta_{T_1,n}^2 = \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \varepsilon^{-1} \left\| \ell_n\left(\mathbf{u}^n - \mathbf{u}\right)\theta_h^n + \ell_{n-1}\left(\mathbf{u}^{n-1}-\mathbf{u}\right)\theta_h^{n-1}\right\|_{\psi,K}^2 ,$$

$$\zeta_{T_2,n}^2 = \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \left\| \min\left\{ \mathcal{L}^{-\frac{1}{2}}, \varepsilon^{-\frac{1}{2}}\right\} (f - f^n + \delta\theta_h - \delta^n\theta_h^n \right.$$
$$\left. + \ell_{n-1}\left(A^n - A^{n-1}\right) + \ell_n\beta^n\theta_h^n + \ell_{n-1}\beta^{n-1}\theta_h^{n-1}\right)\right\|_{\psi,K}^2 .$$

A number of the quantities in these estimator terms are standard, and are computable (up to an approximation for patchwise-defined quantities, see Section 5.6.6) from the solution pair $(\theta_h^n, \mathbf{u}_h^n)$ defined on the triangulation $\mathcal{T}_h^n$. Their computation is by a process identical to the assembly of the discontinuous Galerkin terms in Section 5.4. These standard terms include:

- $\sum_{K\in\mathcal{T}_h^n} \|\varepsilon\Delta\theta_h^n - \mathbf{u}_h^n\cdot\nabla\theta_h^n - \delta^n\theta_h^n\|_K^2$,
- $\sum_{F\in\mathcal{F}_h^n} \|[\![\varepsilon\nabla\theta_h^n]\!]\|_F^2$,
- $\sum_{F\in\mathcal{F}_h^n} h_F\|\mathbf{u}^n\|_{F,\infty}^2 \|[\![\theta_h^n]\!]\|_F^2$.

What is not standard are calculations of the following quantities:

- Helmholtz-based terms, e.g., $\nabla\eta$ and ψ;
- quantities projected from one mesh to another, e.g., $\Pi^n\theta_h^{n-1}$;
- all quantities defined over the union mesh $\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n$;
- integration-in-time of quantities that are nonlinear or non-polynomial in time, e.g., ψ.

We begin by discussing the implementation of the approximate calculation of the Helmholtz decomposition term η. We note that this provides a calculation of the weight ψ through the formula $\psi \coloneqq \exp(-\alpha\eta)$. We then briefly present the implementation of the standard terms, before proceeding to discuss how we might implement the non-standard terms, which require us to fundamentally modify our algorithm and ultimately require a new *auxiliary-mesh* method. This uses a second mesh that lags behind the main triangulation, exactly following the same pattern of refinement and coarsening, but in a delayed manner.

5.6.1 Calculating the Helmholtz-decomposition term η

The Helmholtz decomposition (see Lemma 4.2) is the decomposition of the convection field $\mathbf{u}_h^n(\mathbf{x})$ into the sum of a curl-free term $\nabla\hat{\eta}^n$ and a divergence-free term $\mathbf{curl}\hat{\boldsymbol{\phi}}_h^n$. Since our estimator includes terms dependent on $\hat{\eta}^n$ and $\nabla\hat{\eta}^n$, we calculate an approximation to the field $\hat{\eta}^n$ within the simulation.

Since $\nabla \cdot \mathbf{curl}\hat{\boldsymbol{\phi}}_h^n = 0$, then $\hat{\eta}^n$ satisfies

$$\nabla \cdot \mathbf{u}_h^n = \Delta\hat{\eta}^n.$$

Thus we are able to compute the approximate field η_h^n by solving the FEM problem: find $\eta_h^n \in Y_h^n$ such that

$$(\nabla \cdot \mathbf{u}_h^n, v_h^n) = (\nabla\eta_h^n, \nabla v_h^n)$$

for all $v_h^n \in Z_h^n$, where

$$\begin{aligned} Y_h^n &\coloneqq X_{h,k}^n \cap C^0(\Omega) \cap \left\{v_h \in L^2(\Omega) : v_h|_\Gamma = 0\right\}, \\ Z_h^n &\coloneqq X_{h,k}^n \cap C^0(\Omega), \end{aligned}$$

with k the polynomial degree of the velocity field.

This can be calculated simply by forming the necessary stiffness matrix and load vector at each timestep, evaluating $\nabla \cdot \mathbf{u}_h^n$ at the necessary points.

We note that knowledge of η^n allows us to calculate ψ^n and $\mathcal{L}^n$.

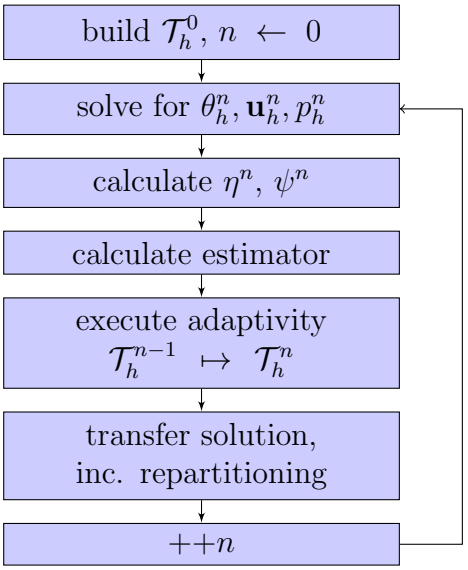


FIGURE 5.2: Mesh adaptivity workflow with Helmholtz term

In simple cases, for example numerical experiments in which we impose the velocity field explicitly, we may be able to also explicitly evaluate the value of η_h^n, and so we provide functionality to impose this if desired, noting that in this case we do not impose zero Dirichlet boundary conditions. We also note that, if we expect to have a very good incompressibility approximation, then we can forget about η in practice, setting it to 0.

5.6.2 A basic workflow with Helmholtz term

With the calculation of η and ψ as above, we are able to implement a basic workflow in the way illustrated in Figure 5.2. We use the common notation $++n$ to denote the *incrementation* of n by 1.

We note that this is simple to implement in the distributed case, since we need only interpolate the solution pair $(\theta_h^n, \mathbf{u}_h^n)$ onto the triangulation $\mathcal{T}_h^{n+1}$, and then transfer data between processes in the `repartition` stage, both of which are handled by the deal.II library.

5.6.3 The projection term

The first extension of the previous workflow is to include the term $\Pi^n \theta_h^{n-1}$, which appears in the term $\zeta^2_{S_2,n}$.

Note that it is not enough to simply interpolate θ_h^{n-1} to $\mathcal{T}_h^n$, since the equivalence of projection and interpolation will not hold anywhere that mesh coarsening takes place between $\mathcal{T}_h^{n-1}$ and $\mathcal{T}_h^n$. Instead, we require a full projection to the mesh $\mathcal{T}_h^n$.

To project a function from the dG space on one triangulation to another, both triangulations must exist simultaneously. This is an issue for the workflow illustrated in Figure 5.2, since there we replace $\mathcal{T}_h^{n-1}$ directly with $\mathcal{T}_h^n$ in the "execute adaptivity" stage. In order to project between the triangulations, we need to have both existing at once. We therefore consider the workflow in Figure 5.3.

This requires a copy of $\mathcal{T}_h^n$ to be made (denoted by the red, rounded box) and a copy of θ_h^n stored with it. After adapting the main mesh, we can use the `FEFieldFunction` functionality, which evaluates the FE function θ_h^{n-1} at any point in the domain. This allows us to interpolate θ_h^{n-1} at all points necessary to create the load vector for the projection operation. With this in place, we can complete the projection operation by using the mass matrix for the temperature field (which has already been assembled for the solution step), and inverting this against the load vector.

With this setup, we are able to add $\Pi^n \theta_h^{n-1}$ to the set of functions we are able to compute. We are also then able to use $\Pi^n \theta_h^{n-1}$ to calculate A^n via the formula (4.6.2), after also calculating the projection of functions f^n and $\delta^n \theta_h^n$ by inverting the mass matrix for the current mesh. This completes the calculation of terms $\zeta^2_{S_1,n}$ and $\zeta^2_{S_3,n}$ at each timestep t^n, $n \in \{0, \dots, N\}$.

However, implementing this functionality in the distributed case requires a different approach, for the reasons to be explained in the next section. This is due to the dependence of terms on the *union mesh* $\mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n$.

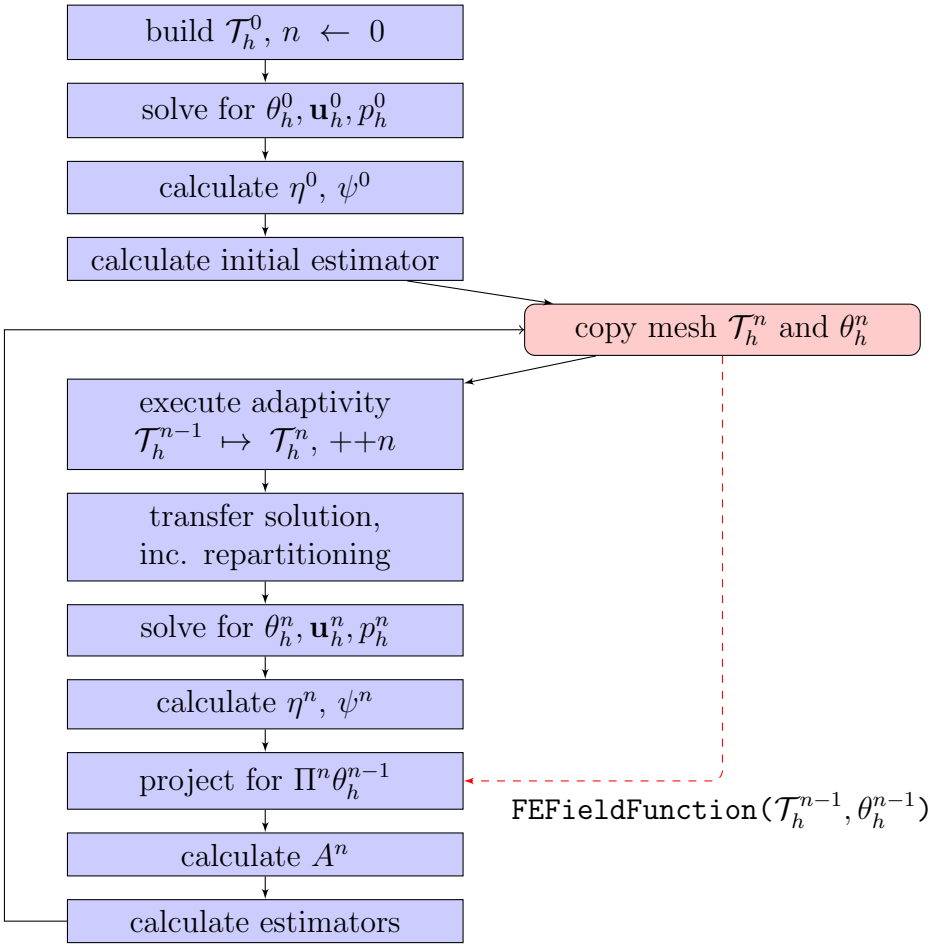


FIGURE 5.3: Mesh adaptivity workflow with $\Pi^n \theta_h^{n-1}$. Blue (rectangular) boxes denote work on the main triangulation. Red (rounded) box denotes work on copied triangulation. Red (dashed) arrow denotes selected information flows.

5.6.4 Union mesh in shared memory

Consider the estimator terms $\zeta_{S_2,n}^2$, $\zeta_{S_4,n}^2$, $\zeta_{T_1,n}^2$, and $\zeta_{T_2,n}^2$. For the computation of each of these terms, we are required to form the union mesh $\mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n$. We also wish to retain the ability to calculate the projection of functions from one mesh to the other. Happily, the deal.II library provides the `create_union_triangulation` function, which, given two meshes defined from the same coarse mesh, outputs the union triangulation. With this functionality, we are able to follow the algorithm outlined in Figure 5.4.

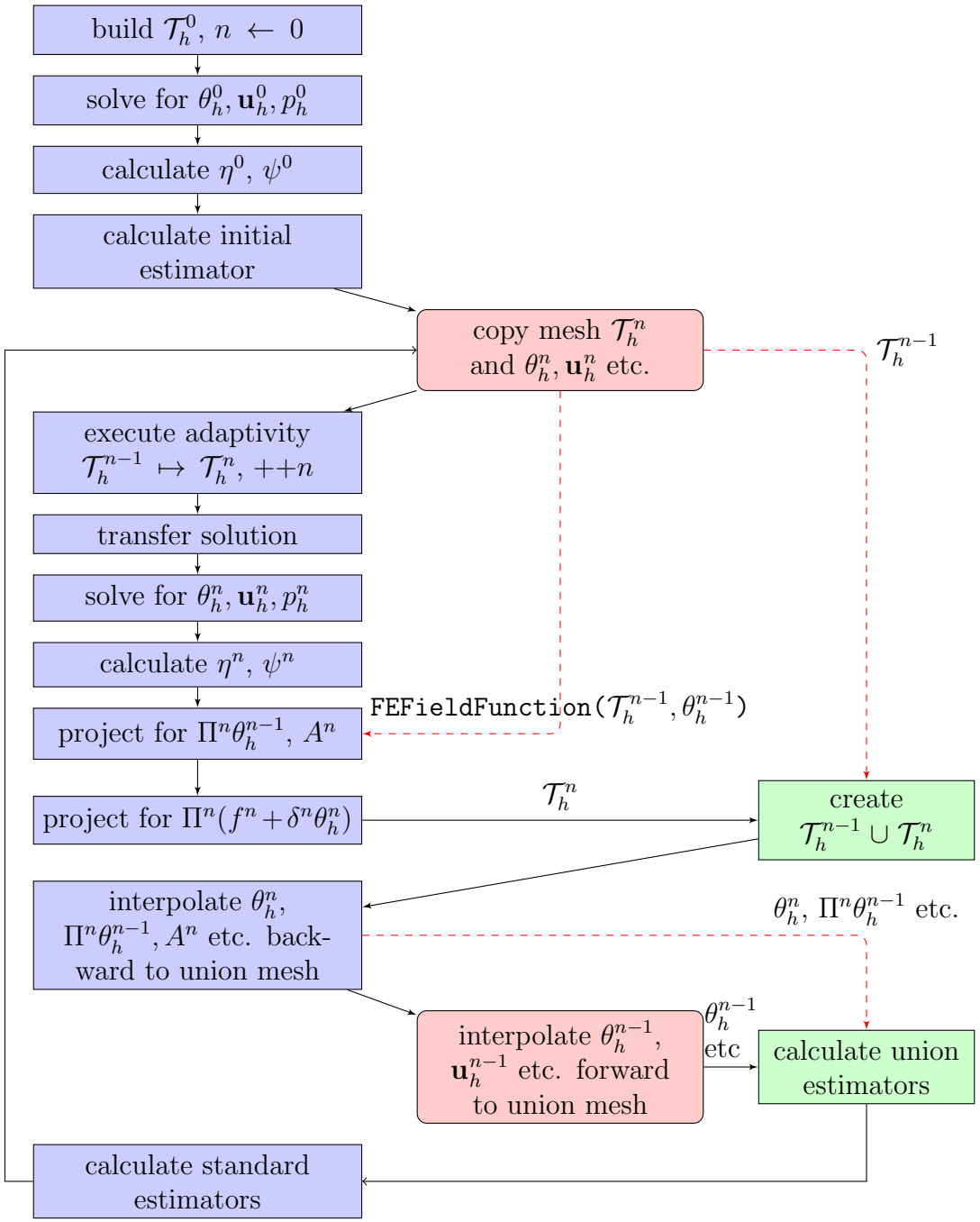


FIGURE 5.4: Mesh adaptivity workflow with terms defined on the union mesh. Blue (rectangular, left-hand side), red (rounded), and green (rectangular, right-hand side) boxes denote work on the main triangulation, copied triangulation, and union triangulation respectively. Red arrows (dashed) denote selected information flows.

Here, we create the union triangulation $\mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n$, and then in turn interpolate the necessary functions from $\mathcal{T}_h^{n-1}$ and $\mathcal{T}_h^n$ onto this union mesh, including the projection $\Pi^n \theta_h^{n-1}$. Since the union mesh is at least as fine as both $\mathcal{T}_h^{n-1}$ and $\mathcal{T}_h^n$, these interpolations are just the embedding operator.

Ultimately though, this approach is unsatisfactory for two reasons: firstly, the presence of three triangulation objects is potentially expensive to maintain. Secondly, the deal.II library lacks an analogue of the `create_union_triangulation` function for distributed triangulations. This is not unreasonable, since repartitioning happens after adaptivity to balance work across processors. Such a function would rely on delaying repartitioning until after the following timestep, or the ability to compare cells which, while occupying the same physical location in the computational domain, may exist on different processors at different timesteps. We instead take an alternative approach, which does not require this functionality. This reduces the number of triangulation objects from three to two, but does affect the resulting mesh sequences.

5.6.5 Union mesh in distributed memory

As noted previously, the union mesh $\mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n$ is exactly the mesh generated by applying only the refinement operations required to move from $\mathcal{T}_h^{n-1}$ to $\mathcal{T}_h^n$. Based on this identity, we are able to utilise the following strategy.

Instead of making a copy of the triangulation at each timestep, we keep an *auxiliary triangulation* $\mathcal{S}_h^n$ throughout the simulation which follows the main triangulation. By saving and re-using the refinement and coarsening flags used on the main triangulation, we can ensure that the auxiliary triangulation follows exactly the same pattern of refinement and coarsening as the main, but at a delayed time in the simulation process.

This allows the workflow detailed in Figure 5.5.

Here, the auxiliary triangulation $\mathcal{S}_h^{n-1}$ is held in the unadapted state while the main triangulation is adapted. This then allows a `FEFieldFunction` to facilitate projection onto $\mathcal{T}_h^n$ as before. Once this is completed, we apply only the refinement process

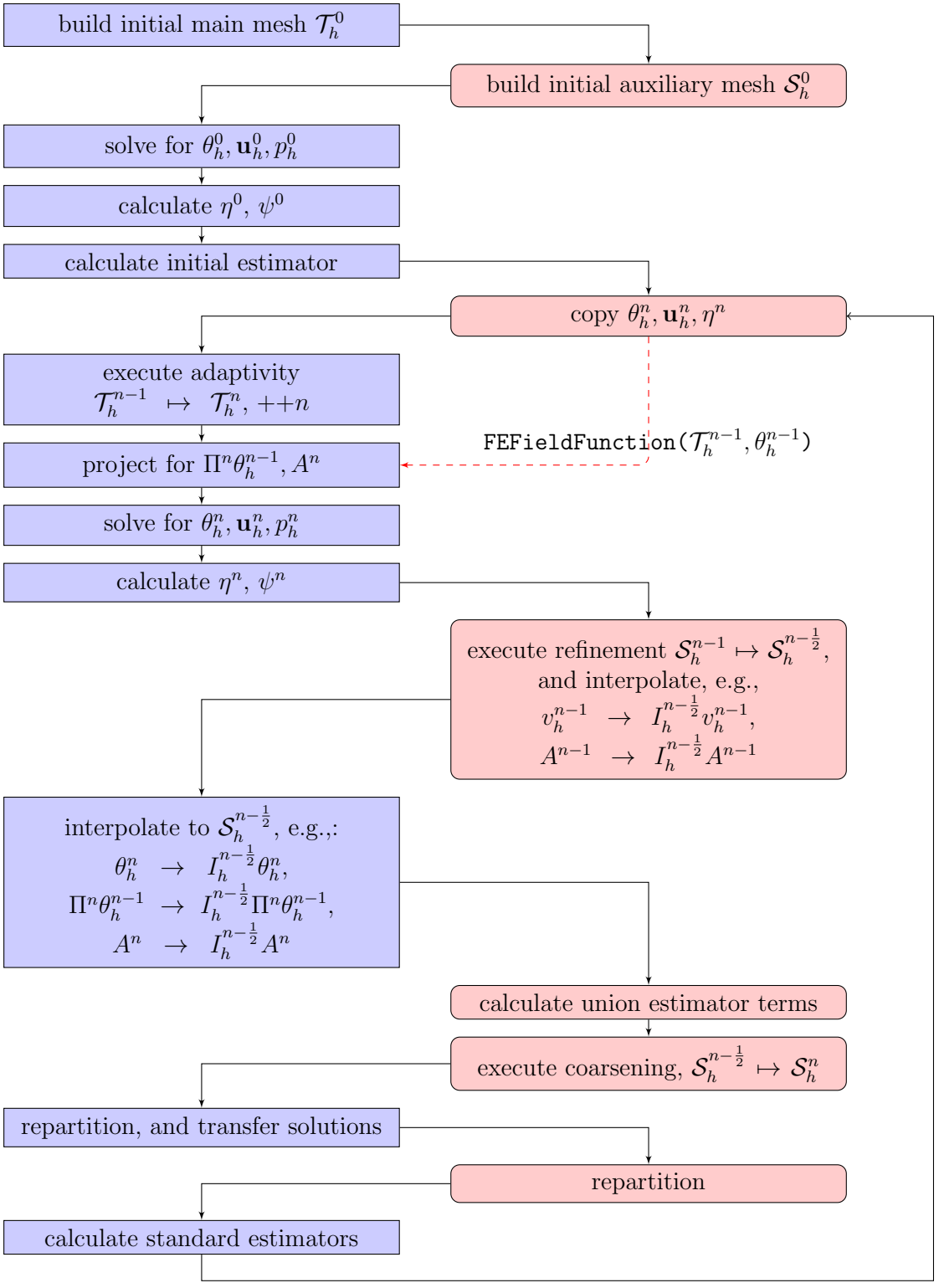


FIGURE 5.5: Simplified mesh adaptivity workflow with terms defined on the union mesh in a distributed simulation. Blue (rectangular) and red (rounded) boxes denote work on the main triangulation and auxiliary triangulation respectively. Red (dashed) arrow denotes selected information flows.

to $\mathcal{S}_h^{n-1}$, yielding $\mathcal{S}_h^{n-\frac{1}{2}}$. Note that this may not be exactly the union triangulation, as in principle a cell may be refined and then its children be coarsened during the same step. However $\mathcal{S}_h^{n-\frac{1}{2}}$ is at least as refined as the union mesh. Interpolation to $\mathcal{S}_h^{n-\frac{1}{2}}$ of all the FE functions from $\mathcal{S}_h^{n-1}$ amounts to the identity operator. We then use the `interpolate_to_different_mesh` function to interpolate the necessary FE functions defined on $\mathcal{T}_h^n$ to $\mathcal{S}_h^{n-\frac{1}{2}}$. Note that, through this approach, only two meshes are stored at any one time. We then have all the necessary information on $\mathcal{S}_h^{n-\frac{1}{2}}$ to calculate the full estimator, so long as we are prepared to pay a small price by interpolating some non-linear functions, such as ψ, by linear-in-time functions. Finally, we apply the coarsening operations to $\mathcal{S}_h^{n-\frac{1}{2}}$, ensuring $\mathcal{S}_h^n = \mathcal{T}_h^n$, and we repartition both meshes, to ensure that work is spread evenly over the set of processors.

