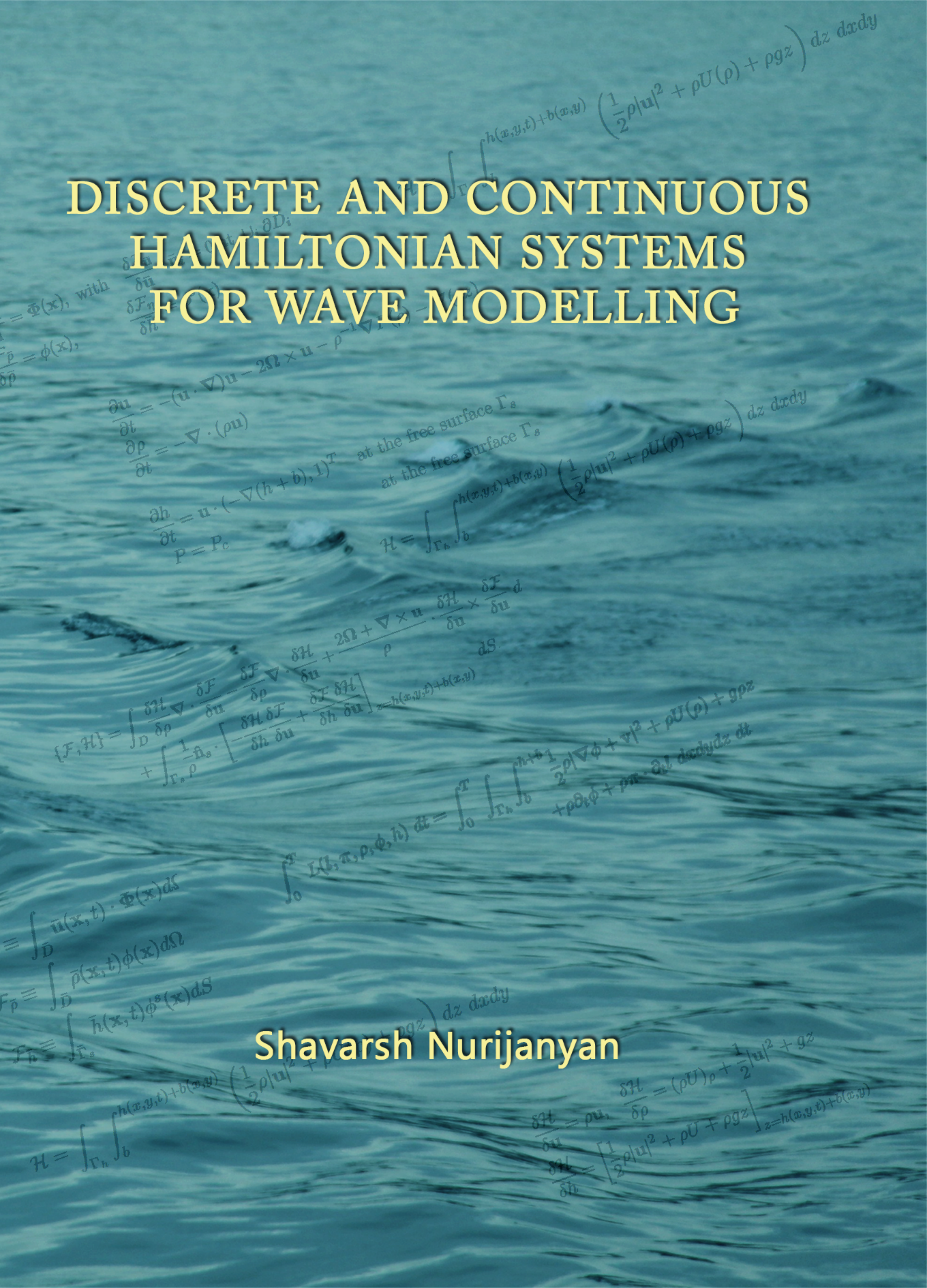


DISCRETE AND CONTINUOUS HAMILTONIAN SYSTEMS FOR WAVE MODELLING

Shavarsh Nurijanyan

DISCRETE AND CONTINUOUS HAMILTONIAN SYSTEMS FOR WAVE MODELLING

Shavarsh Nurijanyan

2013
University of Twente

Colophon
The research presented in this dissertation was carried out at the Mathematics of Computational Science (MACS) group, Department of Applied Mathematics, Faculty of Electrical Engineering, Mathematics and Computer Science of the University of Twente, The Netherlands.