In the next subsection, we discuss limitations on our ability to exactly compute some of these terms, but we remark here briefly on the mesh generated by this process.

Crucially, the error indicators are used by the adaptivity processes in deal.II only as 'hints', that is, they form the basis of the adaptivity process but do not fully define it. Since many applications depend on 'well-behaved' meshes, the library applies smoothing algorithms to the meshes generated under adaptivity. This ensures a number of desirable properties are inherited by new meshes. For example, limits are placed upon the difference in size between edge-adjacent and vertex-adjacent cells, to ensure we have only one hanging node per edge, and that all cells sharing a vertex differ in size by no more than a single level of refinement.

This process of smoothing is applied once per adaptivity stage (refinement, coarsening, or both refinement and coarsening). This means that, in general, the process of adapting by pure refinement, followed by adapting by pure coarsening, is not identical to a single combined stage of refinement and coarsening. To ensure identical main and auxiliary meshes, we simply adapt the main triangulation in two stages, separate refinement and coarsening steps, which ensures the auxiliary mesh will exactly replicate it. While none of the intermediate meshes will be badly-behaved under this regime, it is possible (but not seen in the author's experience) that such a method could reduce the mesh quality compared to the approach of applying smoothing to the combined operations of refinement and coarsening. We note that any effect in this regard can be checked by monitoring $\zeta_{S_2,n}$.

The resulting full algorithm for the calculation of all estimator terms, including the treatment of the adaptivity flags, is shown in Figure 5.6.

5.6.6 Estimator implementation approximations

There are a number of limitations upon our ability to accurately calculate the value of the bound on the error. These fall under two categories: the calculation of maxima over large patches, and integration over time intervals.

The first of these, calculation of maxima over large patches, is manifested in our calculation of $\overline{\psi}_{\tilde{\omega}_F}$, $\|\mathbf{u}^n - \alpha^n \varepsilon \nabla \eta^n\|^2_{\tilde{\omega}_F,\infty}$, and $\|\mathcal{L}^n\|_{\psi,\tilde{\omega}_F,\infty}$ in $\zeta_{S_1,n}$; $\overline{\psi}_{\tilde{\omega}_F}$ in $\zeta_{S_3,n}$; and $\|\mathcal{L}^{-\frac{1}{2}}\|_{\psi,\tilde{\omega}_F,\infty}$ and $\overline{\psi}_{\tilde{\omega}_F}$ in $\zeta_{S_4,n}$.

Each of these requires the calculation of a maximum over all values within $\tilde{\omega}_F$, namely the set of all cells that share at least a vertex with the edge F. However, this calculation is not easily amenable to the setup we have for calculating the estimator. Since the estimator works by iterating over all cells, and all faces of each cell, we in general only have access to the cells on either side of the given edge F. In general, we have no guarantee that all cells in $\tilde{\omega}_F$ are edge-neighbours of the cells on either side of F. Thus, we make an approximation by instead calculating the maximum over the edge patch $\omega_F \subset \tilde{\omega}_F$. This allows us to compute with access to only the two cells sharing F as an edge.

The second approximation is that of integration in time. For example, we have that

$$\begin{aligned}\int_{t^{n-1}}^{t^n} \zeta^2_{S_2,n} \,\mathrm{d}s &= \int_{t^{n-1}}^{t^n} \sum_{K \in \mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n} \rho_K^2 \left\| (I - \Pi^n) \left(f^n + \delta^n \theta_h^n + \frac{\theta_h^{n-1}}{\tau^n} \right) \right\|_K^2 \,\mathrm{d}s \\ &= \sum_{K \in \mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n} \left\| (I - \Pi^n) \left(f^n + \delta^n \theta_h^n + \frac{\theta_h^{n-1}}{\tau^n} \right) \right\|_K^2 \int_{t^{n-1}}^{t^n} \rho_K^2 \,\mathrm{d}s.\end{aligned}$$

The cell weight ρ_K^2 is varying in time, since

$$\rho_K = \frac{1}{\sqrt{\underline{\psi}_K}} \min \left\{ \frac{\overline{\psi}_K}{\sqrt{\underline{\mathcal{L}}_K}}, h_K \max \left\{ \frac{\overline{\nabla \psi}_K}{\sqrt{\underline{\mathcal{L}}_K}}, \frac{\overline{\psi}_K}{\sqrt{\varepsilon}} \right\} \right\},$$

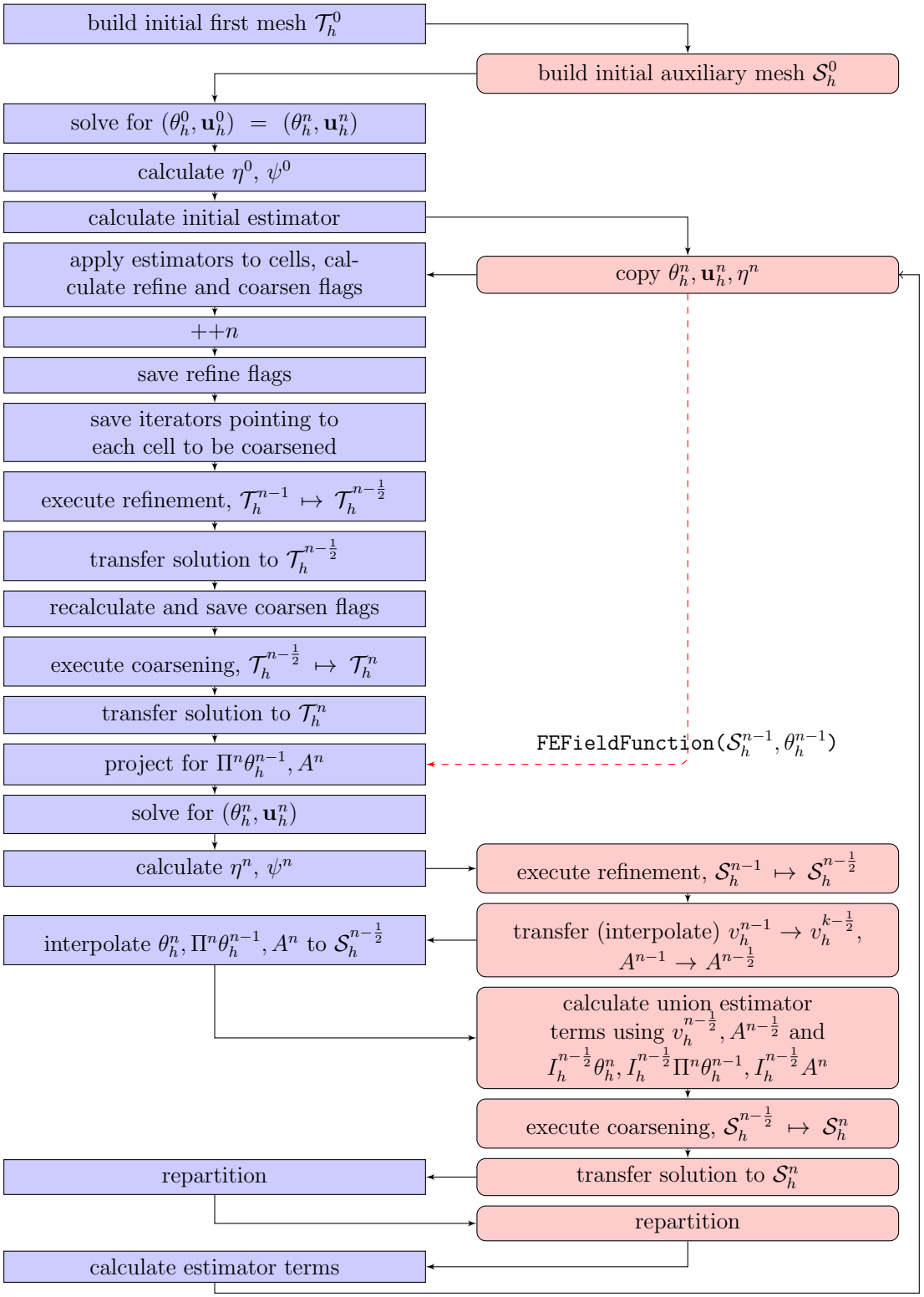


FIGURE 5.6: Mesh adaptivity workflow with space-estimator terms defined on the union mesh in a distributed simulation. Blue (rectangular) and red (rounded) boxes denote work on the main triangulation and auxiliary triangulation respectively. Red (dashed) arrow denotes selected information flows.

and, due to the presence of the exponential function in the weighting function ψ, is in general non-polynomial. Its exact integration is therefore a challenge, but since it is smoothly varying we do not expect its approximate integration to be such. We take different approaches to computing this quantity in the terms $\zeta^2_{S_1,n}$ and $\zeta^2_{S_2,n}$ for simplicity of implementation. Since $\zeta^2_{S_1,n}$ is defined on a single mesh, we evaluate ρ^2_K only at the end of the time interval. In contrast, for the term $\zeta^2_{S_2,n}$, the implementation has access to the union mesh, and the values of the necessary quantities at both ends of each time interval. As such, in $\zeta^2_{S_2,n}$ we can take the approximation that

$$\int_{t^{n-1}}^{t^n} \rho^2_K \,\mathrm{d}s \approx \tau^n \max\left\{\rho^2_K|_{t^{n-1}}, \rho^2_K|_{t^n}\right\},$$

with little extra effort. We can treat the coefficient $\|\mathcal{L}^{-\frac{1}{2}}\|^2_{\psi,\tilde{\omega}_F,\infty}$ in $\zeta^2_{S_4,n}$ in an identical manner, approximating

$$\int_{t^{n-1}}^{t^n} \|\mathcal{L}^{-\frac{1}{2}}\|^2_{\psi,\tilde{\omega}_F,\infty} \,\mathrm{d}s \approx \tau^n \max\left\{\|\mathcal{L}^{-\frac{1}{2}}\|^2_{\psi,\tilde{\omega}_F,\infty}|_{t^{n-1}}, \|\mathcal{L}^{-\frac{1}{2}}\|^2_{\psi,\tilde{\omega}_F,\infty}|_{t^n}\right\}.$$

Finally, we have the integrals

$$\begin{aligned}\int_{t^{n-1}}^{t^n} \zeta^2_{T_1,n} \,\mathrm{d}s \\ = \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \varepsilon^{-1} \int_{t^{n-1}}^{t^n} \left\|\ell_n\left(\mathbf{u}^n - \mathbf{u}\right)\theta_h^n + \ell_{n-1}\left(\mathbf{u}^{n-1} - \mathbf{u}\right)\theta_h^{n-1}\right\|^2_{\psi,K} \,\mathrm{d}s,\end{aligned}$$

and

$$\begin{aligned}\int_{t^{n-1}}^{t^n} \zeta^2_{T_2,n} \,\mathrm{d}s \\ = \sum_{K\in\mathcal{T}_h^{n-1}\cup\mathcal{T}_h^n} \int_{t^{n-1}}^{t^n} \Big\|\min\left\{\mathcal{L}^{-\frac{1}{2}}, \varepsilon^{-\frac{1}{2}}\right\}\big(f - f^n + \delta\theta_h - \delta^n\theta_h^n \\ + \ell_{n-1}\left(A^n - A^{n-1}\right) + \ell_n\beta^n\theta_h^n + \ell_{n-1}\beta^{n-1}\theta_h^{n-1}\big)\Big\|^2_{\psi,K} \,\mathrm{d}s.\end{aligned}$$

Replacing $\mathbf{u}$ by $\ell_{n-1}\mathbf{u}^{n-1} + \ell_n\mathbf{u}^n$ and making the approximation $\psi \approx \ell_{n-1}\psi^{n-1} + \ell_n\psi^n$, we have that, on each cell, the first integral is approximately

$$\varepsilon^{-1} \int_{t^{n-1}}^{t^n} \left\| \ell_n \left(\mathbf{u}^n - \mathbf{u} \right) \theta_h^n + \ell_{n-1} \left(\mathbf{u}^{n-1} - \mathbf{u} \right) \theta_h^{n-1} \right\|_{\psi,K}^2 \, \mathrm{d}s$$
$$\approx \sum_{q \in Q} (\ell_{n-1}\psi^{n-1} + \ell_n \psi^n)[q] * \ell_n^2 \ell_{n-1}^2 \left(\mathbf{u}^n - \mathbf{u}^{n-1} \right)^2 \left(\theta_h^n - \theta_h^{n-1} \right)^2 [q] * \texttt{JxW}[q],$$

a fifth-order polynomial in time. Here, `JxW[q]` is the value of the Jacobian times the quadrature rule weight, evaluated at the quadrature point q. This leads us to use the 3rd-order Gaussian quadrature scheme to exactly compute the approximate integral in time.

The second time integral is more complicated. By approximating $\mathcal{L} \approx \ell_{n-1}\mathcal{L}^{n-1} + \ell_n \mathcal{L}^n$ and $\psi \approx \ell_{n-1}\psi^{n-1} + \ell_n \psi^n$, and evaluating all other terms exactly as their linear Lagrangian interpolant, we come to a similar equation on each cell, which this time shows the integral to be a rational function, with a quartic polynomial in the numerator and a linear polynomial in the denominator. We note that the approximation of ψ by its linear interpolant is second-order, and so will not affect the order of convergence of this term. For ease of implementation, we apply the same 3rd-order Gaussian quadrature scheme to approximate this integral.

5.7 Error indicator choice

While all the above terms form the error estimate, it is a more flexible approach to use only a subset of these as the error indicator. Specifically, we use only the term $\zeta_{S_1,n}^2$ when indicating refinement and coarsening areas.

Since we refine and coarsen in a cellwise manner (that is, we refine and coarsen based on the indicator associated with a cell or patch of cells) rather than an edgewise manner (where weights are associated to each edge and refinement or coarsening occurs in the cells neighbouring such), we map the edge weights of the error indicator to both neighbouring cells, so that the per-cell estimator is

$$\zeta_{n,K}^2 \coloneqq \rho_K^2 \left\| A^n + \varepsilon \Delta \theta_h^n - \mathbf{u}^n \cdot \nabla \theta_h^n - \delta^n \theta_h^n \right\|_K^2$$
$$+ \sum_{F \in \partial K \setminus \Gamma} \rho_{\omega_F} \left\| [\![\varepsilon \nabla \theta_h^n]\!] \right\|_F^2$$

$$+\sum_{F\in\partial K}\left(\frac{\sigma\varepsilon}{h_F}\left(\overline{\psi}_{\omega_F}+\varrho_{\omega_F}\sigma\varepsilon+\frac{\overline{\psi}_F\alpha^2\varepsilon\overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}}\right)+\rho_{\omega_F}\|\mathbf{u}\|_{F,\infty}^2\right.$$
$$\left.+h_F\|\mathcal{L}\|_{\psi,\omega_F,\infty}+\frac{\overline{\psi}_{\omega_F}h_F}{\varepsilon}\|\mathbf{u}-\alpha\varepsilon\nabla\eta\|_{\omega_F,\infty}^2\right)\|[\![\theta_h^n]\!]\|_F^2\,.$$

As an alternative, the code also provides for use of the Kelly error estimator [64] as an error indicator, for the purpose of comparison. While this is a widely-used error indicator among h-refinement codes, it is derived by considering a much simpler problem, without many features of the Boussinesq system (2.5.1). Thus we may expect the two error indicators to exhibit different characteristics and lead to differing adaptivity patterns.

Finally, once we have calculated our error indicators, we have two mesh adaptivity strategies available to us here and in Chapter 6. We begin by ordering the cells by their error indicator value. Strategy 1, which we call adaptivity by fraction of error, uses a pre-defined percentage value (say, 30%) and then marks cells, from highest indicator to lowest, until it has marked enough to account for 30% of the sum of all indicator values. We then mark the lowest-indicator cells for coarsening, until, say, 5% of the sum of indicator values is accounted for. Adaptivity then takes place using these markers, but subject to refinement level and mesh smoothing limits. This strategy offers the ability to adjust the number of cells in the mesh to ensure a certain amount of indicated error is refined per adaptivity step, but is difficult to use in the case where the total number of cells is required to be limited in some way.

Strategy 2, which is adaptivity by fraction of cells, marks cells in order of indicator value, but takes 30%, say, of the total number of cells, rather than referring to the sum of all indicator values. This has the benefit of offering greater control over the number of cells in the simulation, but offers less in the way of user-defined control of error.

5.8 Examining the error bound behaviour

In this section, we use the code developed above to examine the behaviour of the full error estimate under some simple conditions. In the following examples, the initial

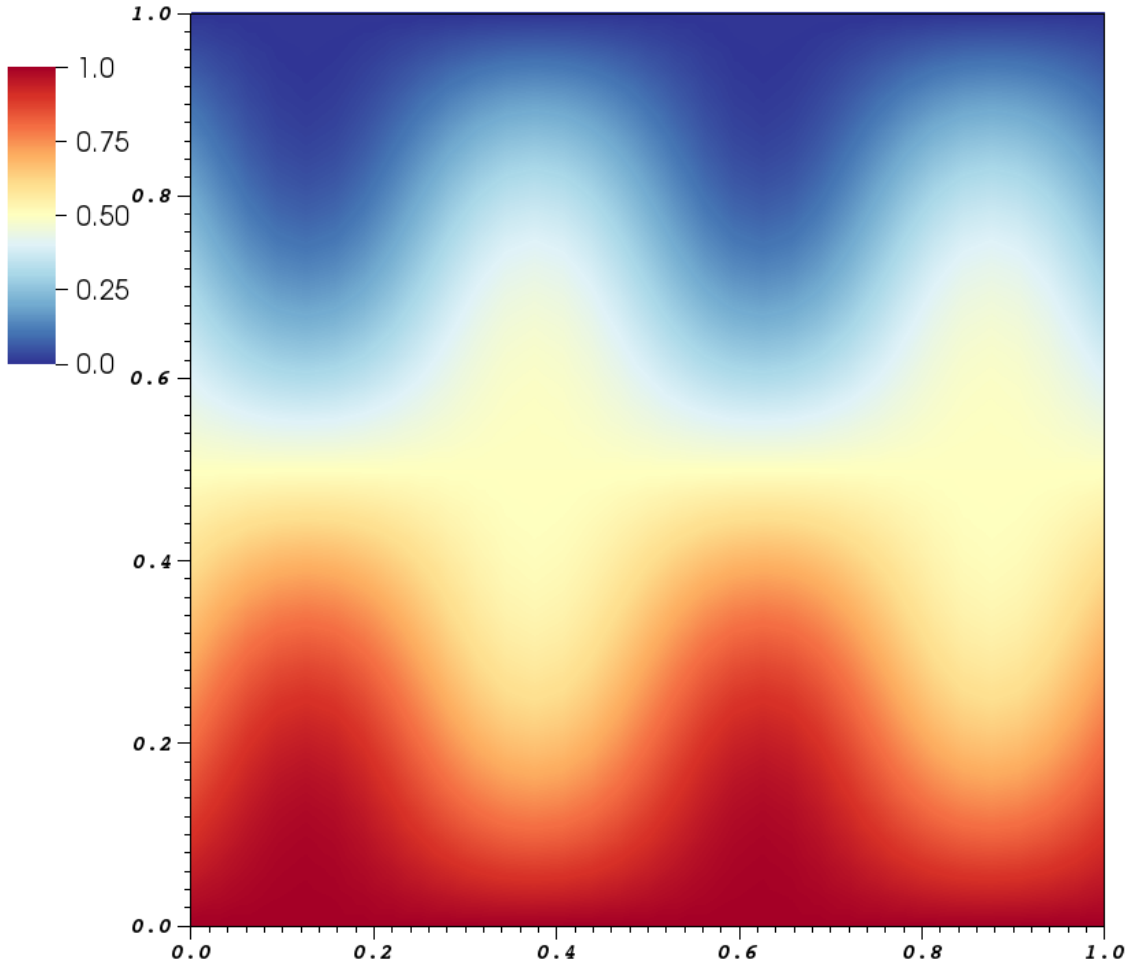


FIGURE 5.7: The initial temperature field in examples 1a, 1b, 2, 3a, 3b.

temperature field is defined by

$$1 - (1 - y + 0.15\sin(4\pi x)\sin(2\pi y)),$$

on a box domain $\Omega = [0,1]^2$, with Dirichlet boundary conditions enforced on all boundaries, with values compatible with the initial temperature field. The diffusion is constant, ε =1e−6, and a uniform mesh is used. This setup is illustrated in Figure 5.7. On each example, we impose a fixed velocity throughout the domain.

We use the notation

$$\zeta_{S,k}^2 := \sum_{n=1}^{k} \int_{t^{n-1}}^{t^n} \left(\zeta_{S_1,n}^2 + \zeta_{S_1,n-1}^2 + \zeta_{S_2,n}^2 + \zeta_{S_4,n}^2\right) \, \mathrm{d}s + \max_{0\le n\le N} \zeta_{S_3,n}^2,$$

and

$$\zeta_{T,k}^2 \coloneqq \sum_{n=1}^{k} \int_{t^{n-1}}^{t^n} \zeta_{T_1,n}^2 + \zeta_{T_2,n}^2 \, \mathrm{d}s,$$

to refer to the full spatial estimate, and time estimate, respectively.