This work has been part of the STW project "A numerical Wave Tank for Complex Wave and Current Interactions". The support from the Technology Foundation STW, MARIN, NIOZ, Deltares, Alkyon, TU Delft is gratefully acknowledged.

The project has been associated with the J. M. Burgers Research School for Fluid Dynamics and the Nanotechnology Research institute of the University of Twente MESA+.
This thesis was typeset in LaTeX by the author and printed by Wöhrmann Printing Service, Zutphen.

ISBN 978-90-365-1154-4

DISCRETE AND CONTINUOUS HAMILTONIAN SYSTEMS FOR WAVE MODELLING

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof.dr. H. Brinksma,
on account of the decision of the graduation committee,
to the publicly defended
on Wednesday the 9th of October 2013 at 12:45

by

Shavarsh Nurijanyan

born on 10 June 1984
in Yerevan, Armenia

This dissertation has been approved by:
prof. dr. ir. J.J.W van der Vegt
prof. dr. ir. O. Bokhove.

Contents

List of Figures

Chapter 1

Introduction

"Ta panta rhei" **Heraclitus**

1.1 Background and motivation

A brief glance at the surrounding world will reveal the abundant presence of fluids. All carbon-based life on Earth would not exist without fluids. Hence, the presence of fluids is a complete and absolute necessity for our existence. Most interestingly, not only the presence, but also the behaviour that fluids exhibit, is equally essential for the existence of life forms. The water flow circulation in the hydrological cycle supports energy exchanges among the atmosphere, ocean and land, which is part of the Earth's climate and has a considerable influence on the natural climate variability. Also, the water circulation in the ocean supports mixing processes that are essential for heat and carbon balances, efficient biological filtration, as well as providing a food delivery service to a variety of organisms, etc.

It is obvious that any knowledge on fluid behaviour will be beneficial for numerous aspects of our life. There are three main approaches to study fluid dynamics: (i) experimental, (ii) theoretical and (iii) computational. Due to their direct relevance to the thematics of this thesis, we will concentrate mainly on the last two approaches, which are one of the central themes of modern applied mathematics.

The mathematical investigation of fluid dynamics has been a challenging quest. It has resulted not only in significant theoretical and practical knowledge on a wide variety of fluid mechanics problems, but also stimulated many studies in pure and applied mathematics. Many famous mathematicians from various generations, such as Newton, Euler, Lagrange, Stokes, Kolmogorov and Leray, have their mark on this subject. Yet, there are many unanswered questions and considerable progress is still being made.

Early formal records of fluid motion studies date back to 15th century with the works of Leonardo da Vinci, but basic practical observations and understanding of fluid behaviour were noted much earlier, already from the time of the ancient Egyptians. The water flushing toilet drainage system (similar to those in modern life) in the houses of ancient Romans is evidence of good understanding of fluid motion. Moreover, the Roman bridges (aqueducts) are still considered a tremendous engineering masterpiece. These examples suggest that reasonable amount of knowledge of fluid dynamics was accumulated to enable such exceptional applications. In 1752, L. Euler presented the first system of equations for modelling an incompressible fluid flow, which subject to minor corrections is widely known as the Euler equations. The father and son Bernoulli continued the theoretical discussions on the fluid motion by introducing the famous Bernoulli equation. Later in the 1820s, a complete system of partial differential equations (PDE) was introduced by C. Navier and expanded further by G. Stokes. In fact, the system, initially proposed by Navier, should have been the basis of a molecular dynamics model. However, the laws of interaction between the molecules introduced by Navier were noticed to be inconsistent from the physical point of view for various materials and, in particular, for liquids. Only twenty years later, the same equations were re-derived by Stokes in a quite general manner by means of the theory of continua. These are the widely known Navier-Stokes equations, and they are present directly or indirectly in any mathematical model of fluid dynamics.