We also use the notation ζ_k^2 to refer to the full error estimate bound as in the right hand side of (4.6.6), apart from the initial discretisation error $\|e(0)\|_\psi^2$.

$$\zeta_k^2 \coloneqq \exp\left(\int_0^{t^k} \max_{\Omega} \frac{\delta^2}{\mathcal{L}}(s) \, \mathrm{d}s\right) \left(\sum_{n=1}^{k} \int_{t^{n-1}}^{t^n} \left(\zeta_{S_1,n}^2 + \zeta_{S_1,n-1}^2 + \zeta_{S_2,n}^2 + \zeta_{S_4,n}^2\right) \, \mathrm{d}s \right.$$
$$\left. + \sum_{n=1}^{k} \int_{t^{n-1}}^{t^n} \left(\zeta_{T_1,n}^2 + \zeta_{T_2,n}^2\right) \, \mathrm{d}s + \max_{0 \le n \le k} \zeta_{S_3,n}^2 \right).$$

In the following, we repeatedly make use of the shorthand for z-independent vector fields as defined in Remark 4.1), that is, we may denote a vector field of the form $\boldsymbol{\Psi} \coloneqq (0, 0, g(x,y))^\intercal$, where $g(x,y)$ is constant in the z-direction, by $g(x,y)$.

5.8.1 Case 1a

We impose the divergence-free flow $\mathbf{u} = \mathbf{curl}\boldsymbol{\phi}$, where $\boldsymbol{\phi} = \frac{x^2+y^2}{2}$. This means $\mathbf{u} = \begin{pmatrix} y \\ -x \end{pmatrix}$, and $\eta = 0$. Thus the weight ψ is equal to 1, and we recover an un-weighted dG norm. Under these circumstances, we have $\mathcal{L} = \delta$, and so we may choose $\delta = 0$ to remove the exponential term in the estimator, but have only an H^1 seminorm bound. Figure 5.8 shows the behaviour of the dominant term $\zeta_{S_1,k}^2$ in the estimator, along with $\zeta_{S,k}^2$, $\zeta_{T,k}^2$ and ζ_k^2.

5.8.2 Case 1b

Identically to Case 1a, $\mathbf{u} = \mathbf{curl}\boldsymbol{\phi}$, where $\boldsymbol{\phi} = \frac{x^2+y^2}{2}$, with $\mathbf{u} = \begin{pmatrix} y \\ -x \end{pmatrix}$, and $\eta = 0$. In contrast, we choose δ as a constant, e.g., $\delta = 0.1$, and we have $\mathcal{L} = \delta$. Thus the error estimate has an exponential term of $e^{0.1T}$, but includes an L^2 term of $0.1\,\|e\|_K^2$. Comparing Figures 5.8 and 5.9, we observe the following. In Case 1a, the lack of an

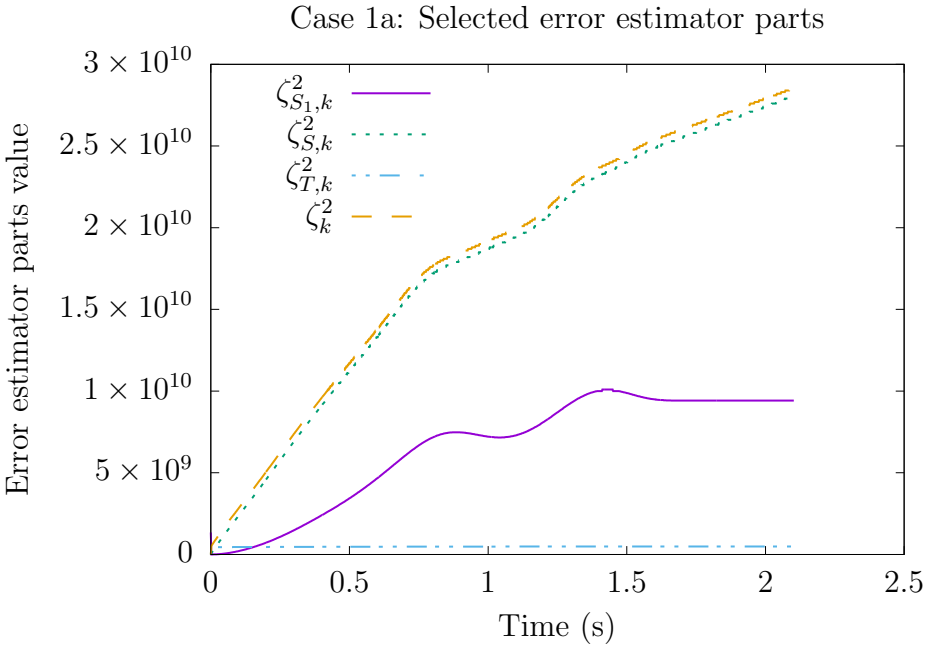


FIGURE 5.8: Estimator terms in Case 1a.

L^2 term forces the estimator to rely on inequalities related to the diffusion ε. This leads to an instant factor of 1e6 in several estimator terms, and so this estimator has a large absolute value, but exhibits only linear growth after $t = 1.5$. On the other hand, in Case 1b we bound the full dG norm including an L^2 term. We are thus able to rely on inequalities involving $\mathcal{L} = 0.1$, leading to a much smaller absolute value for the estimator at small times, but the exponential nature of the error bound begins to show at later times (since the exponent is only $0.1t$, this example exhibits very slow exponential growth, but will eventually overwhelm the estimate of Case 1a).

5.8.3 Case 2

In this example, we take

$$\mathbf{u} = \begin{pmatrix} e^x \sin y + y \\ e^x \cos y - x \end{pmatrix} = \nabla(e^x \sin y) + \mathbf{curl}\frac{x^2 + y^2}{2}.$$

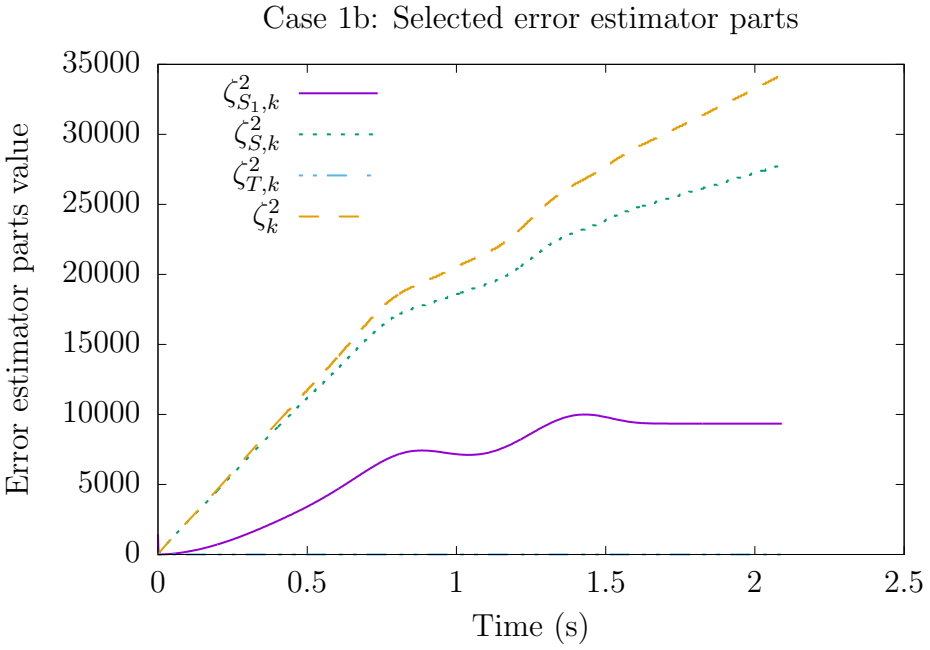


FIGURE 5.9: Estimator terms in Case 1b.

This flow field can no longer be characterised as $\mathbf{u} = \mathbf{curl}\boldsymbol{\phi}$, but it is still divergence-free, and η is harmonic but not zero. Since $\nabla \cdot \mathbf{u} = 0$, then

$$\mathcal{L} = \delta + \frac{1}{2}\alpha\left(\mathbf{u}\cdot\nabla\eta - \alpha\varepsilon|\nabla\eta|^2\right) = \delta + \frac{1}{2}\alpha e^x\left((1-\alpha\varepsilon)\,e^x + y\sin y - x\cos y\right).$$

Since we are considering the box $[0,1]^2$, then $\mathcal{L} > \delta$, and so we can choose $\delta = 0$. Thus we once again have no exponential term, but we do also have an L^2 term in the norm. See Figure 5.10.

5.8.4 Case 3

To consider a case in which the existing literature is not well equipped, we impose the flow

$$\begin{pmatrix} x \\ y \end{pmatrix} = \nabla\left(\frac{x^2+y^2}{2}\right),$$

which has positive divergence ($\nabla\cdot\mathbf{u} = 2$) on $[0,1]^2$. In this example then,

$$\frac{1}{2}\left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right) = \frac{1}{2}\left(1-\alpha\varepsilon\right)\left(\alpha\left(x^2+y^2\right) - 2\right).$$

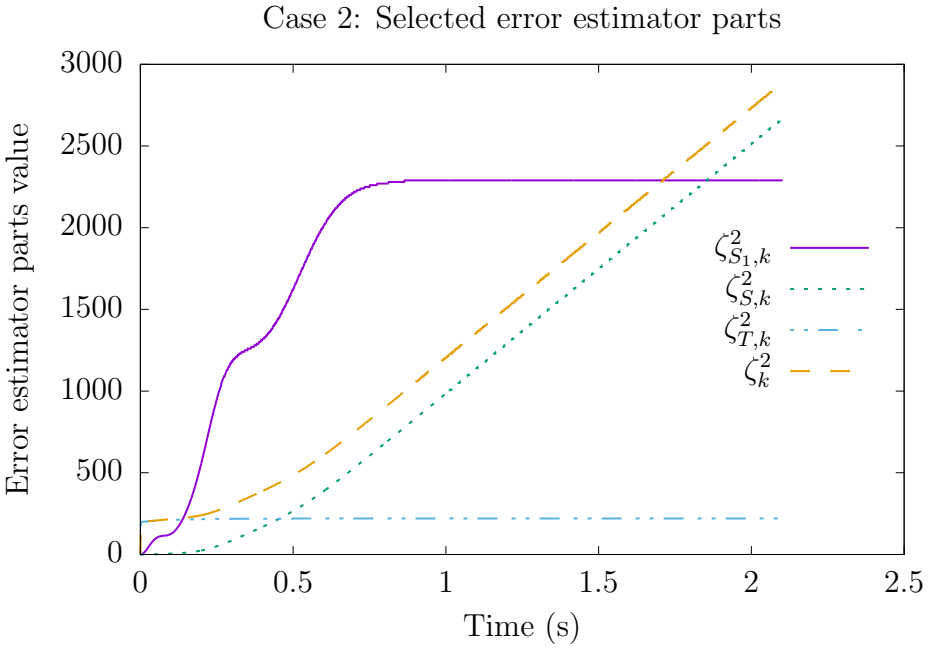


FIGURE 5.10: Estimator terms in Case 2.

Thus we must add an artificial reaction term $\delta = 2(1-\alpha\varepsilon)(2-\alpha(x^2+y^2))$ to satisfy (4.2.9).

We consider the two approaches offered by the error estimate. We first take the simple choice of $\alpha = 1$. Then the minimal artificial reaction we can impose is $\delta = 2(1-\varepsilon)(2-x^2-y^2)$. This leads to an exponential term

$$\exp\left(\int_0^T \max_\Omega \frac{\delta^2}{\mathcal{L}}\,\mathrm{d}t\right) = \exp\left(\frac{8}{3}(1-\varepsilon)\,t\right),$$

in the error estimator. See Figure 5.11 for the result. The full error bound (which is shown against a log scale) shows a clear exponential behaviour, eventually becoming too large for double precision arithmetic to represent.

We remark that, if we had not used the exponential fitting technique, then we would have been required to add enough reaction δ to handle $\frac{1}{2}\nabla\cdot\mathbf{u}$, i.e., we would have required $\delta = 4$, leading to an exponential term $\exp\left(\frac{8}{3}t\right)$, and so the exponential fitting here has enabled us to slightly reduce the factor in the exponential. It is easy to see that there will be examples where this difference is more substantial, particularly in the case where $\mathbf{u} \neq \nabla\eta$ and $\nabla\cdot\mathbf{u} \neq 0$.

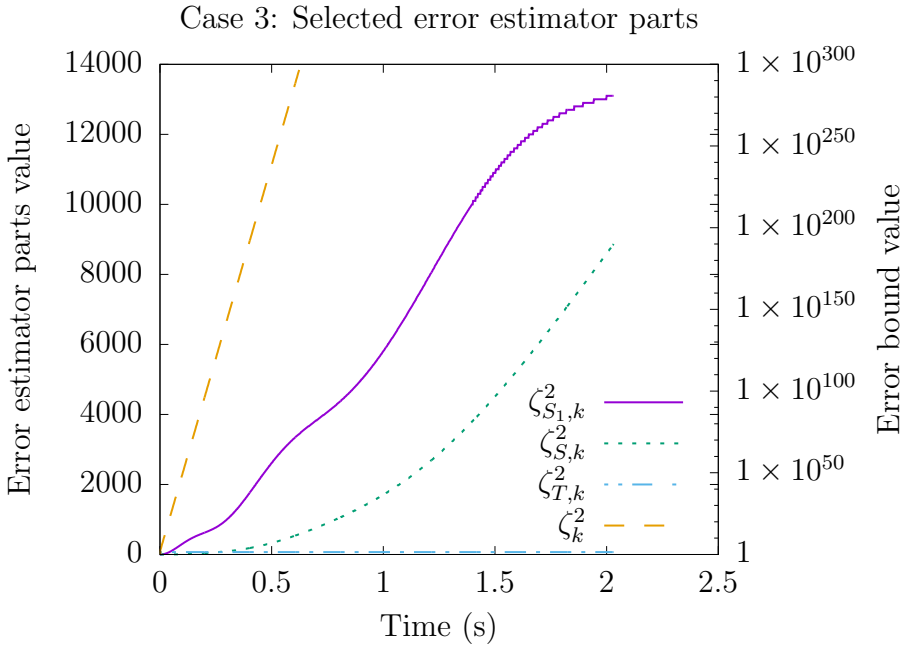


FIGURE 5.11: Estimator term in Case 3, with ζ_k^2 plotted against a log scale.

Alternatively, we could use the other freedom afforded us by the estimator, and alter our value of α to improve this behaviour. However, in the author's experience this is not usually useful in the case of a small diffusion coefficient – to have a measurable effect on the exponential term requires a very large $\alpha \sim \varepsilon^{-1}$, but since the weight ψ depends upon α, we experience terms of the type $\alpha \exp(-\alpha)$, which for large α quickly stretches the ability of double precision arithmetic routines.

5.8.5 Case 4

Finally, we look at the case of a positive-divergence field with a non-zero curl part. Taking

$$\mathbf{u} = \begin{pmatrix} x \\ x^2 + y^2 \end{pmatrix} = \nabla\left(\frac{x^2}{2} + x^2 y\right) + \mathbf{curl}\left(-xy^2\right),$$

and choosing α equal to 1, we have that

$$\begin{aligned} &\frac{1}{2}\left(\alpha\nabla\eta - \nabla\right)\cdot\left(\mathbf{u} - \alpha\varepsilon\nabla\eta\right) \\ &\qquad = \frac{1}{2}\left(\left(1-\varepsilon\right)\left(x^4 + x^2 - 1 - 2y\right) + \left(2 - 4\varepsilon\right)x^2 y + \left(1 - 4\varepsilon\right)x^2 y^2\right), \end{aligned}$$

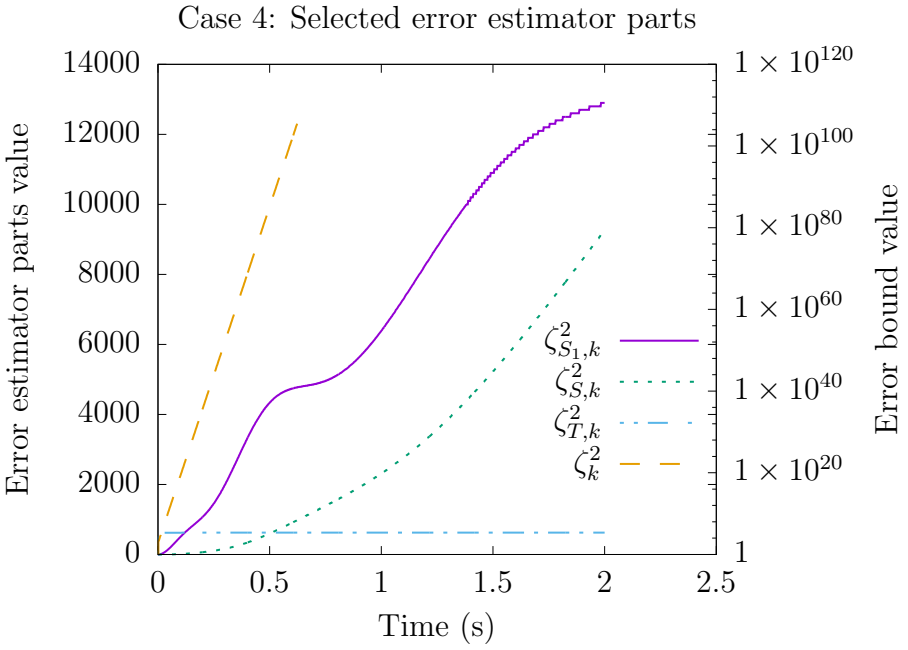


FIGURE 5.12: Estimator terms in Case 4, with ζ_k^2 plotted against a log scale.

and so we must add reaction

$$-2\left((1-\varepsilon)\left(x^4+x^2-1-2y\right)-(2-4\varepsilon)\,x^2y-(1-4\varepsilon)\,x^2y^2\right).$$

This leads to an exponential term of $\exp\left(8\,(1-\varepsilon)\,t\right)$, which is demonstrated in Figure 5.12, where ζ_k^2 is plotted against a log scale.

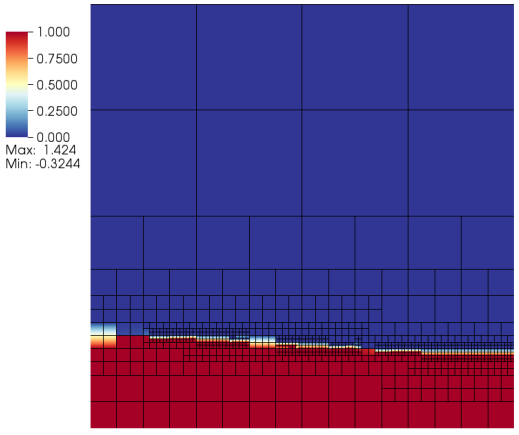
5.9 Comparison between Kelly and new indicators

In this section, we run two identical simulations, comparing the behaviour of the newly-derived adaptivity indicator to that of the Kelly adaptivity indicator. We run a simulation based upon the thermochemical convection benchmark of [63]. In this case, we initialise the system with a base of warm material below a colder material, with a small perturbation imposed on the interface to reliably initiate a convective flow. The temperature boundary conditions in this example are fixed Dirichlet, compatible with the initial field. This means that there is a discontinuity in the

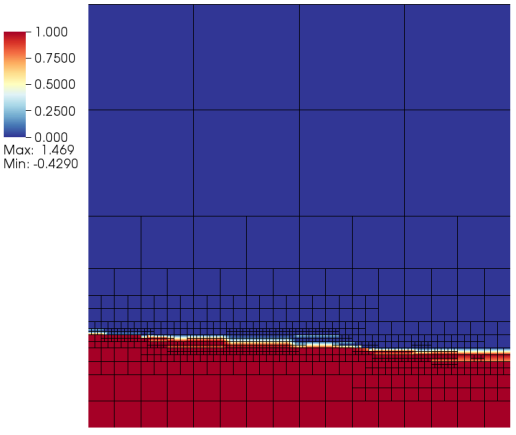
boundary conditions where the temperature jumps from 0 to 1 on the left and right boundaries.

Figure 5.13 shows the results of the two simulations, with both discretised by the dG method. Adaptivity strategy 2, (by fraction of cells) is employed. The two estimators behave very similarly in this example, with near-identical refinement patterns, and near-identical numbers of DoFs at each timestep (see Figure 5.14).

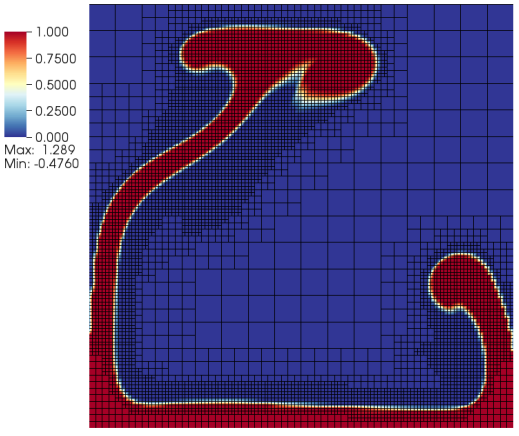
In order to explore this behaviour, we illustrate a number of the indicator terms in Figure 5.15. Subfigure 5.15a and 5.15b demonstrate the reason for the close matching of adaptivity results: the Kelly indicator and derived indicator agree largely upon the ordering of the cells in terms of per-cell indicator. Subfigure 5.15d confirms that, as we expect, the edge value jump terms dominate in most areas. Finally, subfigures 5.15e and 5.15f show a key difference between the two indicators. By plotting on a linear scale, we can identify that the derived estimator is strongly



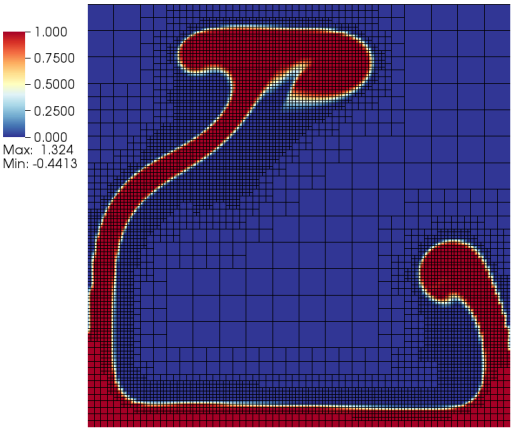
(a) Kelly, $t = 0$



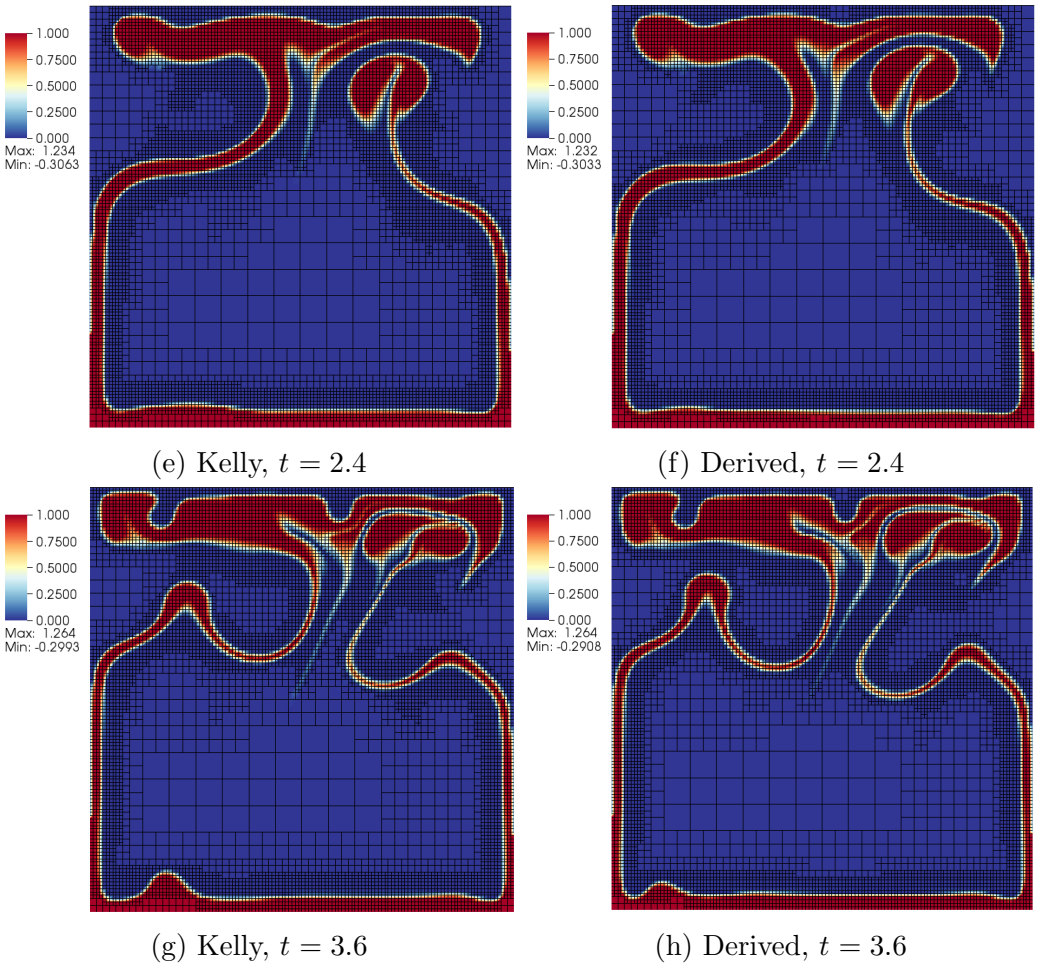
(b) Derived, $t = 0$



(c) Kelly, $t = 1.2$



(d) Derived, $t = 1.2$



(e) Kelly, $t = 2.4$ (f) Derived, $t = 2.4$

(g) Kelly, $t = 3.6$ (h) Derived, $t = 3.6$

FIGURE 5.13: A comparative display of two dG simulations through time, plotting the temperature spatial distribution and mesh in the two cases of Kelly-driven and derived-indicator-driven adaptivity.

dominated by values near to the discontinuities of the boundary data, although our mesh refinement limits do not allow the adaptivity process to act upon this. On the other hand, the Kelly indicator does not identify this area strongly, instead attributing a more evenly spread indicator to the areas of motion. Thus, if we were to use the strategy of adaptivity by fraction of error, we would expect the derived indicator to produce few refined cells away from the singularities.

In order to illustrate the effect of the two indicator choices upon the accuracy of the solution, we present another set of numerical simulations. Using the same thermo-chemical convection benchmark as before, we first calculate a reference solution on a uniform, highly refined mesh. In this case, the simulation consists of a 128-by-128 uniform grid. Since the exact solution is unknown, this high-resolution simulation

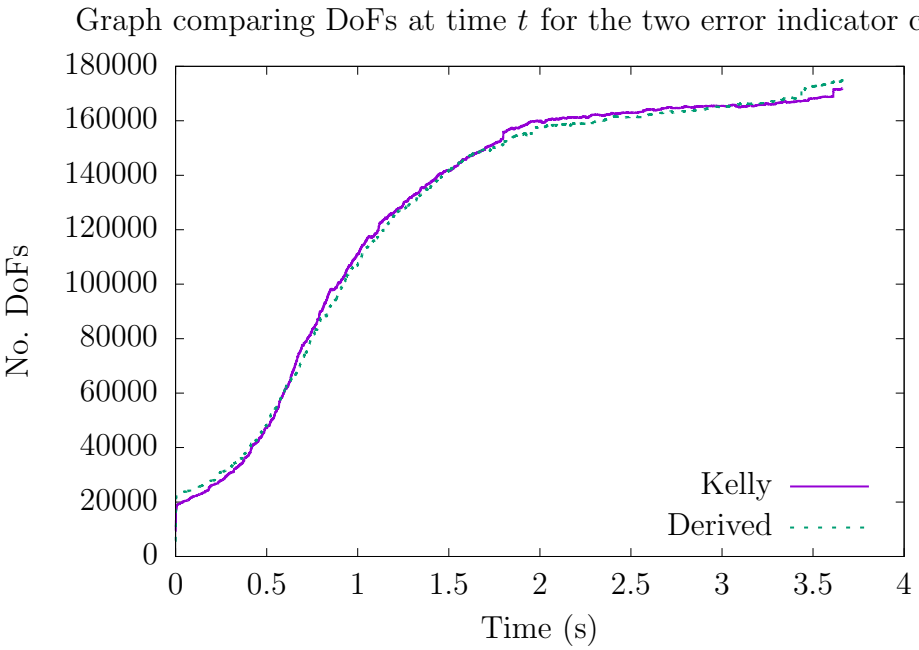
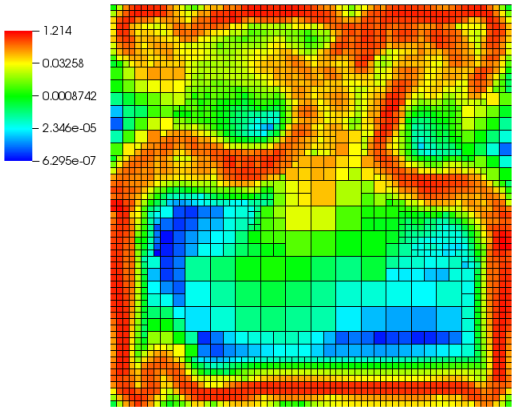


FIGURE 5.14: Graph comparing the DoF counts of two simulations, using Kelly indicator and derived indicator.

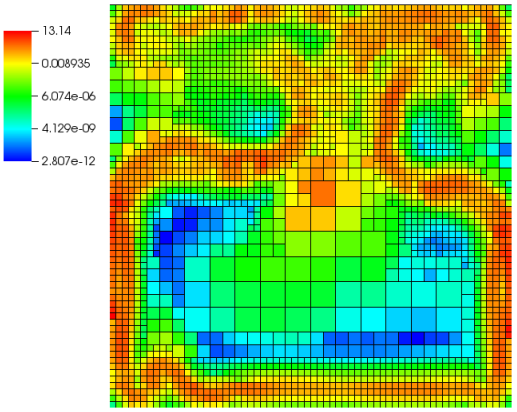
acts as a proxy for the exact solution. We are then able to calculate a proxy for the exact error by calculating the difference between the reference solution and the simulation in question, at the end of the simulation, $t = 5$. In these simulations we refer to the error in the dG norm as defined in (4.2.6), and additionally present the L^2-error for a more usual view of the solution.

With this setup, we explore the relationship between weighted average number of DoFs, and the resulting error, for a range of adaptivity parameter choices, both adaptivity indicator choices, and both adaptivity strategies. The results are presented in Figures 5.18–5.17. These demonstrate that in the L^2 norm there is little to distinguish the two adaptivity indicators under both adaptivity strategies. Figure 5.16 demonstrates a small decrease in the dG norm of the error under adaptivity strategy 1, for smaller numbers of DoFs. The results towards the right of this graph demonstrate that both adaptivity indicators are able to resolve the solution sufficiently when the initial mesh is more greatly refined. However, those results towards the left suggest the derived adaptivity indicator is in some cases able to produce a roughly 25% decrease in the dG norm of the error when the initial mesh does not fully resolve the problem. Under adaptivity strategy 2, we see no such similar effect.

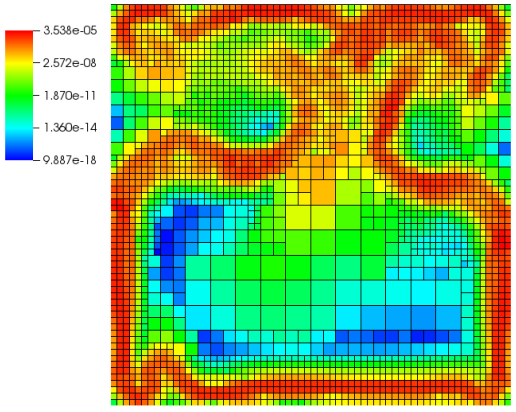
This can be explained by the fact that the two adaptivity indicators result in similar *ordering* of cells, but with different weights. Using an adaptivity strategy that ignores the relative weights of cells beyond their ordering therefore results in both adaptivity indicators recommending very similar adaptivity patterns.



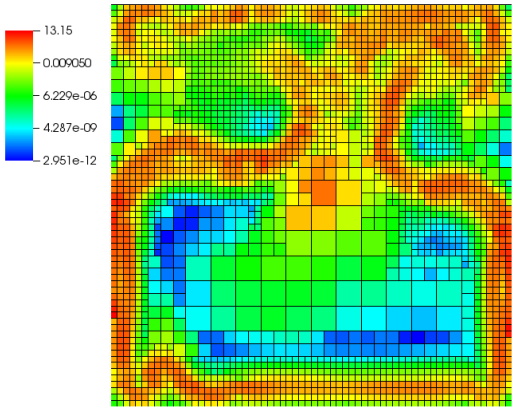
(a) Kelly indicator on a log scale



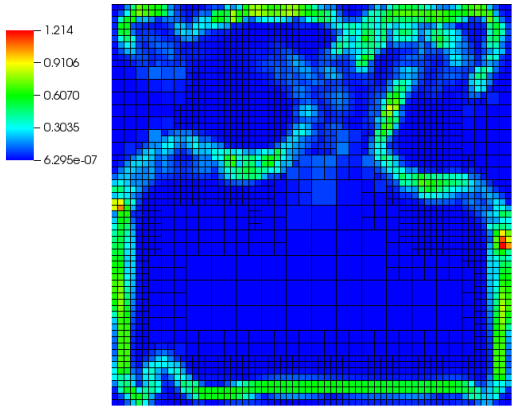
(b) Edge value jump terms on a log scale



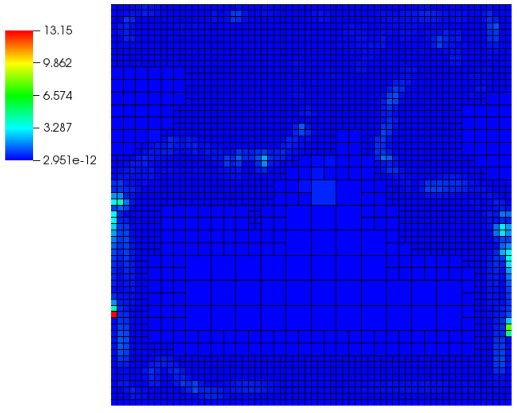
(c) Gradient jump terms on a log scale



(d) Derived indicator on a log scale



(e) Kelly indicator



(f) Derived indicator

FIGURE 5.15: Figures illustrating values of terms within the Kelly and derived indicators, for a simulation based upon the van Keken thermochemical benchmark.

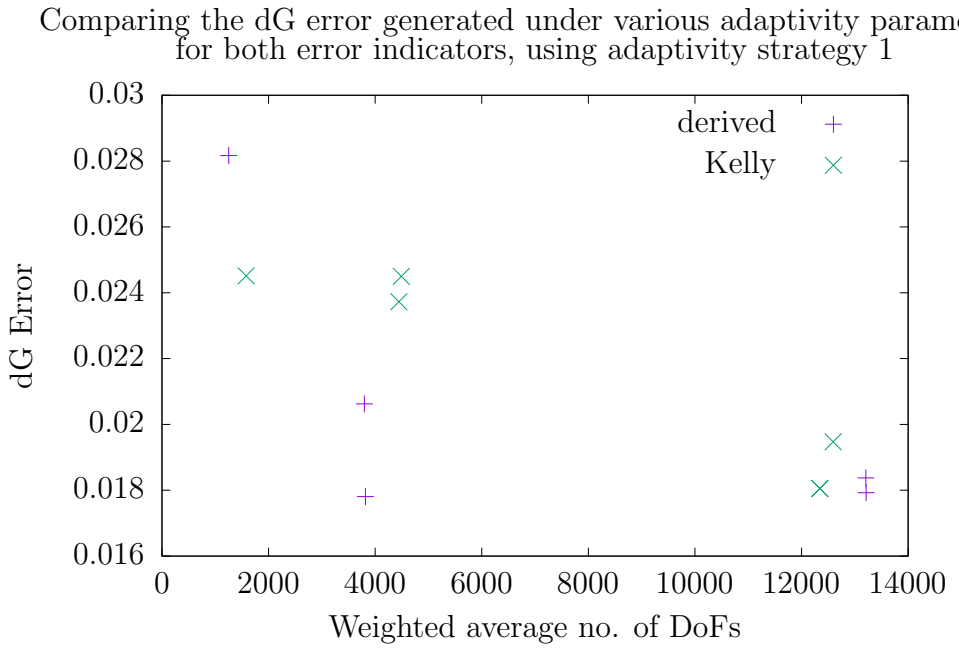


FIGURE 5.16: ψ-weighted dG error computed against a reference solution, versus average weighted DoFs, under Kelly and derived indicators, using adaptivity strategy 1.

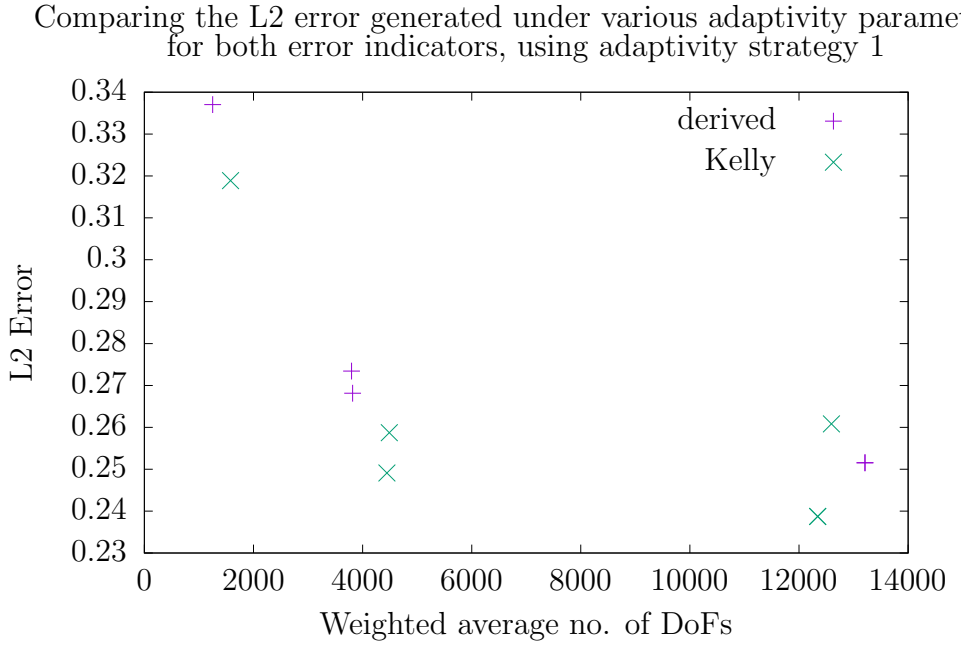


FIGURE 5.17: L^2 error computed against a reference solution, versus average weighted DoFs, under Kelly and derived indicators, using adaptivity strategy 1.

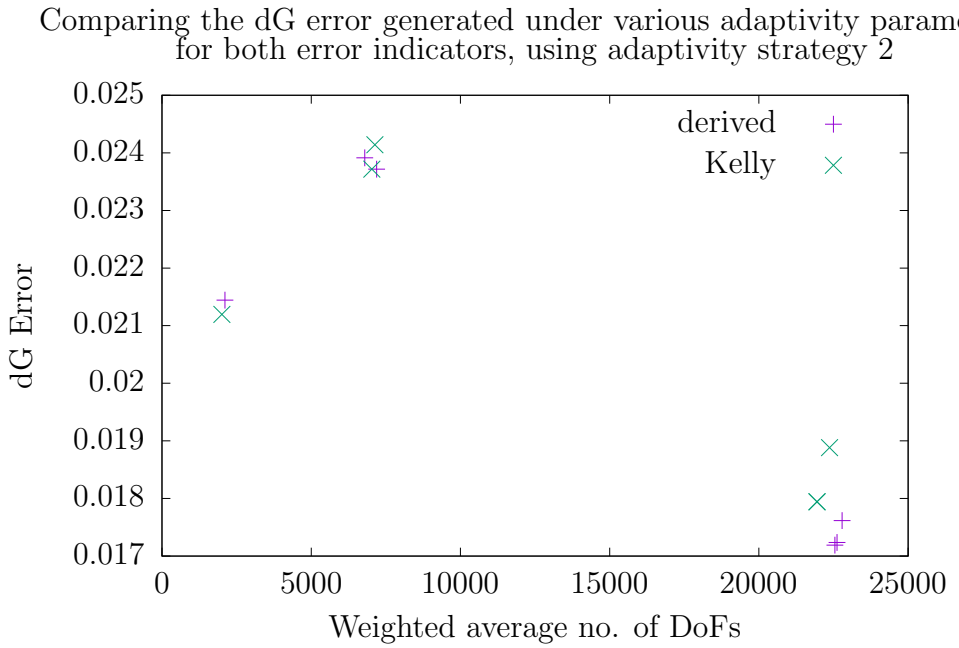


FIGURE 5.18: ψ-weighted dG error computed against a reference solution, versus average weighted DoFs, under Kelly and derived indicators, using adaptivity strategy 2.

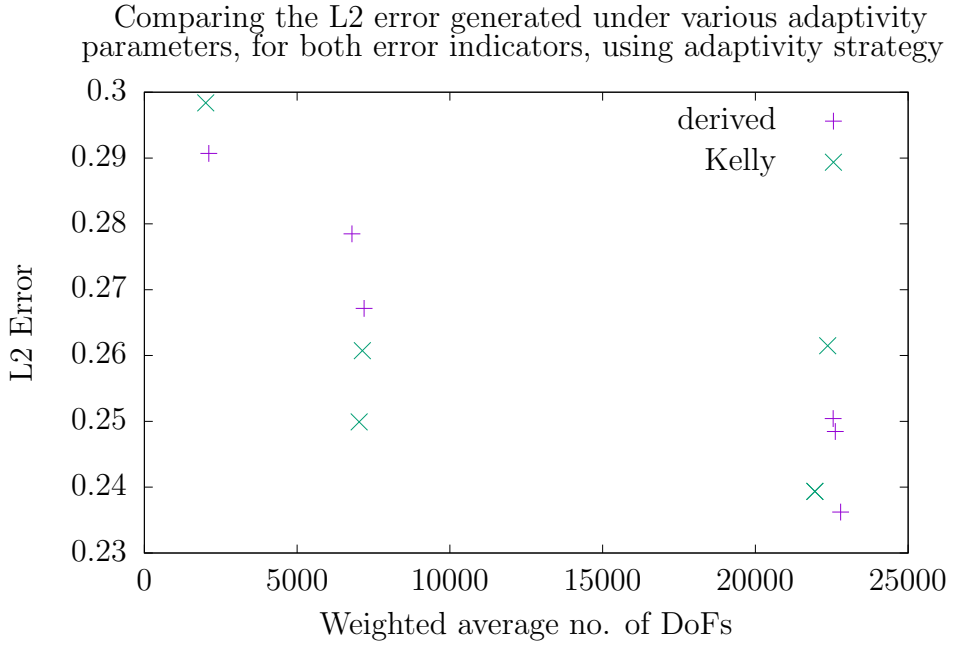


FIGURE 5.19: L^2 error computed against a reference solution, versus average weighted DoFs, under Kelly and derived indicators, using adaptivity strategy 2.

Chapter 6

ASPECT implementation

ASPECT is a community-developed and maintained mantle convection simulation code, built upon the deal.II C++ library, with a focus on extensibility and research usability. As such, it includes a great many features that the step-32 code does not, as well as improving in many areas in respect to solution speed and extensibility for use with varied scenarios. It allows modelling of equations that do not fit within the restrictions in Chapters 4 and 5, for example including compressibility, varying diffusivity, and the inclusion of compositional fields. A number of material, gravity, and heating models are available, covering a number of models used within the geodynamical modelling community. The inclusion of compositional fields allows the user to model the transport of material with varying properties within the simulation. Compositional field may advect either passively, following the flow of the material but not affecting it, or actively, in which the properties of the advected compositional field feed back into the Stokes and temperature equations, altering the course of the simulation.

This chapter discusses the implementation within ASPECT of the discontinuous Galerkin method for both the temperature and compositional fields, along with an adaptivity indicator based upon that derived in Chapter 4 and implemented in Chapter 5.

6.1 Details of implementation of dG

Implementing the dG method within ASPECT largely follows identical logic to its implementation within step-32, save that it is also implemented for the compositional fields. These fields do not diffuse, being merely advected. Thus the partial differential equation is that of time-dependent convection, lacking the terms found in the temperature equation that derive from the diffusion term.

6.2 Details of implementation of simplified estimator

Since ASPECT is a community code built mainly for use in geodynamical modelling, it is not realistic to implement the full estimator as in step-32. Indeed, this would require modifying the kernel substantially, in a way that would not reflect the needs of other users. In a research code, where the focus is upon the scientific research output for the geodynamic and geophysical communities, it would be unreasonable to make such wide-reaching changes as are necessary to implement the full estimator. Instead, we make use of the mesh refinement plugin functionality to build a self-contained plugin to guide the adaptivity process. The aim is to be able to present an error indicator for the purposes of adaptivity, in order to minimise the error of the simulation while also minimising the computational cost.