Any large scale fluid motion is governed by the Navier-Stokes equations coupled with appropriate boundary conditions (e.g. solid walls or dynamic and kinematic boundary conditions at a free surface). Generally, this system of equations models any compressible and incompressible viscous fluid flow. However, various simplified mathematical models can be derived from the Navier-Stokes equations by making different assumptions. For example, if the fluid is assumed to be inviscid (a very common assumption for free surface waves) the Euler equations arise. Under the assumption of the flow being irrotational and inviscid, the full Navier-Stokes equations are simplified to potential flow theory, where the velocity is represented via the gradient of a scalar field, the velocity potential. The potential flow assumption is successfully used in various free surface water wave models. Under the assumption that the water depth is much smaller than the length of the waves, the Navier-Stokes equations reduce to the shallow water equations, that are widely used in models for the wave motion near the shore and long tsunami waves in the ocean. It is apparent, that all these simplifications were made to solve, at least partially, the full system of Navier-Stokes equations. For some of these models (under various restrictions) analytical or semi-analytical solutions do exist, however, in general, they can not be solved with analytical techniques.

Until the middle of the 20th century there were mainly two approaches for studying fluid dynamics: theoretical and experimental. Since then, due to the major increase in power of modern computers, a new opportunity to study fluid dynamics is gaining popularity: computational fluid dynamics, or CFD in brief. During the past decades, this has resulted in a shift of focus from experimental and analytical techniques to numerical solutions of the equations of fluid mechanics. In engineering, CFD is now one of the major tools for fluid flow analysis. CFD is also an important tool to study fundamental flow physics, since it allows idealised ”experiments” and provides a wealth of data that are often difficult to obtain experimentally. The importance of CFD has stimulated the development of new numerical techniques that allow the robust, accurate and fast approximation of continuum models with numerical solutions (digital simulations). The general underlying idea is to replace the continuous fields in the respective system with a corresponding discrete counterpart. This process, called discretisation, allows one to obtain a numerical solution of a problem even if the (exact) solution of the original continuous equations is hard or impossible to obtain. The numerical solution will, however, only be an approximation of the analytical solution. Nevertheless, for properly designed numerical algorithms an increase in the computational resources (number of grid points, particles, etc.) will reduce the error according to the method's order of accuracy.

The main focus of this thesis is on the computation of inertial and water waves, using a particular class of numerical models. Numerical solutions for water wave models can largely be classified into the following categories

- **Discrete particle based numerical methods:** This class of numerical models is mainly based on a discrete particle representation of the fluid. The fluid and free surface motion are described through the movement of a large collection of particles. These methods are mainly used for breaking waves. The main representatives of such models are smoothed-particle hydrodynamics (SPH) [41] and the particle finite element method (PFEM) [49]. In SPH, the dynamics is defined via a Lagrangian description (the position and physical properties of the particles are described in terms of the material or referential coordinates and time). In PFEM, each particle also moves based on its mass and the external/internal forces applied to it. These Lagrangian methods are quite robust, but tend to be rather inaccurate and inefficient for real life problems, due to the large number of particles needed to obtain even a relatively small accuracy.

- **PDE-based numerical discretisations:** This class of numerical models is based on a direct discretisation of partial differential equations representing fluid flow expressed in material coordinates. The discretisation techniques may vary, starting from the simplest and historically oldest finite difference method (FDM) to the more recent finite volume method (FVM), [30], finite element method (FEM), [96], and the discontinuous (Galerkin) finite element method (DGFEM), [83, 46, 57]. For an overview on these methods, see [26]. There are also many variations of these numerical methods, such as the volume of fluids method (VOF) for multiphase flows, i.e [47], the immersed boundary method (IBM), i.e [81], and the boundary element method (BEM), i.e [50]. All mentioned methods rely on a mesh-based approximation. The main difference is the numerical approximation.

In the FDM, a grid is used and the spatial derivatives at the nodal points are approximated by finite differences. One of the most appealing aspects of this method is its simplicity, easy and fast implementation. Finite difference methods are, however, difficult to use for the discretisation of partial differential equations on domains with a complex shape.

In the FVM, the computational domain is divided into a number of elements and the integral form of the conservation laws is the starting point for the numerical discretisation. In each element the solution is assumed to be constant. The flux terms at the element boundaries need to be chosen carefully to ensure numerical stability. For higher order accuracy a reconstruction process is needed to obtain a more accurate representation of the solution using data in neighbouring elements. The main advantage of the FVM over the FDM is the geometric flexibility and the exact preservation of conservative variables.