There is a balance to be struck here. Calculating the error indicator is not a free operation, and particularly so if it requires the solution of another linear system. This costs both time in computation and memory for the storage of the necessary matrices and vectors. However, balanced against this are the benefits to be gained by adapting in the 'best' way. In the worst case, a field with for example a corner singularity is dependent upon correct adaptivity to gain any accuracy – refinement far away from the singularity will yield almost no benefit compared to the benefit of a single judicious refinement in the area around the singularity.

This simple example illustrates that 'smart' adaptivity is perhaps worth investing some computation in.

Generally, it is only acceptable for the process of error indicator calculation to take a few percent of the total time. Thus, the indicator is restrained in what it is able to afford. However, adaptivity usually only occurs at intervals throughout a simulation, rather than at each timestep. This is particularly so in the case of fluid moving as slowly as mantle, with negligible inertial effects.

The error indicator uses only the simplest terms from that shown in the previous example, identically to the discussion in Section 5.7.

$$\begin{aligned}\zeta_{n,K}^2 \coloneqq{}& \rho_K^2 \left\| A^n + \varepsilon\Delta\theta_h^n - \mathbf{u}^n\cdot\nabla\theta_h^n - \delta^n\theta_h^n \right\|_K^2 \\ &+ \sum_{F\in\partial K\setminus\Gamma} \rho_{\omega_F} \left\| [\![\varepsilon\nabla\theta_h^n]\!] \right\|_F^2 \\ &+ \sum_{F\in\partial K} \left(\frac{\sigma\varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F}\sigma\varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla\eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \right. \\ &\qquad \left. + h_F \|\mathcal{L}\|_{\psi,\omega_F,\infty} + \frac{\overline{\psi}_{\omega_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha\varepsilon\nabla\eta\|_{\omega_F,\infty}^2 \right) \left\| [\![\theta_h^n]\!] \right\|_F^2 .\end{aligned}$$

In particular, this does not contain any of the estimator terms defined on the union mesh $\mathcal{T}_h^{n-1} \cup \mathcal{T}_h^n$, nor does it contain the term $\zeta_{S_3,n}$ which bounds the term $\left\|\theta_h^d\right\|_{\psi,L^\infty(0,t^n;L^2(\Omega))}^2$.

Note that we make the approximation that maxima and minima over $\tilde{\omega}_F$ are instead computed only over ω_F.

We also make another adjustment to the cell residual term. We recall (4.6.2), which states that

$$A^n = \Pi^n \left(f^n + \delta^n\theta_h^n \right) - \left(\theta_h^n - \Pi^n\theta_h^{n-1} \right) / \tau^n,$$

where Π^n is the L^2-projection into $X_{h,1}^n$. We are unable to fully compute A^n without projecting the old temperature θ_h^{n-1} forwards onto the new mesh. As we do not have the ability to cheaply project forward, but we do have access to the interpolated previous solution, we make the approximation that $\Pi^n\theta_h^{n-1} \approx I_h^n\theta_h^{n-1}$, where I_h^n is the interpolation operator onto V_h^n. This difference can be bounded, of optimal order. Additionally, in most computations we do not adapt on every timestep, but on every k timesteps, for some $k \geq 2$. In this scenario, each time that we evaluate the adaptivity indicator, the current and previous meshes will be identical, and

thus $\theta_h^{n-1} = I_h^n \theta_h^{n-1} = \Pi^n \theta_h^{n-1}$. We note that, additionally, in geophysically relevant scenarios we expect to take large timesteps τ^n. Thus it suffices to calculate $\tilde{A}^n = \Pi^n \left(f^n + \delta^n \theta_h^n\right) - \left(\theta_h^n - I_h^n \theta_h^{n-1}\right)/\tau^n$, in the knowledge that in almost all useful scenarios this will be identical to (4.6.2), and in circumstances where it is not identical then it can be bounded with optimal order. The error indicator is then

$$\begin{aligned}\zeta_{\mathrm{ASP},n,K}^2 \coloneqq \rho_K^2 \left\| \Pi^n \left(f^n + \delta^n \theta^n\right) - \frac{\theta_h^n - I_h^n \theta_h^{n-1}}{\tau^n} + \varepsilon \Delta \theta_h^n - \mathbf{u}^n \cdot \nabla \theta_h^n - \delta^n \theta_h^n \right\|_K^2 \\ + \sum_{F \in \partial K \setminus \Gamma} \rho_{\omega_F} \left\| [\![\varepsilon \nabla \theta_h^n]\!] \right\|_F^2 \\ + \sum_{F \in \partial K} \left(\frac{\sigma \varepsilon}{h_F} \left(\overline{\psi}_{\omega_F} + \varrho_{\omega_F} \sigma \varepsilon + \frac{\overline{\psi}_F \alpha^2 \varepsilon \overline{\nabla \eta}_F^2}{\underline{\mathcal{L}}_{\omega_F}} \right) + \rho_{\omega_F} \|\mathbf{u}\|_{F,\infty}^2 \right. \\ \left. + h_F \|\mathcal{L}\|_{\psi,\omega_F,\infty} + \frac{\overline{\psi}_{\omega_F} h_F}{\varepsilon} \|\mathbf{u} - \alpha \varepsilon \nabla \eta\|_{\omega_F,\infty}^2 \right) \left\| [\![\theta_h^n]\!] \right\|_F^2 .\end{aligned}$$

This estimator is computed using the same algorithms as in Chapter 5, iterating over locally owned cells and, as a subiteration, looping over faces of each cell.

6.3 Comparisons with alternative strategies

In this section, we examine a number of simple examples, which highlight the differences between the existing strategies (using FEM and the Kelly error indicator) and the new strategies (using dG and the derived error indicator).

6.3.1 Example 1a

In our first example, we compare the behaviour of an FE simulation against a dG simulation. We choose the example of a box domain, with prescribed, variable velocity boundary conditions. This example is found in the ASPECT manual, §5.2.3, and begins with a smooth temperature gradient through the domain, from 0 at the top to 1 at the bottom. Flow is thus a product of buoyancy and of the boundary velocities.

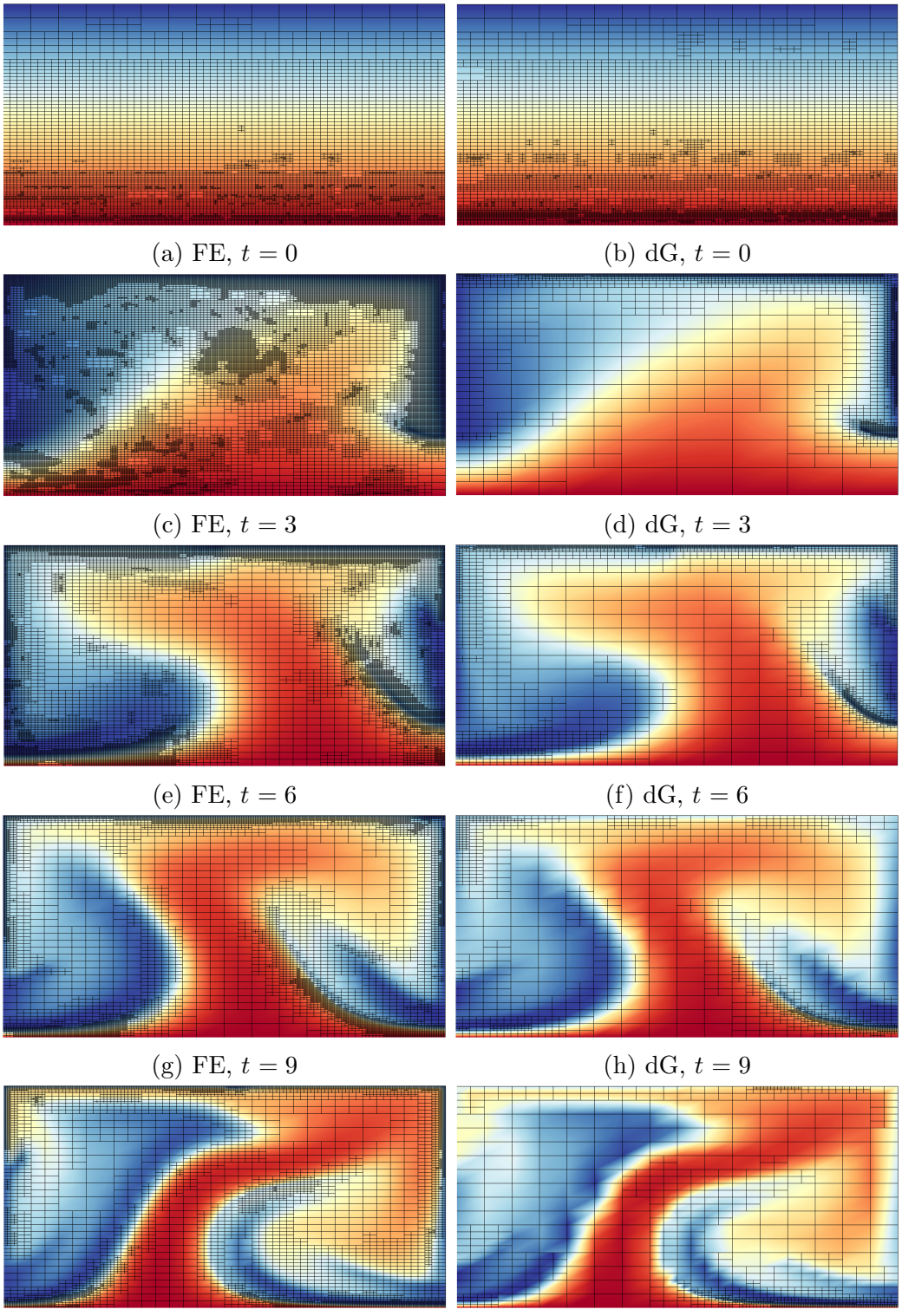
We examine the case where we use the default ASPECT mesh adaptivity algorithm, with default parameter choices, and set 6 initial refinements and 2 adaptive refinement stages. The only change is the discretisation choice for the temperature variable. Figure 6.1 compares the effect of the choice of discretisation upon the Kelly error indicator, the mesh adaptivity, and the full solution. Here, we use adaptivity strategy 1, that marks cells for adaptation based on their contribution to the total error.

We observe that the dG discretisation results in the Kelly indicator focussing on a smaller area for refinement, and allowing more cells to be coarsened. The resultant lower number of DoFs (see Fig 6.2) results in a faster execution time (on 12 processors, this simulation took $1.43 \cdot 10^4$ s for the FE case, and $3.87 \cdot 10^3$ s for the dG case). It is clear that the dG discretisation highlights the region at the front of the downwelling strongly, as well as the region of the discontinuity in the variable velocity on the upper boundary. However, it is not clear whether this results in a better per-cost solution. On the contrary, it seems that the combination of the dG method and the Kelly error indicator means that the indicator is dominated by a small number of cells, so that the mesh adaptivity strategy ignores most other areas, to the detriment of the total solution.

6.3.2 Example 1b

We re-run the same experiment, but this time with 5 levels of initial global refinement and 2 adaptive refinements, and with adaptivity strategy 2, refining a given percentage of cells, rather than the cells that make up a certain percentage of the total estimated error. This is an attempt to overcome the difficulties of the previous example, in which a small number of cells dominated the Kelly error indicator. The results are shown in Figures 6.3 and 6.4. These show that the dG discretisation initially results in a lower number of DoFs, but this gradually grows. On the other hand, the FE method quickly reaches a large number of DoFs, but then plateaus.

The proportions of cells to refine (10%) and coarsen (5%) are such that, given entirely free reign, the adaptivity algorithm should reduce the number of cells on average. However, since we limit the refinement levels allowed, and smooth the mesh to avoid



(a) FE, $t = 0$ (b) dG, $t = 0$

(c) FE, $t = 3$ (d) dG, $t = 3$

(e) FE, $t = 6$ (f) dG, $t = 6$

(g) FE, $t = 9$ (h) dG, $t = 9$

(i) FE, $t = 12$ (j) dG, $t = 12$

FIGURE 6.1: Example 1a, comparing FE and dG methods under Kelly refinement with adaptivity strategy 1.

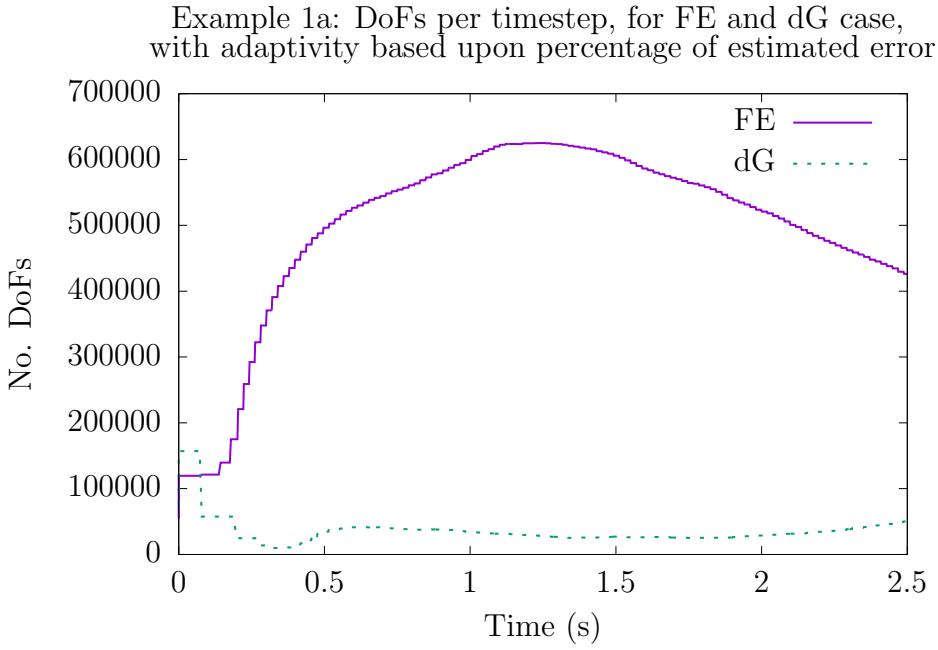
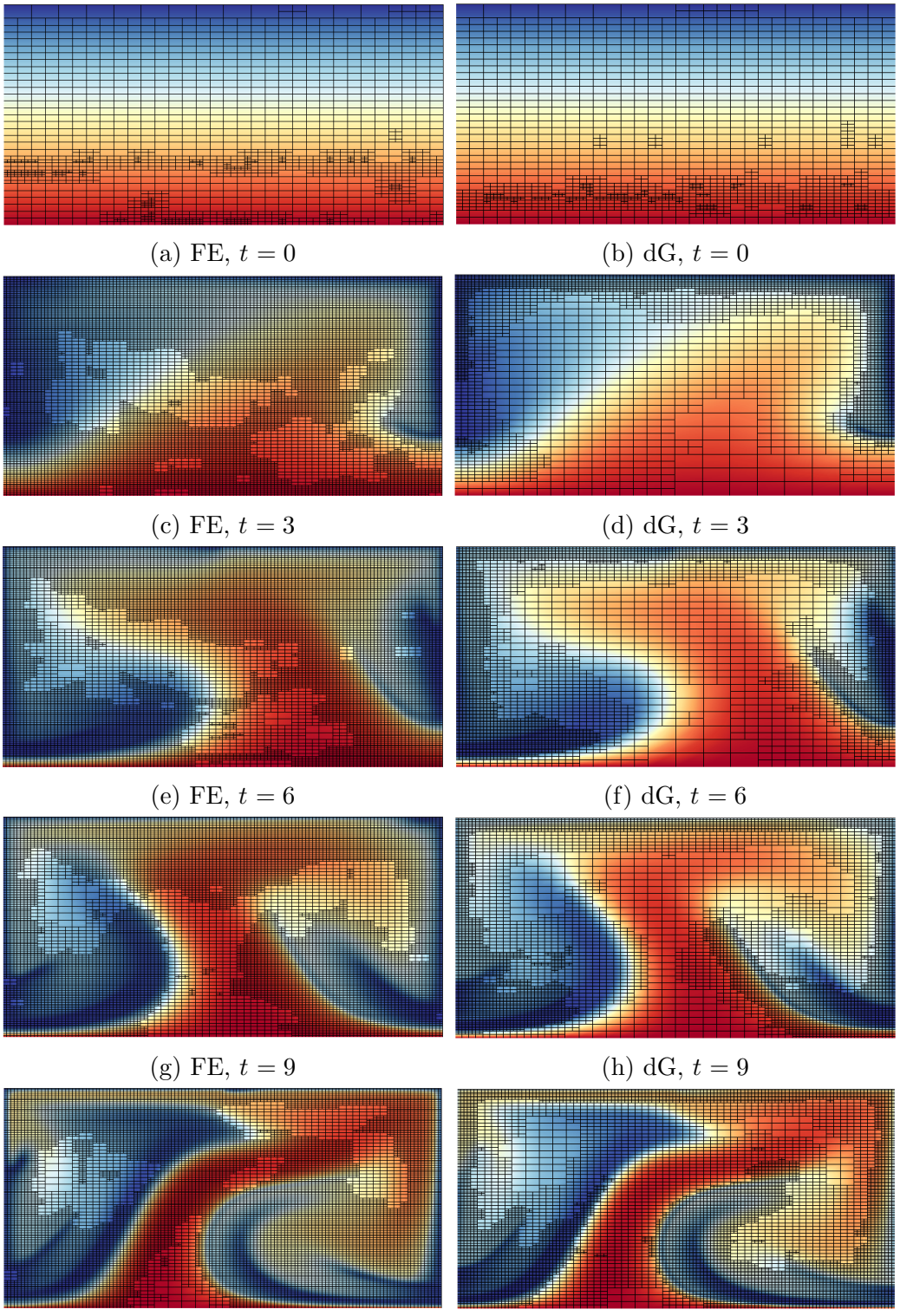


FIGURE 6.2: Example 1a: DoF count per timestep, for the FE and dG methods under Kelly refinement, using adaptivity strategy 1.

difficulties, we see a net gain in the numbers of cells. These results suggest that the FE method results in the Kelly indicator progressively refining cells in different parts of the mesh, with refinement significantly affecting the indicator, so that more refined cells no longer feature amongst the greatest contributors to the error. On the other hand, the dG method seems to result in an indicator behaviour which is not so strongly affected by the mesh size, and is more strongly affected by the values of the solution itself. Thus, refinement of cells does not always significantly reduce their ordering amongst the largest contributors to the error.

6.3.3 Example 2

In this example, we compare the results of using the dG method versus the FE method, in the case of a purely hyperbolic problem: composition-based flow. The van Keken composition benchmark, introduced in [63], models a Rayleigh-Taylor instability between two fluids of differing density. A fluid of density 1010 sits above a second fluid of density 1000, in a box of size 0.9142–by–1, with zero velocity at the top and bottom boundaries, and free-slip flow on the left and right. The less



(a) FE, $t = 0$ (b) dG, $t = 0$

(c) FE, $t = 3$ (d) dG, $t = 3$

(e) FE, $t = 6$ (f) dG, $t = 6$

(g) FE, $t = 9$ (h) dG, $t = 9$

(i) FE, $t = 12$ (j) dG, $t = 12$

FIGURE 6.3: Example 1b, comparing FE and dG methods under Kelly adaptive refinement with adaptivity strategy 2.

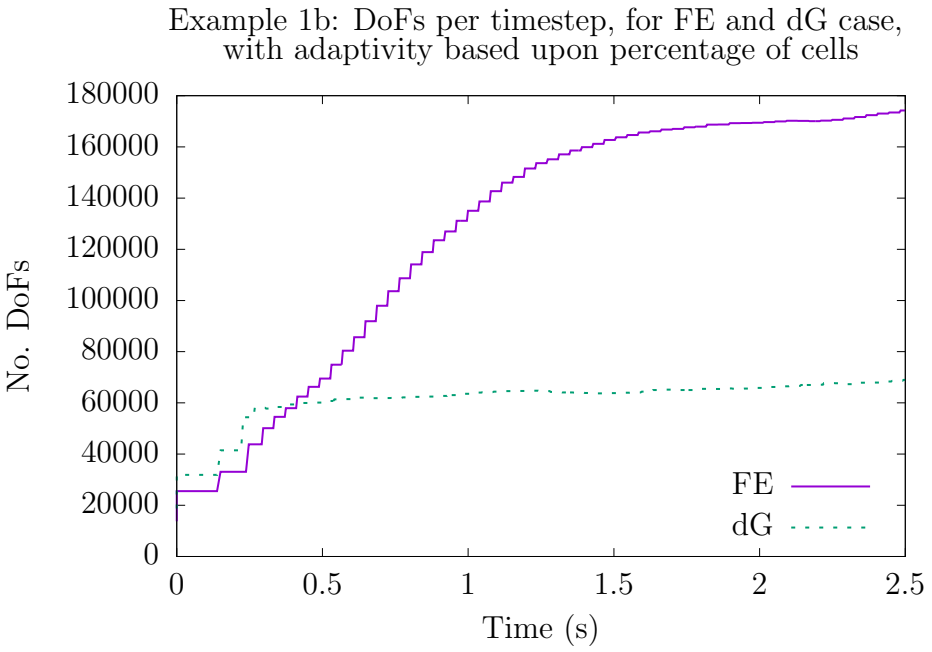


FIGURE 6.4: Example 1b: DoF count per timestep, for the FE and dG methods under Kelly refinement, using adaptivity strategy 2.

dense material is initially at the base of the box, with a perturbed top surface (see Fig 6.5). The simulation is run for 2000 simulated seconds. The temperature field is zero everywhere, thus the flow is entirely buoyancy-driven.

In our example, we compare the results from discretisation of the compositional field by FE and dG elements, on a fixed, uniform grid. For a compositional field, we set the diffusion parameter to zero: even in the hyperbolic limit, the dG upwind flux results in a stable method. Thus we are interested in this example not in the indicator choice, but in the ability of the dG method to approximate a sharp moving boundary, and a zero-diffusion field.

Since the composition fields follow a purely hyperbolic flow law, there should be no diffusion of the composition field. Figure 6.6 demonstrates that the use of the dG method can effectively conserve the sharp interfaces of the composition field, resulting in much less 'smearing' of the field as time increases. This is in comparison to the FE case, which imposes an artificial diffusion term to stabilise the field. However, it should be noted that the dG method also has drawbacks in this situation. Firstly, it is more expensive to compute, as illustrated in Table 6.1: the increase in DoFs translates into increased setup time for the DoF systems, and the assembly

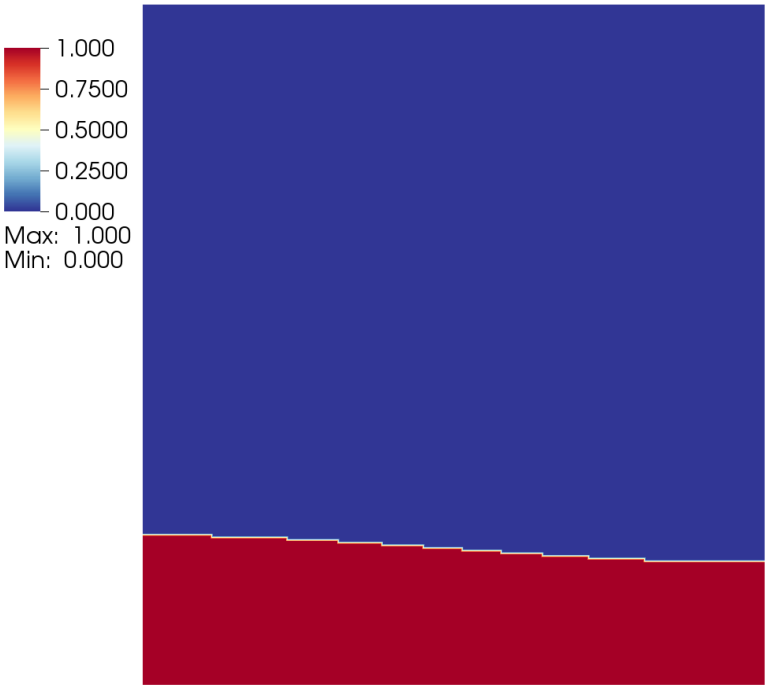
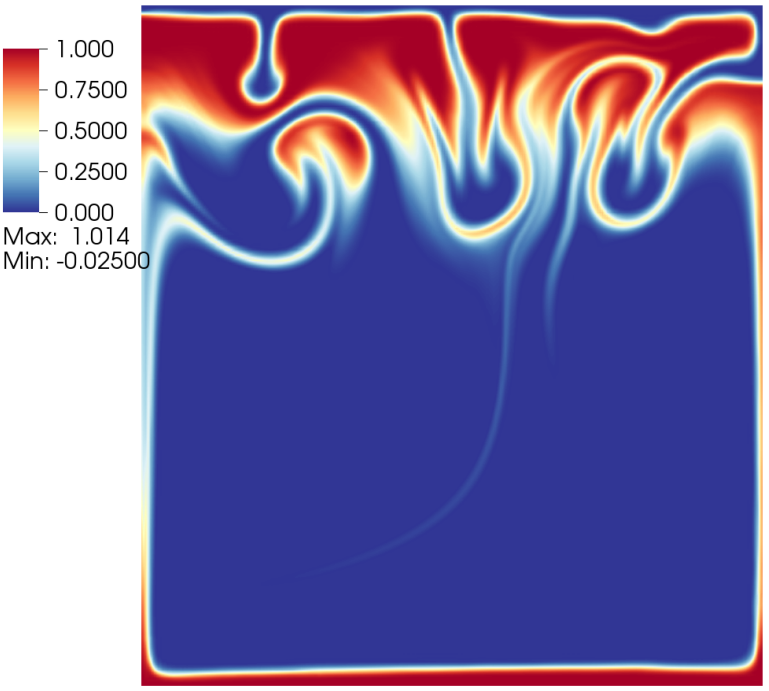


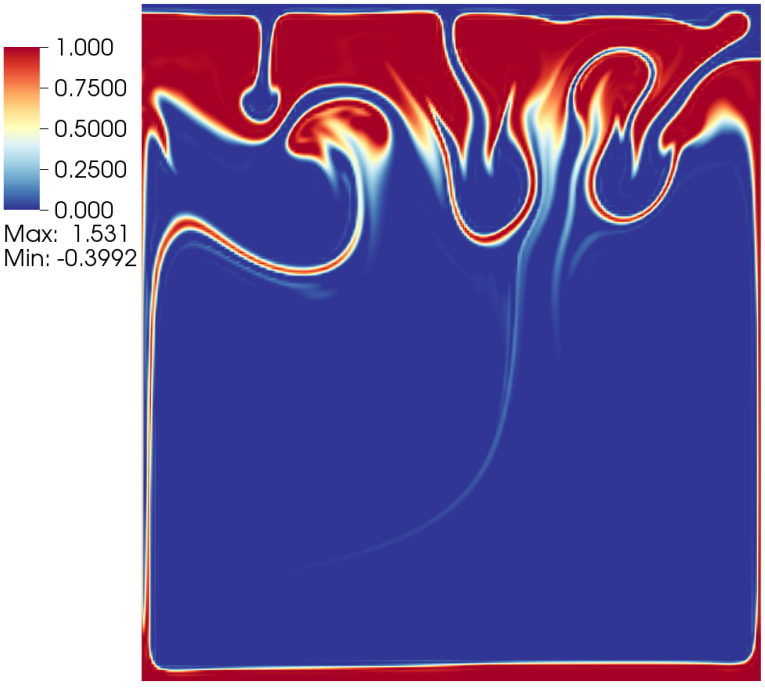
FIGURE 6.5: Example 2: The initial distribution of the less dense composition in the van Keken benchmark, on a 7-times refined mesh.

time for the composition system, and composition preconditioner, are both increased significantly, leading to a slower computation. It is expected that the cost of the preconditioner could be reduced, along with the composition solve, by adjusting parameters to values more suited to the form of the dG system – the values used in this simulation are those which have been shown to work well in the FE case, and thus there is further research to do to identify the best parameter choices for the dG case.

Secondly, the dG method produces overshoots and undershoots near to the discontinuities. This is a sign that we are not fully resolving the solution with this mesh-size. Worse still, however, is that these overshoots can be a numerical instability within the simulation, since material properties dependent on composition concentration may be ill-defined for negative, or larger than one, values of composition concentration. Thus further work is necessary to limit the size of these overshoots for use within complex material models. The dG method is able to incorporate flux limiters, and such techniques have in fact been implemented in ASPECT for this reason [52], limited to the case of divergence-free flow, building on the methods introduced in



(a) FE discretisation



(b) DG discretisation

FIGURE 6.6: Example 2: A comparison between (A) FE and (B) dG discretisation for the van Keken isoviscous composition benchmark. Mesh used is refined 7 times. Field shown is the value of the less dense fluid, at time $t = 2000$.

Section	FE time (s)	dG time (s)
Assemble composition system	268	514
Build composition preconditioner	38.5	201
Solve composition system	45.9	55.1
Assemble Stokes system	89.6	80.6
Solve Stokes system	388	368
Setup DoF systems	0.38	2.52
Total	950	1320

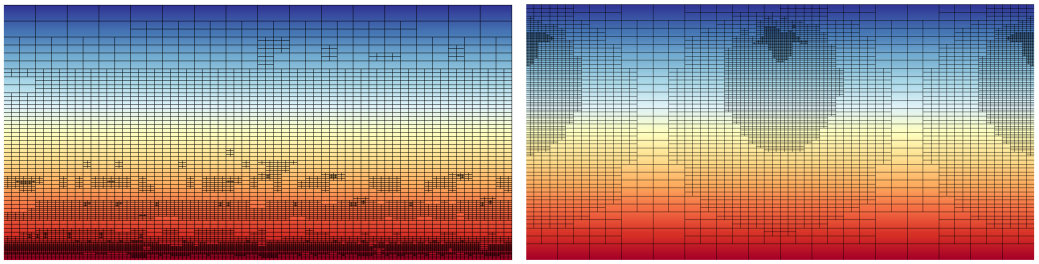
TABLE 6.1: Comparison of FE and dG method computational time for the van Keken benchmark, on 8 CPUs. Figures shown cover only some sections of the code, thus the total per run is not the sum of the individual timings shown. Composition DoFs number 66,049 for the FE case, and 147,456 for the dG case. Stokes DoFs number 148,739 in both cases.

[115] and [116], although we opt not use this in this work, in order to separate the effect of the dG method from the limiter.

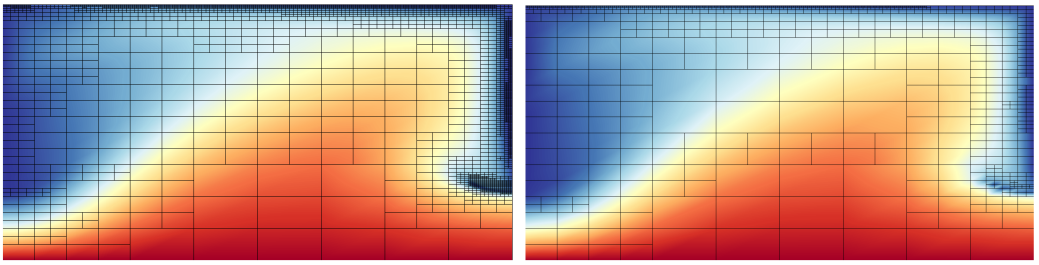
6.3.4 Example 3a

We return to the case of variable velocity boundary conditions, as used in Example 1. We compare the effect of the choice of adaptivity indicator upon the solution and mesh, in the case where both simulations use the dG method. We begin again with a linear vertical temperature profile from 0 to 1, use the default ASPECT mesh adaptivity algorithm parameter choices, and set 6 initial refinements and 2 adaptive refinement stages. We examine the difference between the Kelly error indicator and the derived error indicator. We employ adaptivity strategy 1, refining and coarsening cells responsible for a fraction of the total indicated error.

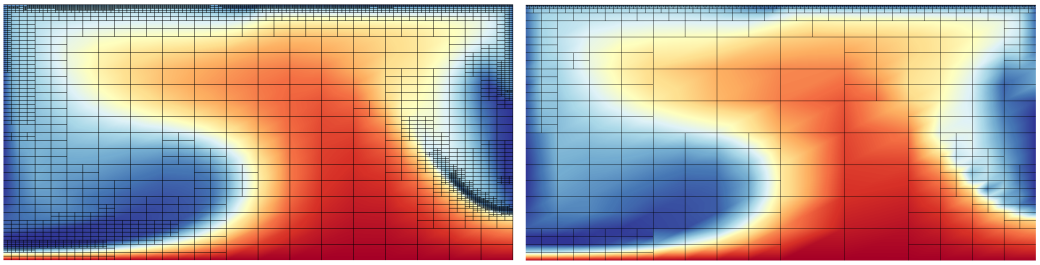
Figure 6.7 clearly indicates that in this case the derived error indicator is overly-focussed upon the top boundary, which contains a large velocity area. Under these conditions, it is clear that the error indicator is dominated by the velocity contributions to the face-jump term. In this example, the derived indicator results in a faster solve, because of the reduced number of DoFs (see Fig 6.8), but it is unlikely to be resulting in a more accurate solution, being overwhelmed by the velocities at the upper boundary, and thus neglecting the rest of the simulation.



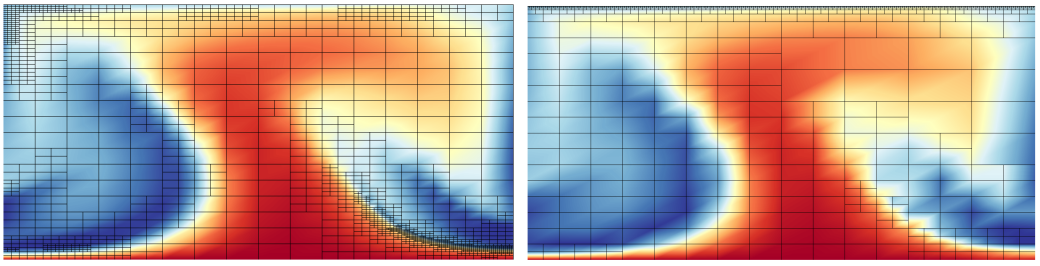
(a) Kelly, $t = 0$ (b) Derived, $t = 0$



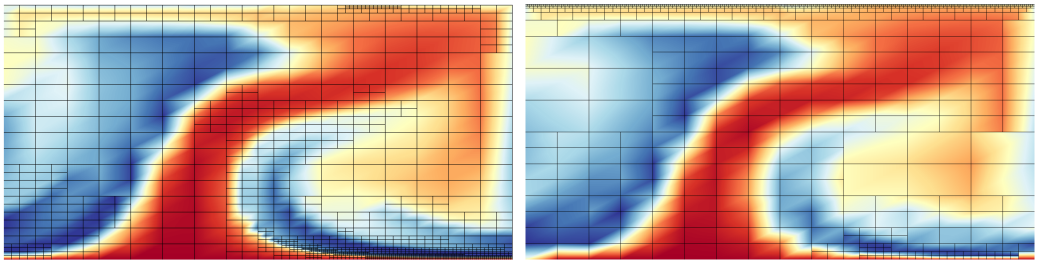
(c) Kelly, $t = 3$ (d) Derived, $t = 3$



(e) Kelly, $t = 6$ (f) Derived, $t = 6$



(g) Kelly, $t = 9$ (h) Derived, $t = 9$



(i) Kelly, $t = 12$ (j) Derived, $t = 12$

FIGURE 6.7: Example 3a: Comparing mesh behaviour under Kelly and derived error indicators, adaptivity strategy 1.

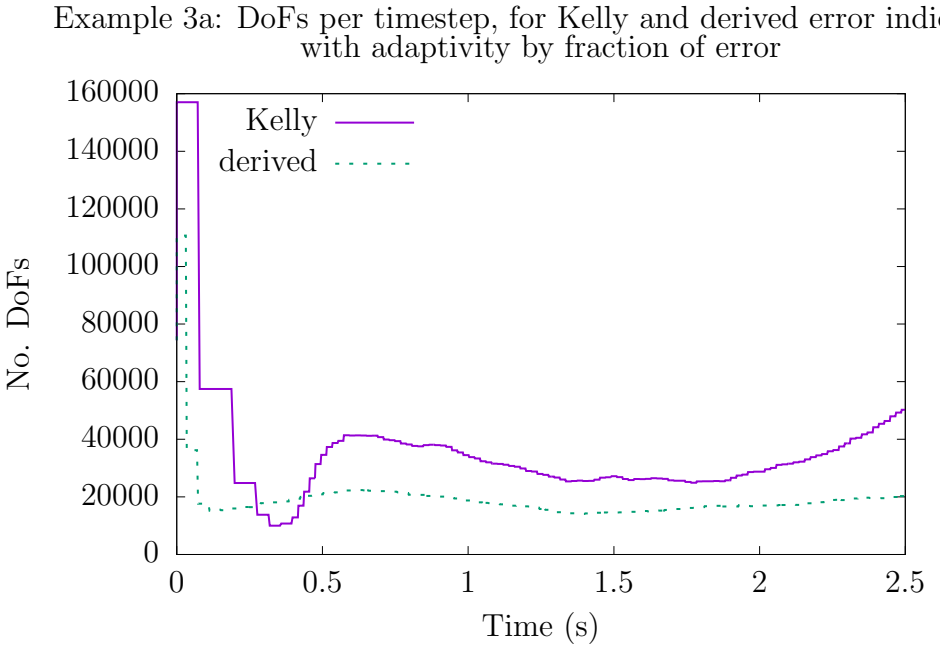


FIGURE 6.8: Example 3a: DoF count per timestep, for the dG method under Kelly and derived indicator refinement, using adaptivity strategy 1.

6.3.5 Example 3b

To explore the effect of the dominance of a few cells upon the total error, we run an identical simulation, but this time use adaptivity strategy 2, refining a fixed fraction of the total cells. This is shown in Figure 6.10.

This shows much closer agreement between the Kelly and derived indicator strategies, suggesting that, while Example 3a showed that the derived indicator shows the majority of indicated error to be at the top of the domain, the overall ordering of cells by indicator size remains similar between the two indicators. Figure 6.9 shows that, after the derived indicator initiates an extended period of initial refinement, the two indicators follow very similar paths in terms of total cells added at each timestep.

6.3.6 Example 4

In our final example, we compare three 3D simulations, this time based on the example in the ASPECT manual §5.2.2. In the first simulation, we use the FE method

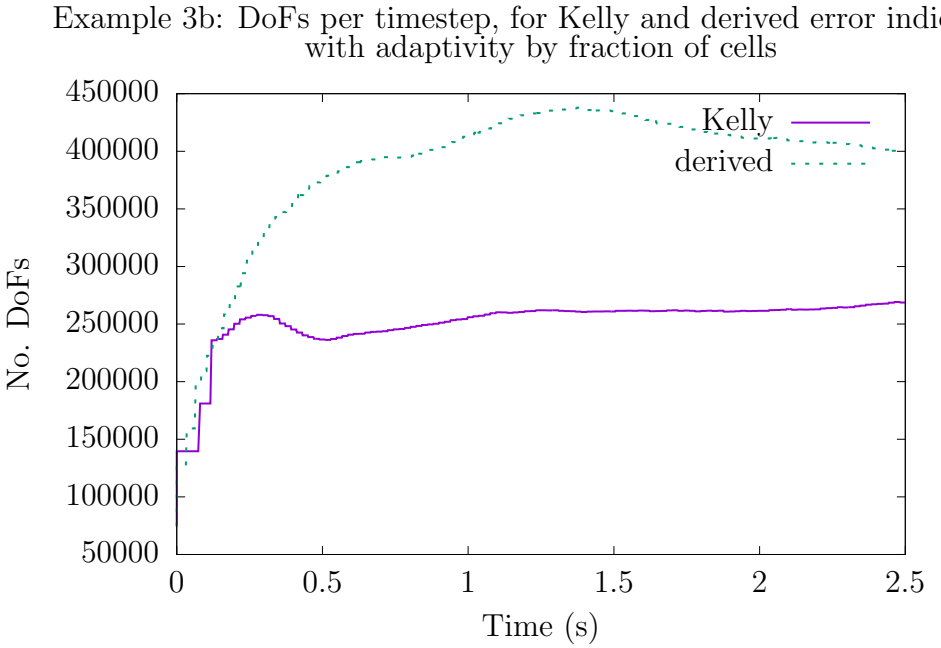
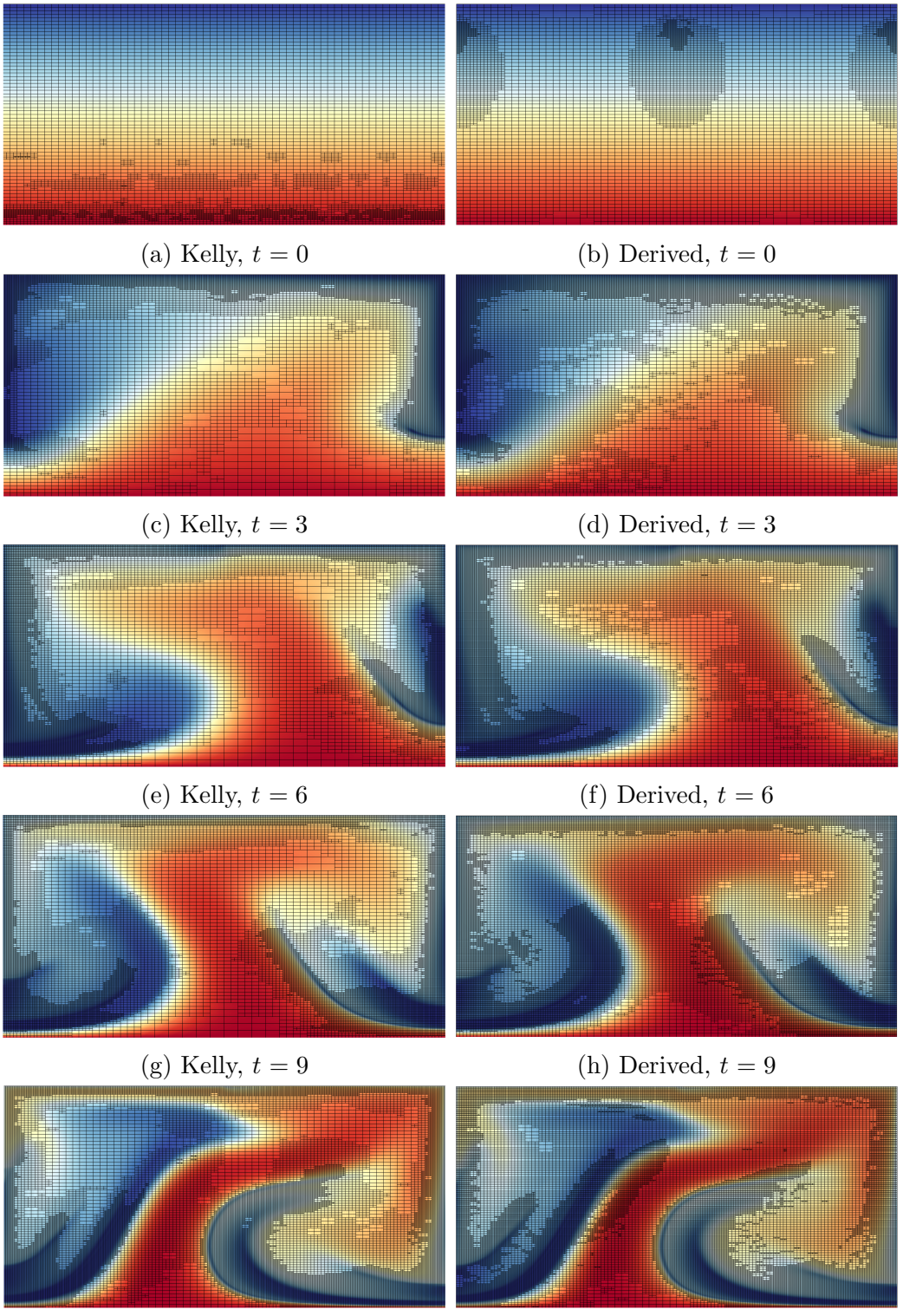


FIGURE 6.9: Example 3b: DoF count per timestep, for the dG method under Kelly and derived indicator refinement, using adaptivity strategy 2.

and the Kelly indicator; in the second, we use dG and the Kelly indicator; and in the third we use the dG method with the derived indicator. Figure 6.11 shows the solution from the three simulations, using isocontours to visualise the temperature field in 3D. This shows the methods show large agreement in the solution.