In FEM the numerical discretisation is based on a variational formulation or more generally a Galerkin weak formulation. In classical, node based FEM the local solution in every element in an (un)structured mesh is expressed as a linear combination of locally defined basis functions. In this case, every element shares nodes with neighbouring elements, which enforces continuity of the global numerical solution. Extension to higher order approximations is relatively straightforward, unlike in FVM.

A smart combination of FVM and FEM discretisation methods results in the discontinuous Galerkin finite element method (DGFEM). In DGFEM, the domain of the problem is also tessellated using a collection of non-overlapping elements. On each element the solution is approximated by a linear combination of basis functions without enforcing continuity of the approximation with neighbouring elements. Now unlike the nodal FEM method, the set of basis functions can be easily varied from element to element. The DGFEM results in a rather weak coupling of the solution between different elements, which is beneficial for h and p-adaptation in which the mesh is, respectively, locally refined or the local polynomial order of the basis functions is adjusted.

Many PDE based mathematical models have a variational structure, which expresses important features of the problem, such as energy conservation and a Hamiltonian structure. Important physical phenomena then can be represented by an energy-related functional, the Lagrangian (e.g. [76]). This representation qualitatively differs from the PDE-representation, even though it can be reduced to it. In contrast to the PDE representation, the variational description models the dynamics of the system via specification of the systems energy behaviour. This formalism naturally embeds all conservation laws of the system and preserves the symmetry and other mathematical properties. Because of the emphasis on energy and conservation principles, a variational formulation can provide physical insight into the dynamical processes not readily apparent from the PDE formulation. Moreover, the system described via the variational formalism is coordinate independent and has a concise and exquisite structure.

An important class of variational models also has a Hamiltonian structure. The Hamiltonian formulation is a system with finite or infinite number of degrees of freedom with a specific structure. The whole dynamics is described using a phase-space and two geometric objects: the total energy functional – the Hamiltonian $\mathcal{H}$ – and a skew-symmetric Poisson bracket $\{\ ,\ \}$ [9, 76, 92]. The time evolution of a general state functional $\mathcal{F}$ is then described by the following expression

$$\frac{d\mathcal{F}}{dt} = \{\mathcal{F}, \mathcal{H}\}. \tag{1.1}$$

The (generalised) Poisson bracket $\{\ ,\ \}$ has to satisfy the following properties:

$$\begin{aligned}
\text{skew-symmetry: } & \{\mathcal{F}, \mathcal{H}\} = -\{\mathcal{H}, \mathcal{F}\}, && (1.2)\\
\text{linearity: } & \{\alpha\mathcal{F} + \beta\mathcal{G}, \mathcal{H}\} = \alpha\{\mathcal{F}, \mathcal{H}\} + \beta\{\mathcal{G}, \mathcal{H}\}, && (1.3)\\
\text{the Jacobi identity: } & \{\mathcal{F}, \{\mathcal{G}, \mathcal{H}\}\} + \{\mathcal{G}, \{\mathcal{H}, \mathcal{F}\}\} + \{\mathcal{H}, \{\mathcal{F}, \mathcal{G}\}\} = 0 && (1.4)\\
\text{the Leibniz identity: } & \{\mathcal{F}\mathcal{G}, \mathcal{H}\} = \mathcal{F}\{\mathcal{G}, \mathcal{H}\} + \{\mathcal{F}, \mathcal{H}\}\mathcal{G}, && (1.5)
\end{aligned}$$

where α and β are constants, and $\mathcal{F}$, $\mathcal{G}$ and $\mathcal{H}$ arbitrary functionals. The skew-symmetry of the bracket automatically results in energy conservation: $d\mathcal{H}/dt = \{\mathcal{H}, \mathcal{H}\}$=0.

This elegant geometrical formalism is not only succinct and has a unified structure, but it also embeds the theory of symmetries. In addition, the Hamiltonian geometrical structure has a direct connection to conservation laws, which makes it, in principle, easier to construct approximations that conserve discrete analogues of the exact constants of motion. Moreover, it can identify new conservation laws (somewhat artificial), which arise from various discretisations. In other words, Hamilton's and Lagrange's mechanics are based on quantities such as energy and action, which from a theoretical point of view are arguably more fundamental than the usual quantities such as position and velocity. It is, essentially, a more holistic approach to represent the dynamics: its specifies the energy behaviour, for any arbitrary system, and the whole of its future motion is determined from there.