Figures 6.12 and 6.13 compare the meshes generated adaptively by the three methods. Figure 6.12 shows the outer surface of the meshes, and Figure 6.13 shows the full meshes. It is clear that the Kelly indicator generates similar meshes in both the FE and dG case, but the derived indicator favours more localised refinement, resulting in a less-refined mesh overall. This is very evident in the disparity between the cell numbers shown in Figure 6.14.



(a) Kelly, $t = 0$ (b) Derived, $t = 0$

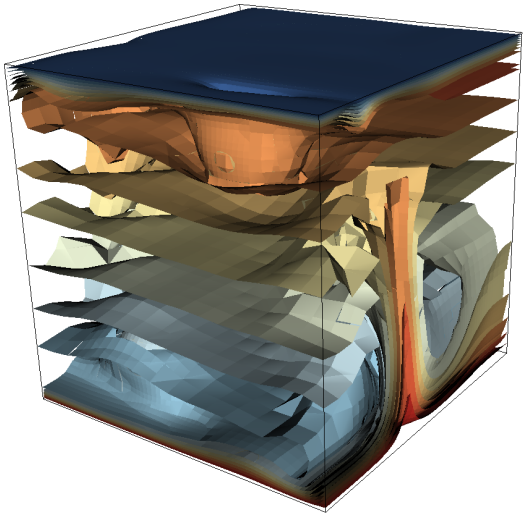
(c) Kelly, $t = 3$ (d) Derived, $t = 3$

(e) Kelly, $t = 6$ (f) Derived, $t = 6$

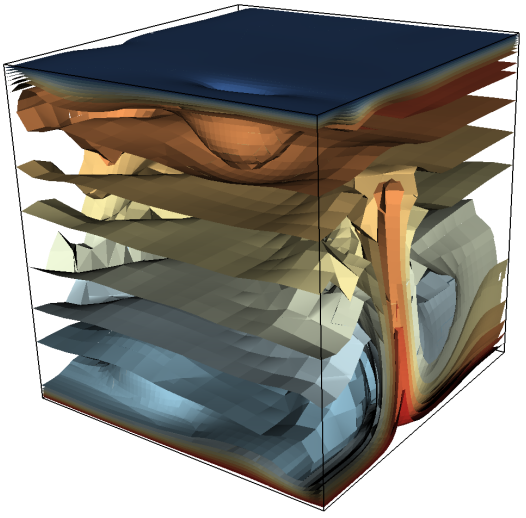
(g) Kelly, $t = 9$ (h) Derived, $t = 9$

(i) Kelly, $t = 12$ (j) Derived, $t = 12$

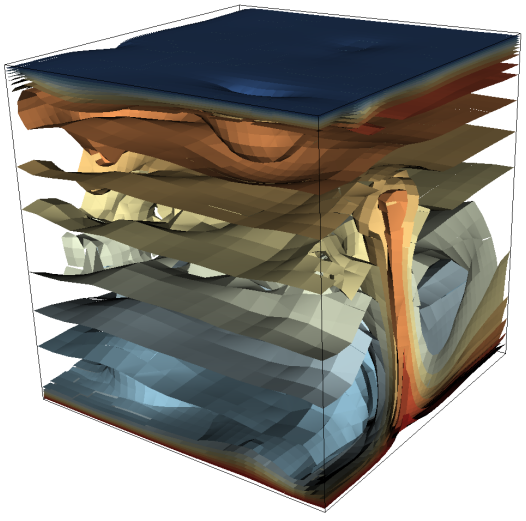
FIGURE 6.10: Example 3b: Comparing mesh behaviour under the Kelly and derived error indicators, adaptivity strategy 2.



(a) FE, Kelly indicator

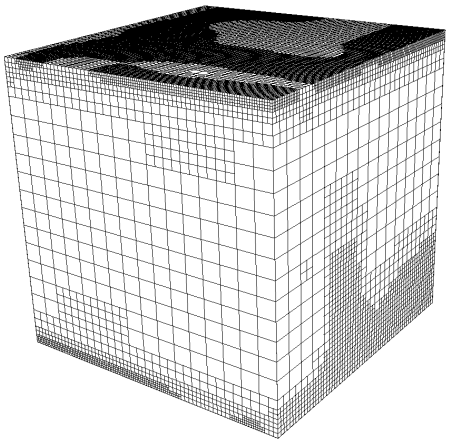


(b) dG, Kelly indicator

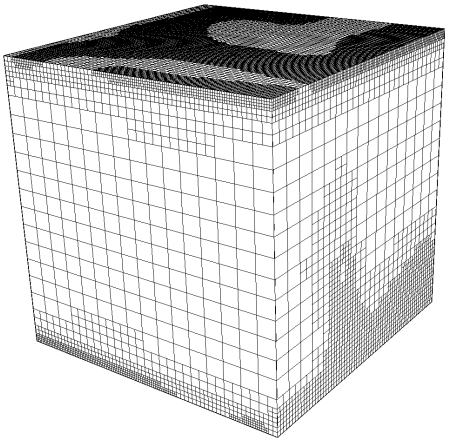


(c) dG, Derived indicator

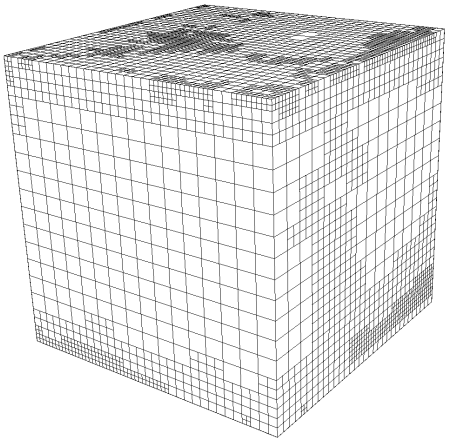
FIGURE 6.11: Example 4: Comparison of solutions between the three methods



(a) FE, Kelly indicator

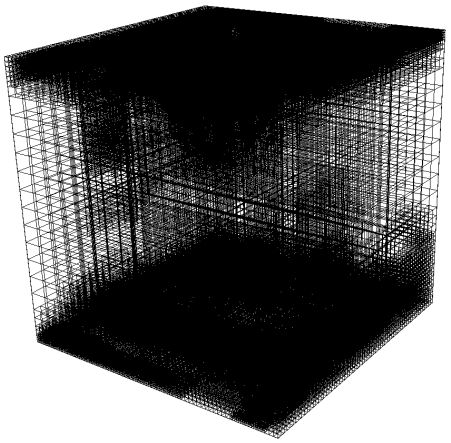


(b) dG, Kelly indicator

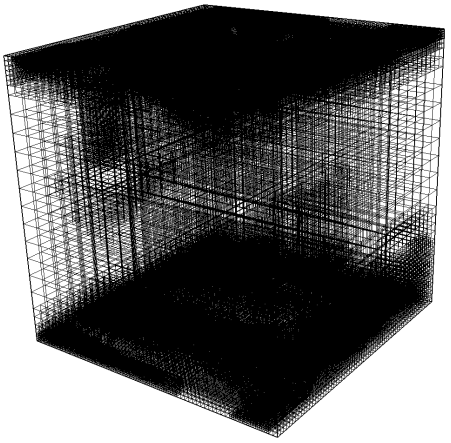


(c) dG, Derived indicator

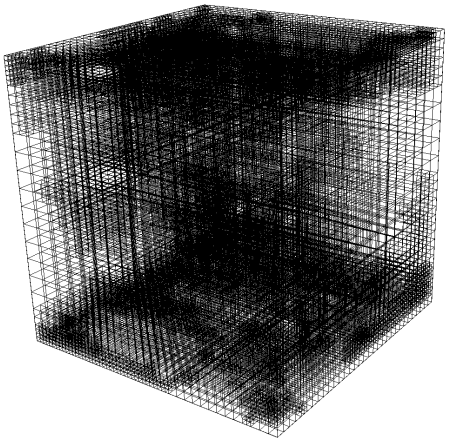
FIGURE 6.12: Example 4: Comparison of outer meshes generated by the three methods



(a) FE, Kelly indicator



(b) dG, Kelly indicator



(c) dG, Derived indicator

FIGURE 6.13: Example 4: Comparison of full meshes generated by the three methods

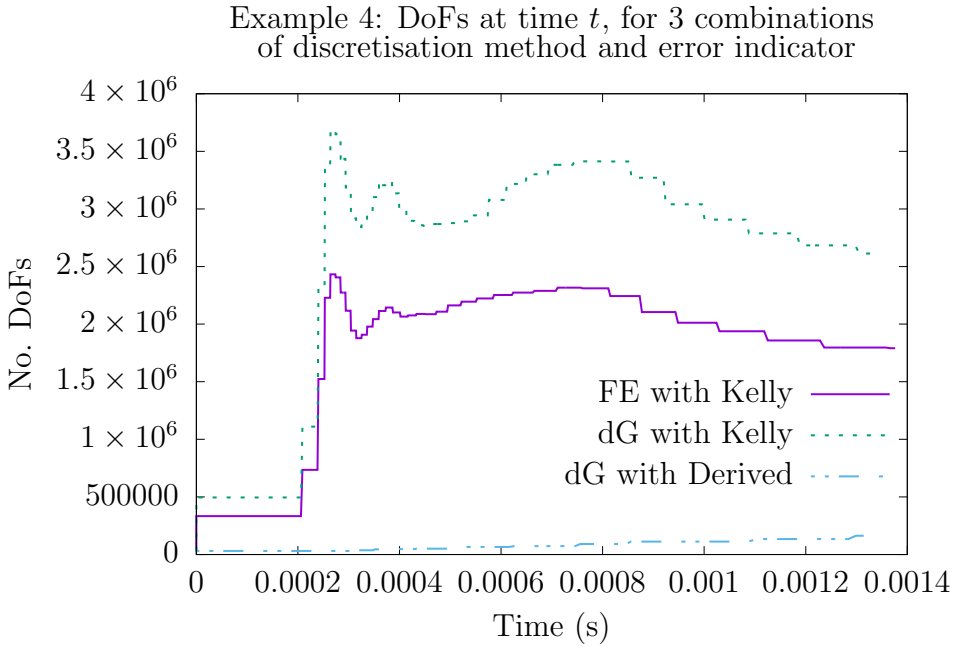


FIGURE 6.14: Example 3b: DoF count per timestep, for the three combinations of discretisation and indicator.

Chapter 7

Conclusions and future work

In this thesis, we have improved and expanded the mathematical tools available in the simulation of mantle convection. We have derived an *a posteriori* error bound for the convection-diffusion equation, in a modified norm, without the usual restrictions placed upon the divergence of the velocity field. This bound is subject to an exponential term in the event of non-negative divergence. However, this must occur in any result capable of handling a general flow. Further work remains to understand the full consequences of varying choices of parameter α in this bound, and to identify the exact circumstances under which this result improves on existing known bounds. The error bound leads to an adaptivity indicator designed for the problem in question, enabling the adaptivity strategy to be guided in a more rigorously supported fashion.

We have presented an implementation of the discontinuous Galerkin method in a convection simulation code, coupled to an FE Stokes simulation, built upon the step-32 tutorial code of deal.II. This code is parallelised for a distributed computing setting, and makes use of well-tested and documented code libraries to ease its development and maintenance. It includes a novel auxiliary-mesh method to enable calculation of terms defined over the union mesh between meshes at adjacent timesteps, while preserving the distributed computing capabilities. This bears fruit in illustrating the behaviour of the *a posteriori* error bound and adaptivity indicator within simulations of interest. We intend to use this code for further numerical experiments, to better understand the regimes in which the derived indicator is more

effective than an *ad hoc,* cheaper indicator, particularly in the setting of more complex geometries and geodynamically-relevant flow regimes. We also intend to use this to address the issue of appropriate choice of parameters and the behaviour of the error bound in situations which do not easily lend themselves to analysis.

An area which we have not explored is the use of the discontinuous Galerkin method in the Stokes discretisation. This offers the hope of improving the solution process for this part of the algorithm, which in most cases in 3D dominates the computational time. In addition to this, an *a posteriori* error bound upon the full nonlinear system of convection-diffusion coupled to Stokes flow is not yet possessed. Work on this should lead to an adaptivity indicator that takes into account the mutual dependence of the Stokes flow and temperature solution, and is truly designed for the entire solution algorithm.

Finally, we presented an implementation of the dG method within the community code ASPECT, and demonstrated its abilities in reducing the number of cells required in some simulations. Much work is needed to understand how to best precondition the resulting systems. These exhibit a natural block structure that could be the basis for further attempts to improve the speed of solution of the dG method within ASPECT. We note that the use of flux limiters holds promise in combating the over- and undershoots witnessed when discretising with the dG method.

Future work lies in comparing the performance of the implemented methods relative to other available methods. We expect the new adaptivity strategy based on the *a posteriori* error estimator presented here to result in better approximation of full mantle simulations, since, so far, it appears to give computational savings with no detriment to the observed convection patterns. Additional implementation optimisations remain possible, particularly in terms of using the block structure of the matrices, to improve the competitiveness of the dG method against FE methods on fixed grids.

This research used the ALICE High Performance Computing Facility at the University of Leicester.

Appendix A

Proof of bounds for L2 projection

Let I be the identity. The L^2-projection $\Pi : V_h + H^1_D(\Omega) \to X_{h,1}$ (the space of linear polynomials as defined by (3.2.1)–(3.2.2)) is defined as the unique $w_h = \Pi w$ such that

$$(w, v_h) = (\Pi w, v_h) \qquad \forall v_h \in X_{h,1}.$$

We begin by the following observations, under the assumption $\mathcal{L} > 0$:

$$\|v\|^2_K \leq \frac{1}{\underline{\psi}_K} \|v\|^2_{\psi,K} \leq \frac{1}{\underline{\psi}_K \underline{\mathcal{L}}_K} \left\|\sqrt{\mathcal{L}} v\right\|^2_{\psi,K} \leq \frac{1}{\underline{\psi}_K \underline{\mathcal{L}}_K} |||v|||^2_{\psi,K}, \tag{A.0.1}$$

$$\|\psi v\|^2_K \leq \frac{\overline{\psi}^2_K}{\underline{\psi}_K} \|v\|^2_{\psi,K} \leq \frac{\overline{\psi}^2_K}{\underline{\psi}_K \underline{\mathcal{L}}_K} \left\|\sqrt{\mathcal{L}} v\right\|^2_{\psi,K} \leq \frac{\overline{\psi}^2_K}{\underline{\psi}_K \underline{\mathcal{L}}_K} |||v|||^2_{\psi,K},$$

$$\|\nabla v\|^2_K \leq \frac{1}{\varepsilon \underline{\psi}_K} \varepsilon \|\nabla v\|^2_{\psi,K} \leq \frac{1}{\varepsilon \underline{\psi}_K} |||v|||^2_{\psi,K},$$

$$\begin{aligned}
\|\nabla(\psi v)\|^2_K &\leq \|v \nabla \psi\|^2_K + \|\psi \nabla v\|^2_K \\
&\leq \overline{\nabla \psi}^2_K \|v\|^2_K + \overline{\psi}^2_K \|\nabla v\|^2_K \leq \frac{\overline{\nabla \psi}_{K^2}}{\underline{\psi}_K \underline{\mathcal{L}}_K} |||v|||^2_{\psi,K} + \frac{\overline{\psi}^2_K}{\varepsilon \underline{\psi}_K} |||v|||^2_{\psi,K} \\
&\leq \frac{1}{\underline{\psi}_K} \left(\frac{\overline{\nabla \psi}^2_K}{\underline{\mathcal{L}}_K} + \frac{\overline{\psi}^2_K}{\varepsilon} \right) |||v|||^2_{\psi,K} \leq \frac{2}{\underline{\psi}_K} \max \left\{ \frac{\overline{\nabla \psi}^2_K}{\underline{\mathcal{L}}_K}, \frac{\overline{\psi}^2_K}{\varepsilon} \right\} |||v|||^2_{\psi,K}.
\end{aligned} \tag{A.0.2}$$

We note that, in the case where $\mathcal{L} \leq 0$, we must have $\eta = 0$ and, thus, $\psi = 1$. In this case, (A.0.2) collapses to the statement

$$\|\nabla(\psi v)\|_K^2 = \|\nabla v\|_K^2 \leq \frac{1}{\varepsilon}|||v|||_{\psi,K}^2.$$

Then for any $v \in H_D^1(\Omega)$, we have

$$\|(I - \Pi)(\psi v)\|_K \leq \|\psi v\|_K + \|\Pi(\psi v)\|_K \lesssim \|\psi v\|_K,$$

by the triangle inequality, the stability of Π, and (A.0.1).

On the other hand, by [85, (3.5.22)],

$$\|(I - \Pi)(\psi v)\|_K \lesssim h_K \|\nabla(\psi v)\|_K \leq \sqrt{2}\frac{h_K}{\sqrt{\underline{\psi}_K}} \max\left\{\frac{|\overline{\nabla\psi}_K|}{\sqrt{\underline{\mathcal{L}}_K}}, \frac{\overline{\psi}_K}{\sqrt{\varepsilon}}\right\} |||v|||_{\psi,K}.$$

or, in the case of $\mathcal{L} \leq 0$,

$$\|(I - \Pi)(\psi v)\|_K = \|(I - \Pi)v\|_K \leq h_K \|\nabla v\|_K \lesssim \frac{h_K}{\sqrt{\varepsilon}}|||v|||_{\psi,K}.$$

Thus, we are able to say

$$\rho_K^{-1} \|(I - \Pi)(\psi v)\|_K \lesssim |||v|||_{\psi,K},$$

with ρ_K defined as in (4.3.1).

We prove the edge-based bound by combining the cell-based result above with the following multiplicative trace inequality (e.g., [31, Lemma 3.1]): for every cell K, and every edge $F \subset \partial K$, and for every function $v \in H^1(K)$, we have that

$$\|v\|_F^2 \lesssim h_K^{-1}\|v\|_K^2 + \|v\|_K \|\nabla v\|_K.$$

Then, since $(I - \Pi)(\psi v) \in H^1(K)$, and additionally using the stability of the L^2-projection,

$$\begin{aligned}\|(I - \Pi)(\psi v)\|_F^2 &\lesssim h_K^{-1}\|(I - \Pi)(\psi v)\|_K^2 + \|(I - \Pi)(\psi v)\|_K \|\nabla(I - \Pi)(\psi v)\|_K \\ &\lesssim h_K^{-1}\|(I - \Pi)(\psi v)\|_K^2 + \|(I - \Pi)(\psi v)\|_K \|\nabla(\psi v)\|_K\end{aligned}$$

$$\lesssim h_K^{-1}\rho_K^2|||v|||_{\psi,K}^2 + \frac{\rho_K}{\sqrt{\underline{\psi}_K}}\max\left\{\frac{\overline{\nabla\psi}_K}{\sqrt{\underline{\mathcal{L}}_K}}, \frac{\overline{\psi}_K}{\sqrt{\varepsilon}}\right\}|||v|||_{\psi,K}|||v|||_{\psi,K}$$

$$\lesssim \frac{\rho_K}{\sqrt{\underline{\psi}_K}}\max\left\{\frac{\overline{\nabla\psi}_K}{\sqrt{\underline{\mathcal{L}}_K}}, \frac{\overline{\psi}_K}{\sqrt{\varepsilon}}\right\}|||v|||_{\psi,K}^2,$$

or, as above, in the case of $\mathcal{L} \leq 0$,

$$\|(I-\Pi)(\psi v)\|_F^2 \lesssim \frac{\rho_K}{\sqrt{\varepsilon}}|||v|||_{\psi,K}^2.$$

Then,

$$\rho_{\omega_F}^{-1}\|(I-\Pi)(\psi v)\|_K^2 \lesssim |||v|||_{\psi,K}^2,$$

with ρ_{ω_F} defined as in (4.3.1). The results (4.4.2) and (4.4.3) follow.

Bibliography

[1] Adams, R. A., and Fournier, J. J. F. *Sobolev spaces*, second ed., vol. 140 of *Pure and Applied Mathematics (Amsterdam)*. Elsevier/Academic Press, Amsterdam, 2003.

[2] Agmon, S. *Lectures on elliptic boundary value problems*. AMS Chelsea Publishing, Providence, RI, 2010. Prepared for publication by B. Frank Jones, Jr. with the assistance of George W. Batten, Jr., Revised edition of the 1965 original.

[3] Ainsworth, M., and Oden, J. T. A posteriori error estimation in finite element analysis. *Comput. Methods Appl. Mech. Engrg. 142*, 1-2 (1997), 1–88.

[4] Ainsworth, M., and Oden, J. T. *A posteriori error estimation in finite element analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2000.

[5] Araya, R., Behrens, E., and Rodríguez, R. An adaptive stabilized finite element scheme for the advection-reaction-diffusion equation. *Appl. Numer. Math. 54*, 3-4 (2005), 491–503.

[6] Araya, R., Poza, A. H., and Stephan, E. P. A hierarchical a posteriori error estimate for an advection-diffusion-reaction problem. *Math. Models Methods Appl. Sci. 15*, 7 (2005), 1119–1139.

[7] Arnold, D. N. An interior penalty finite element method with discontinuous elements. *SIAM J. Numer. Anal. 19*, 4 (1982), 742–760.

[8] Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, L. D. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal. 39*, 5 (2001/02), 1749–1779.

[9] Ayuso, B., and Marini, L. D. Discontinuous Galerkin methods for advection-diffusion-reaction problems. *SIAM J. Numer. Anal. 47*, 2 (2009), 1391–1420.

[10] Babuška, I., Chandra, J., and Flaherty, J. E., Eds. *Adaptive computational methods for partial differential equations*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1983.

[11] Baker, G. A. Finite element methods for elliptic equations using nonconforming elements. *Math. Comp. 31*, 137 (1977), 45–59.

[12] Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brun, P., Buschelman, K., Dalcin, L., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., McInnes, L. C., Rupp, K., Smith, B. F., Zampini, S., Zhang, H., and Zhang, H. PETSc Web page. `http://www.mcs.anl.gov/petsc`, 2016.