From latter follows, that discretisations based on variational or Hamiltonian formalisms may inherently possess some or even all properties of the underlying formalism, but it can be highly nontrivial to develop numerical discretisations that preserve this structure.

1.2 Mathematical modelling of water waves

Generally speaking, wave motion is energy transportation from one point to another without (or almost without) transferring the mass of the continuum oscillations, which are recognised as waves. There is a whole variety of oscillations and waves occurring in different media and at all scales: from elementary particles to gravitational waves in general relativity theory.

In general, while attempting to model the propagation and transformation of water waves, one has to take care that the factual absence of dissipation is also reflected in the numerical discretisation for it to be of practical value. As a consequence, numerical models for the various internal and water wave problems discussed in this thesis are based mainly on a Hamiltonian representation. This not only avoids numerical dissipation and spurious modes, but also preserves conservation laws associated with physical phenomena.

The first mathematical model for fluid dynamics in a variational framework was formulated by Clebsch in 1859, [22]. A set of PDEs for an inviscid, incompressible fluid was derived from the variational formalism. Additional results were obtained by Hargreaves [45], and extended by Bateman [11] and later Luke [60] to include appropriate boundary conditions for incorporating a free surface. After those pioneering studies the variational formalism has gained popularity and a large series of works have been published [31, 108, 95, 13, 90].

The Hamiltonian formalism in fluid dynamics was introduced to the scientific community with works of Arnold [9] on canonical Hamiltonians, and Zakharov [110], Broer [20] and by Miles [75], on the Hamiltonian dynamical equivalent of Luke's variational model for free surface water waves. The Hamiltonian formalism for water waves in irrotational, incompressible flow is discussed in [25, 74, 75, 110], where the velocity potential and free surface elevation are canonically conjugated. Hamiltonian systems in canonical form for incompressible flows with the vorticity are discussed in [1, 23, 56, 63].

The utilisation of a continuous Hamiltonian formulation does not automatically result in a discretisation with zero numerical dissipation and the preservation of conservation laws for a given problem in fluid dynamics. A specific discretisation has to be applied to preserve the unique properties of the continuous model at the discrete level. In this work we have chosen a DGFEM method for a symmetry and structure preserving discretisation of given a Hamiltonian formulation. The method described in this thesis adds a novel numerical model to the limited number of applications in fluid dynamics that use a variational or Hamiltonian formulation in combination with a (DG)FEM discretisation [5, 109]. The motivation and advantages of this choice are discussed in corresponding chapters.

1.2.1 Waves in the interior of fluids

Waves in fluids owe their existence to restoring forces, acting on a perturbation of a background equilibrium state. This initial perturbation is driven back towards its position, overshoots since it has a finite velocity when reaches it, and then the sign of the forces reverse. This process is repeated, resulting in oscillation. The type of wave depends on the origin of the restoring force. For example, surface water waves are generated due to the gravity driven restoring force and the surface tension (small capillary waves at the surface); sound waves owe their existence to the pressure gradient; Poincaré waves and Rossby waves occur mainly due to Coriolis forces. The relevance of a particular type of the wave depends on the scale and specific setting one is interested in.

Here, we are interested in a particular class of water waves that propagate through the fluid, called internal waves, which have their maximum amplitude in the interior of the fluid. Two main representatives are internal gravity waves and inertial 'gyroscopic' waves. The origin of these waves is different. Internal gravity waves occur in both density-stratified and rotating fluids, as well as in the combined case, but the main restoring force is gravity. Unlike internal gravity waves, inertial 'gyroscopic' waves owe their existence to Coriolis forces only.

Inertial ‘gyroscopic’ waves are not widely known and thus not extensively studied. A more detailed discussion of these waves and their unique behaviour (chaotic wave attractors) in particular setups, can be found in Chapter 2 (numerical solution) and Chapter 3 (semi-analytical solution).

1.2.2 Free surface waves

In contrast to internal waves, free surface waves are easy to observe experimentally, one just needs to track the motion of the free surface. These free surface waves, as seen in the oceans, seas and lakes, are (in general) generated by wind and influenced by tides. However, after reaching equilibrium the propagation of the wave (to a greater extent) is due to the Earth’s gravitational restoring force. Basically, the energy produced by the wind is travelling by means of a deformation of the fluid surface (molecules remain at the same position on average) towards the shore and the wave is breaking near the shallower regions, releasing most of the transferred energy.

Usually, free surface waves are classified into two categories: deep-water waves and shallow-water waves. This distinction between deep and shallow waves does not depend directly on the absolute water depth. It is rather determined by the ratio of the water depth to the length of the wave. The dynamics of the surface water waves changes quite drastically from deep to shallow water. Thus, the numerical models should also account for these differences. A Hamiltonian based free surface water wave model for the deep-ocean that includes vorticity is discussed in Chapter 4.

1.3 Outline of the thesis

This thesis consists of an introduction, three main chapters and an appendix. The material of the chapters is quite independent, thus the chapters can be read separately.

In Chapter 1, a brief introduction to the fluid dynamics is given. It includes a short historical review of mathematical models for fluid flows. In particular, variational and/or Hamiltonian based systems are highlighted, due to their relevance to this thesis. The presented material is compiled using following sources [107, 35, 72, 70, 68]

In Chapter 2, a novel Hamiltonian based DGFEM discretisation for linear inviscid fluid flows in a rotating domain is introduced. Initially, the discretisation is presented in a compressible setup to enable intermediate verification against an analytical solution of compressible inviscid fluid flow, thus validating the chosen approaches and discretisation technique. Next, the incompressible limit is achieved by application of the Dirac constraints theory. The resulting DGFEM discretisations in both the compressible and incompressible regimes inherit the Hamiltonian properties of the original continuous representation. Several numerical test cases are discussed, including a simulation of inertial waves in a rotating rectangular parallelepiped and allegedly chaotic wave attractors in a tilted parallelepiped. This chapter is an extended version of the material presented in [79].

In Chapter 3, a new semi-analytical solution for inertial waves in a rotating rectangular parallelepiped is presented. This semi-analytical solution is compared with another semi-analytical solution, available in [62], and its advantages and disadvantages are identified. As an alternative, a FEM based numerical solution is presented, with further qualitative comparisons with semi-analytical solutions. This chapter is based on [78].

In Chapter 4, the Hamiltonian structure of nonlinear three dimensional compressible fluid flow in a rotating domain with a free surface is determined. Even though generally the rotation of the domain (e.g Earth rotation) has a small influence on free surface waves it allows the modelling of waves subject to Coriolis forces.

In the Appendix A, the philosophy and structure of the new hpGEM2.0, C++ software framework developed for a fast and robust implementation of DGFEM discretisations, is summarised. Additionally, a small tutorial is given, based on particular example, the implementation of the Hamiltonian based DGFEM discretisation that was introduced in Chapter 2.

Finally, conclusions and recommendations for the future work are given in Chapter 5.

Chapter 2

Hamiltonian DGFEM for linear Euler equations: inertial waves

A discontinuous Galerkin finite element method (DGFEM) has been developed and tested for the linear, three-dimensional, rotating incompressible Euler equations. These equations admit complicated wave solutions, which poses numerical challenges.

These challenges concern: (i) discretisation of a divergence-free velocity field; (ii) discretisation of geostrophic boundary conditions combined with no-normal flow at solid walls; (iii) discretisation of the conserved, Hamiltonian dynamics of the inertial-waves; and, (iv) large-scale computational demands owing to the three-dimensional nature of inertial-wave dynamics and possibly its narrow zones of chaotic attraction. These issues have been resolved, for example: (i) by employing Dirac's method of constrained Hamiltonian dynamics to our DGFEM for linear, compressible flows, thus enforcing the incompressibility constraints; (ii) by enforcing no-normal flow at solid walls in a weak form and geostrophic tangential flow along the wall; and, (iii) by applying a symplectic time discretisation.

We compared our simulations with exact solutions of three-dimensional incompressible flows, in (non)rotating periodic and partly periodic cuboids (Poincaré waves). Additional verifications concerned semi-analytical eigenmode solutions in rotating cuboids with solid walls. Finally, a simulation in a tilted rotating tank, yielding more complicated wave dynamics, demonstrates the potential of our new method.