[13] Balay, S., Abhyankar, S., Adams, M. F., Brown, J., Brune, P., Buschelman, K., Dalcin, L., Eijkhout, V., Gropp, W. D., Kaushik, D., Knepley, M. G., McInnes, L. C., Rupp, K., Smith, B. F., Zampini, S., Zhang, H., and Zhang, H. PETSc Users Manual. Tech. Rep. ANL-95/11 - Revision 3.7, Argonne National Laboratory, 2016.

[14] Balay, S., Gropp, W. D., McInnes, L. C., and Smith, B. F. Efficient Management of Parallelism in Object Oriented Numerical Software Libraries. In *Modern Software Tools in Scientific Computing* (1997), E. Arge, A. M. Bruaset, and H. P. Langtangen, Eds., Birkhäuser Press, pp. 163–202.

[15] Bangerth, W., Hartmann, R., and Kanschat, G. deal.II—a general-purpose object-oriented finite element library. *ACM Trans. Math. Software 33*, 4 (2007), Art. 24, 27.

[16] Baumgardner, J. R., and Frederickson, P. O. Icosahedral discretization of the two-sphere. *SIAM J. Numer. Anal. 22*, 6 (1985), 1107–1115.

[17] Becker, R., Hansbo, P., and Larson, M. G. Energy norm a posteriori error estimation for discontinuous Galerkin methods. *Comput. Methods Appl. Mech. Engrg. 192*, 5-6 (2003), 723–733.

[18] Bercovici, D., Schubert, G., and Glatzmaier, G. A. Three-dimensional spherical models of convection in the Earth's mantle. *Science 244*, 4907 (1989), 950–955.

[19] Bercovier, M., and Pironneau, O. Error estimates for finite element method solution of the Stokes problem in the primitive variables. *Numer. Math. 33*, 2 (1979), 211–224.

[20] Berrone, S., and Canuto, C. Multilevel a posteriori error analysis for reaction-convection-diffusion problems. *Appl. Numer. Math. 50*, 3-4 (2004), 371–394.

[21] Boussinesq, J. *Théorie analytique de la chaleur: mise en harmonie avec la thermodynamique et avec la théorie mécanique de la lumière*, vol. 2. Gauthier-Villars, 1903.

[22] Brezzi, F., and Fortin, M. *Mixed and hybrid finite element methods*, vol. 15 of *Springer Series in Computational Mathematics*. Springer-Verlag, New York, 1991.

[23] Burstedde, C., Ghattas, O., Gurnis, M., Stadler, G., Tan, E., Tu, T., Wilcox, L. C., and Zhong, S. Scalable adaptive mantle convection simulation on petascale supercomputers. In *Proceedings of the 2008 ACM/IEEE conference on Supercomputing* (2008), IEEE Press, p. 62.

[24] Burstedde, C., Wilcox, L. C., and Ghattas, O. p4est: scalable algorithms for parallel adaptive mesh refinement on forests of octrees. *SIAM J. Sci. Comput. 33*, 3 (2011), 1103–1133.

[25] Buttari, A., Dongarra, J., Kurzak, J., Langou, J., Luszczek, P., and Tomov, S. The impact of multicore on math software. In *International Workshop on Applied Parallel Computing* (2006), Springer, pp. 1–10.

[26] Cangiani, A., Georgoulis, E. H., and Houston, P. *hp*-version discontinuous Galerkin methods on polygonal and polyhedral meshes. *Math. Models Methods Appl. Sci. 24*, 10 (2014), 2009–2041.

[27] Cangiani, A., Georgoulis, E. H., and Metcalfe, S. Adaptive discontinuous Galerkin methods for nonstationary convection-diffusion problems. *IMA J. Numer. Anal. 34*, 4 (2014), 1578–1597.

[28] Clément, P. Approximation by finite element functions using local regularization. *Rev. Française Automat. Informat. Recherche Opérationnelle Sér. RAIRO Anal. Numér. 9*, R-2 (1975), 77–84.

[29] Corrieu, V., Ricard, Y., and Froidevaux, C. Converting mantle tomography into mass anomalies to predict the earth's radial viscosity. *Phys. Earth Planet. Inter. 84*, 1-4 (1994), 3–13.

[30] Davies, D. R., Wilson, C. R., and Kramer, S. C. Fluidity: A fully unstructured anisotropic adaptive mesh computational modeling framework for geodynamics. *Geochem. Geophys. Geosyst. 12*, 6 (2011).

[31] Dolejší, V., Feistauer, M., and Schwab, C. A finite volume discontinuous Galerkin scheme for nonlinear convection-diffusion problems. *Calcolo 39*, 1 (2002), 1–40.

[32] Douglas, Jr., J., and Dupont, T. Interior penalty procedures for elliptic and parabolic Galerkin methods. In *Computing methods in applied sciences (Second Internat. Sympos., Versailles, 1975)*. Springer, Berlin, 1976, pp. 207–216. Lecture Notes in Phys., Vol. 58.

[33] Dziewonski, A. M., and Anderson, D. L. Preliminary reference earth model. *Phys. Earth Planet. Inter. 25*, 4 (1981), 297–356.

[34] Ern, A., and Proft, J. A posteriori discontinuous Galerkin error estimates for transient convection-diffusion equations. *Appl. Math. Lett. 18*, 7 (2005), 833–841.

[35] Ern, A., Stephansen, A. F., and Vohralík, M. Guaranteed and robust discontinuous Galerkin a posteriori error estimates for convection-diffusion-reaction problems. *J. Comput. Appl. Math. 234*, 1 (2010), 114–130.

[36] Ern, A., and Vohralík, M. A posteriori error estimation based on potential and flux reconstruction for the heat equation. *SIAM J. Numer. Anal. 48*, 1 (2010), 198–223.

[37] Evans, L. C. *Partial differential equations*, vol. 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, 1998.

[38] Fowler, A. *Mathematical geoscience*, vol. 36 of *Interdisciplinary Applied Mathematics*. Springer, London, 2011.

[39] Fowler, C. *The Solid Earth: An Introduction to Global Geophysics*. Cambridge University Press, 2005.

[40] Georgoulis, E. H., Hall, E., and Houston, P. Discontinuous Galerkin methods for advection-diffusion-reaction problems on anisotropically refined meshes. *SIAM J. Sci. Comput. 30*, 1 (2007/08), 246–271.

[41] Georgoulis, E. H., Hall, E., and Makridakis, C. An a posteriori error bound for discontinuous Galerkin approximations of convection-diffusion problems, Submitted for publication.

[42] Georgoulis, E. H., Lakkis, O., and Virtanen, J. M. A posteriori error control for discontinuous Galerkin methods for parabolic problems. *SIAM J. Numer. Anal. 49*, 2 (2011), 427–458.

[43] Gerya, T. V., and Yuen, D. A. Characteristics-based marker-in-cell method with conservative finite-differences schemes for modeling geological flows with strongly variable transport properties. *Phys. Earth Planet. Inter. 140*, 4 (2003), 293–318.

[44] Gilbarg, D., and Trudinger, N. S. *Elliptic partial differential equations of second order*. Classics in Mathematics. Springer-Verlag, Berlin, 2001. Reprint of the 1998 edition.

[45] Girault, V., and Raviart, P.-A. *Finite element methods for Navier-Stokes equations*, vol. 5 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1986. Theory and algorithms.

[46] Golub, G. H., and Van Loan, C. F. *Matrix computations*, third ed. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, 1996.

[47] Guermond, J.-L., Pasquetti, R., and Popov, B. Entropy viscosity method for nonlinear conservation laws. *J. Comput. Phys. 230*, 11 (2011), 4248–4267.

[48] Hansen, U., and Ebel, A. Experiments with a numerical model related to mantle convection: boundary layer behaviour of small-and large scale flows. *Phys. Earth Planet. Inter. 36*, 3 (1984), 374–390.

[49] Haskell, N. The motion of a viscous fluid under a surface load. *J. Appl. Phys. 6*, 8 (1935), 265–269.

[50] Haskell, N. The motion of a viscous fluid under a surface load. part II. *J. Appl. Phys. 7*, 2 (1936), 56–61.

[51] Haskell, N. A. The viscosity of the asthenosphere. *Am. J. Sci.*, 193 (1937), 22–28.

[52] He, Y., Puckett, E. G., and Billen, M. I. A discontinuous Galerkin method with a bound preserving limiter for the advection of non-diffusive fields in solid Earth geodynamics. *Phys. Earth Planet. Inter. 263* (2017), 23–37.

[53] Heroux, M., Bartlett, R., Hoekstra, V. H. R., Hu, J., Kolda, T., Lehoucq, R., Long, K., Pawlowski, R., Phipps, E., Salinger, A., Thornquist, H., Tuminaro, R., Willenbring, J., and Williams, A. An Overview of Trilinos. Tech. Rep. SAND2003-2927, Sandia National Laboratories, 2003.

[54] Heroux, M. A., Bartlett, R. A., Howle, V. E., Hoekstra, R. J., Hu, J. J., Kolda, T. G., Lehoucq, R. B., Long, K. R., Pawlowski, R. P., Phipps, E. T., Salinger, A. G., Thornquist, H. K., Tuminaro, R. S., Willenbring, J. M., Williams, A., and Stanley, K. S. An overview of the Trilinos project. *ACM Trans. Math. Software 31*, 3 (2005), 397–423.

[55] Houston, P., Perugia, I., and Schötzau, D. Mixed discontinuous Galerkin approximation of the Maxwell operator. *SIAM J. Numer. Anal. 42*, 1 (2004), 434–459 (electronic).

[56] Houston, P., Schötzau, D., and Wihler, T. P. Energy norm a posteriori error estimation of *hp*-adaptive discontinuous Galerkin methods for elliptic problems. *Math. Models Methods Appl. Sci. 17*, 1 (2007), 33–62.

[57] Houston, P., and Süli, E. Adaptive Lagrange-Galerkin methods for unsteady convection-diffusion problems. *Math. Comp. 70*, 233 (2001), 77–106.

[58] Hughes, T. J., Brooks, A., et al. A theoretical framework for petrov-galerkin methods with discontinuous weighting functions: Application to the streamline-upwind procedure. *Finite elements in fluids 4*, 2 (1982), 47.

[59] Hughes, T. J. R., and Brooks, A. A multidimensional upwind scheme with no crosswind diffusion. In *Finite element methods for convection dominated flows (Papers, Winter Ann. Meeting Amer. Soc. Mech. Engrs., New York, 1979)*, vol. 34 of *AMD*. Amer. Soc. Mech. Engrs. (ASME), New York, 1979, pp. 19–35.

[60] Karakashian, O. A., and Pascal, F. A posteriori error estimates for a discontinuous Galerkin approximation of second-order elliptic problems. *SIAM J. Numer. Anal. 41*, 6 (2003), 2374–2399 (electronic).

[61] Karakashian, O. A., and Pascal, F. Adaptive discontinuous Galerkin approximations of second-order elliptic problems. *ECCOMAS 2004 - European Congress on Computational Methods in Applied Sciences and Engineering* (2004).

[62] Kaufmann, G., and Lambeck, K. Glacial isostatic adjustment and the radial viscosity profile from inverse modeling. *J. Geophys. Res. Solid Earth 107*, B11 (2002).

[63] Keken, P. v., King, S., Schmeling, H., Christensen, U., Neumeister, D., and Doin, M.-P. A comparison of methods for the modeling of thermochemical convection. *J. Geophys. Res. Solid Earth 102*, B10 (1997), 22477–22495.

[64] Kelly, D. W., Gago, J. P. d. S. R., Zienkiewicz, O. C., and Babuška, I. A posteriori error analysis and adaptive processes in the finite element method. I. Error analysis. *Internat. J. Numer. Methods Engrg. 19*, 11 (1983), 1593–1619.

[65] Kelly, D. W., Nakazawa, S., Zienkiewicz, O. C., and Heinrich, J. C. A note on upwinding and anisotropic balancing dissipation in finite element approximations to convective diffusion problems. *Internat. J. Numer. Methods Engrg. 15*, 11 (1980), 1705–1711.

[66] Kronbichler, M., Bangerth, W., and Heister, T. *The deal.II Library: The step-32 tutorial program*, 2015 (accessed September 24, 2016). `https://www.dealii.org/developer/doxygen/deal.II/step_32.html`.

[67] Kronbichler, M., Heister, T., and Bangerth, W. High accuracy mantle convection simulation through modern numerical methods. *Geophys. J. Int. 191*, 1 (2012), 12–29.

[68] Kunert, G. A posteriori error estimation for convection dominated problems on anisotropic meshes. *Math. Methods Appl. Sci. 26*, 7 (2003), 589–617.

[69] Leng, W., and Zhong, S. Viscous heating, adiabatic heating and energetic consistency in compressible mantle convection. *Geophys. J. Int. 173*, 2 (2008), 693–702.

[70] Lowrie, W. *Fundamentals of Geophysics*. Cambridge University Press, 2007.

[71] Lozinski, A., Picasso, M., and Prachittham, V. An anisotropic error estimator for the Crank-Nicolson method: application to a parabolic problem. *SIAM J. Sci. Comput. 31*, 4 (2009), 2757–2783.

[72] Makridakis, C., and Nochetto, R. H. Elliptic reconstruction and a posteriori error estimates for parabolic problems. *SIAM J. Numer. Anal. 41*, 4 (2003), 1585–1594.

[73] May, D., Schellart, W., and Moresi, L. Overview of adaptive finite element analysis in computational geodynamics. *Journal of Geodynamics 70* (2013), 1–20.

[74] McKenzie, D. P., Roberts, J. M., and Weiss, N. O. Convection in the earth's mantle: towards a numerical simulation. *J. Fluid Mech. 62*, 03 (1974), 465–538.

[75] Métivier, L., Caron, L., Greff-Lefftz, M., Pajot-Métivier, G., Fleitout, L., and Rouby, H. Evidence for postglacial signatures in gravity gradients: A clue in lower mantle viscosity. *Earth Planet. Sci. Lett. 452* (2016), 146–156.

[76] Mihaljan, J. M. A rigorous exposition of the Boussinesq approximations applicable to a thin layer of fluid. *Astrophys. J. 136* (1962), 1126–1133.

[77] Minear, J. W., and Toksöz, M. N. Thermal regime of a downgoing slab and new global tectonics. *J. Geophys. Res. 75*, 8 (1970), 1397–1419.

[78] Mitrovica, J., and Forte, A. A new inference of mantle viscosity based upon joint inversion of convection and glacial isostatic adjustment data. *Earth Planet. Sci. Lett. 225*, 1 (2004), 177–189.

[79] Mitrovica, J. X. Haskell [1935] revisited. *J. Geophys. Res. Solid Earth 101*, B1 (1996), 555–569.

[80] Moresi, L.-N., and Solomatov, V. Numerical investigation of 2d convection with extremely large viscosity variations. *Phys. Fluids 7*, 9 (1995), 2154–2162.

[81] Morton, G. M. *A computer oriented geodetic data base and a new technique in file sequencing*. International Business Machines Company New York, 1966.

[82] Nakagawa, T., and Tackley, P. J. Effects of a perovskite-post perovskite phase change near core-mantle boundary in compressible mantle convection. *Geophys. Res. Lett. 31*, 16 (2004).

[83] O'Neill, C., Moresi, L., Müller, D., Albert, R., and Dufour, F. Ellipsis 3d: A particle-in-cell finite-element hybrid code for modelling mantle convection and lithospheric deformation. *Computers & Geosciences 32*, 10 (2006), 1769–1779.

[84] Paulson, A., Zhong, S., and Wahr, J. Limitations on the inversion for mantle viscosity from postglacial rebound. *Geophys. J. Int. 168*, 3 (2007), 1195–1209.

[85] Quarteroni, A., and Valli, A. *Numerical approximation of partial differential equations*, vol. 23 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1994.

[86] Rhebergen, S., Wells, G. N., Katz, R. F., and Wathen, A. J. Analysis of block preconditioners for models of coupled magma/mantle dynamics. *SIAM J. Sci. Comput. 36*, 4 (2014), A1960–A1977.

[87] Rhebergen, S., Wells, G. N., Wathen, A. J., and Katz, R. F. Three-field block preconditioners for models of coupled magma/mantle dynamics. *SIAM J. Sci. Comput. 37*, 5 (2015), A2270–A2294.

[88] Sangalli, G. Robust a-posteriori estimator for advection-diffusion-reaction problems. *Math. Comp. 77*, 261 (2008), 41–70 (electronic).

[89] Schmeling, H., and Jacoby, W. On modeling the lithosphere in mantle convection with non-linear rheology. *Journal of Geophysics-Zeitschrift Fur Geophysik 50*, 2 (1981), 89–100.

[90] Schötzau, D., and Zhu, L. A robust a-posteriori error estimator for discontinuous Galerkin methods for convection-diffusion equations. *Appl. Numer. Math. 59*, 9 (2009), 2236–2255.

[91] Schubert, G., Turcotte, D. L., and Olson, P. *Mantle convection in the Earth and planets*. Cambridge University Press, 2001.

[92] Schwab, C. *p- and hp-finite element methods*. Numerical Mathematics and Scientific Computation. The Clarendon Press, Oxford University Press, New York, 1998. Theory and applications in solid and fluid mechanics.

[93] Šebestová, I. Two-sided a posteriori error estimates for the DGMs for the heat equation. In *Numerical mathematics and advanced applications 2011*. Springer, Heidelberg, 2013, pp. 379–387.

[94] Spiegel, E., and Veronis, G. On the Boussinesq approximation for a compressible fluid. *Astrophys. J. 131* (1960), 442.

[95] Sun, S., and Wheeler, M. F. A posteriori error estimation and dynamic adaptivity for symmetric discontinuous Galerkin approximations of reactive transport problems. *Comput. Methods Appl. Mech. Engrg. 195*, 7-8 (2006), 632–652.

[96] Tabata, M., and Suzuki, A. A stabilized finite element method for the Rayleigh-Bénard equations with infinite Prandtl number in a spherical shell. *Comput. Methods Appl. Mech. Engrg. 190*, 3-4 (2000), 387–402. Fourth Japan-US Symposium on Finite Element Methods in Lage-scale Computational Fluid Dynamics (Tokyo, 1998).

[97] Tabata, M., and Suzuki, A. Mathematical modeling and numerical simulation of Earth's mantle convection. In *Mathematical modeling and numerical simulation in continuum mechanics (Yamaguchi, 2000)*, vol. 19 of *Lect. Notes Comput. Sci. Eng.* Springer, Berlin, 2002, pp. 219–231.

[98] Tackley, P. J. Effects of strongly variable viscosity on three-dimensional compressible convection in planetary mantles. *J. Geophys. Res. Solid Earth 101*, B2 (1996), 3311–3332.

[99] Tackley, P. J. Modelling compressible mantle convection with large viscosity contrasts in a three-dimensional spherical shell using the yin-yang grid. *Phys. Earth Planet. Inter. 171*, 1 (2008), 7–18.

[100] Tackley, P. J., Stevenson, D. J., Glatzmaier, G. A., and Schubert, G. Effects of an endothermic phase transition at 670 km depth in a spherical model of convection in the earth's mantle. *Nature 361*, 6414 (1993), 699–704.

[101] Temam, R. *Navier-Stokes equations. Theory and numerical analysis*. North-Holland Publishing Co., Amsterdam-New York-Oxford, 1977. Studies in Mathematics and its Applications, Vol. 2.

[102] Torrance, K., and Turcotte, D. Thermal convection with large viscosity variations. *J. Fluid Mech. 47*, 01 (1971), 113–125.

[103] Turcksin, B., Kronbichler, M., and Bangerth, W. WorkStream – A Design Pattern for Multicore-Enabled Finite Element Computations. *ACM Trans. Math. Softw. 43*, 1 (Aug. 2016), 2:1–2:29.

[104] Verfürth, R. Error estimates for a mixed finite element approximation of the Stokes equations. *RAIRO Anal. Numér. 18*, 2 (1984), 175–182.

[105] Verfürth, R. A posteriori error estimation and adaptive mesh-refinement techniques. *J. Comput. Appl. Math. 50*, 1-3 (1994), 67–83.

[106] Verfürth, R. A posteriori error estimators for convection-diffusion equations. *Numer. Math. 80*, 4 (1998), 641–663.

[107] Verfürth, R. Robust a posteriori error estimates for nonstationary convection-diffusion equations. *SIAM J. Numer. Anal. 43*, 4 (2005), 1783–1802 (electronic).

[108] Verfürth, R. Robust a posteriori error estimates for stationary convection-diffusion equations. *SIAM J. Numer. Anal. 43*, 4 (2005), 1766–1782 (electronic).

[109] Von Neumann, J., and Richtmyer, R. D. A method for the numerical calculation of hydrodynamic shocks. *J. Appl. Phys. 21* (1950), 232–237.

[110] Warburton, T., and Hesthaven, J. S. On the constants in *hp*-finite element trace inverse inequalities. *Comput. Methods Appl. Mech. Engrg. 192*, 25 (2003), 2765–2773.

[111] Wheeler, M. F. An elliptic collocation-finite element method with interior penalties. *SIAM J. Numer. Anal. 15*, 1 (1978), 152–161.

[112] Young, R. E. Finite-amplitude thermal convection in a spherical shell. *J. Fluid Mech. 63*, 04 (1974), 695–721.

[113] Yuen, D. A., Quareni, F., and Hong, H.-J. Effects from equation of state and rheology in dissipative heating in compressible mantle convection. *Nature 326* (1987), 67–69.

[114] Zhang, S., and Yuen, D. A. Various influences on plumes and dynamics in time-dependent, compressible mantle convection in 3-D spherical shell. *Phys. Earth Planet. Inter. 94*, 3 (1996), 241 – 267.

[115] Zhang, X., and Shu, C.-W. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *J. Comput. Phys. 229*, 23 (2010), 8918–8934.

[116] Zhang, Y., Zhang, X., and Shu, C.-W. Maximum-principle-satisfying second order discontinuous Galerkin schemes for convection-diffusion equations on triangular meshes. *J. Comput. Phys. 234* (2013), 295–316.

[117] Zhong, S., Zuber, M. T., Moresi, L., and Gurnis, M. Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection. *J. Geophys. Res. Solid Earth 105*, B5 (2000), 11063–11082.

[118] Zhu, L., and Schötzau, D. A robust *a posteriori* error estimate for *hp*-adaptive DG methods for convection-diffusion equations. *IMA J. Numer. Anal. 31*, 3 (2011), 971–1005.