2.1 Introduction

In the geophysical context, wave motion plays a very important role in energy and angular momentum transport within the oceans and lakes, in particular in the interior of the fluid. These waves often cause mixing, and this mixing forms a very important part of the ocean circulation. Internal gravity (e.g., [100]) and 'gyroscopic' waves, further on referred to as inertial waves (e.g., [43]), are the main representatives of transverse ocean waves which have their maximum particle displacement not at the free surface, but in the interior of the fluid domain. In contrast to internal gravity waves, where density stratification is the main restoring mechanism, inertial waves exist solely due to the angular momentum stratification. Coriolis forces caused by the rotation of the Earth act as a restoring force on the wave motion. While the influence of rotation in comparison with stratification in geophysical applications is weaker, inertial waves remain of importance in several cases. Inertial waves influence the liquid outer core of the Earth ([64, 2, 3, 88]), orbiting and/or spinning spaceships and satellites carrying liquid payload ([3, 66]), relatively homogeneous parts of the ocean ([104, 8, 33]), lake hydrodynamics ([32]), and are important in some astrophysical applications ([27]). An important property of these inertial waves is that their propagation direction is determined by ratio of the wave frequency and Coriolis frequency (at twice the rotation rate), and is not altered by the reflection from the boundaries of the fluid domain. The latter results in wave focussing and defocussing phenomena in the absence of a "local reflectional symmetry", in which case the domain walls are asymmetric, i.e., neither parallel nor perpendicular to the rotation axis. Repeated reflection in which wave focusing is dominating gives in general rise to wave attractors: narrow regions onto which the wave energy converges. In a limited set of geometries these attractors were theoretically predicted ([82, 98, 99, 89, 69]) and experimentally observed ([69]), especially in quasi-2D set-ups. The purpose of this work is to provide numerical tools such that we are able to increase our understanding of inertial waves via numerical simulations of a rotating homogeneous fluid.

Inertial waves are best studied in isolation, in a homogeneous fluid, in the absence of viscosity and nonlinearity. We therefore focus on the development and testing of finite element numerical solution techniques for the linear, three-dimensional incompressible Euler equations in rotating (closed) domains, instead of focussing directly on the more complex Navier-Stokes equations.

It is useful to contrast two types of waves admitted by the linear, incompressible Euler equations: (a) inertial waves in closed rotating domains, and (b) surface-trapped waves in half-closed domains with the free surface of the liquid acting under gravity. Surface waves arise due to the restoring force of gravity at the interface between a heavier fluid (e.g., sea water) and a lighter fluid or vacuum. Linear surface waves in the absence of Coriolis forces only involve the potential-flow component, while the vortical components of the velocity or the vorticity (the three-dimensional curl of the velocity vector) are zero. In contrast, inertial waves involve nonzero vortical components of the velocity and exhibit multi-scale behaviour, especially when wave focusing occurs. These inertial-wave solutions are thus challenging to compute, either analytically or numerically. In addition, the linear three-dimensional Euler equations form a Hamiltonian system. The wave dynamics of both wave types thus concern geometric, Hamiltonian dynamics, as an initial value problem, in which invariants such as mass, energy and phase-space volume derive from this geometric structure. Furthermore, the Hamiltonian system is constrained since the total density is constant and the divergence of the velocity field is zero. Preservation of these discrete invariants in the numerical discretisation ensures numerical stability without any loss of wave amplitude due to artificial numerical damping. The compatible numerical discretisation we aim to develop for these linear incompressible Euler equations should therefore preferably inherit a discrete analog of this characteristic Hamiltonian geometric structure.

To wit, our goal is to develop and test a Hamiltonian discontinuous Galerkin finite element method (DGFEM) for inertial-wave dynamics of the linear, incompressible, three-dimensional, rotating Euler equations. The features of the inertial waves indicate that the following mathematical and numerical challenges should be met: (i) The constraint of incompressibility of the flow, or the zero divergence of the velocity, needs to be inherited by the discretisation in a weak or strong form. This is a classical issue in computational fluid dynamics, in which the pressure acts as a Lagrange multiplier to ensure time consistency of the secondary constraint of incompressibility (namely the zero divergence). The zero perturbation density acts here as primary constraint. (ii) The discretisation needs to satisfy the geostrophic balance relations along the wall together with the no-normal flow condition imposed either weakly or strongly. Rotation in combination with the no-normal flow requirement at solid walls yields geostrophic balance conditions on the tangential velocity components. It is nontrivial to satisfy these consistency boundary conditions discretely (e.g., see [6]). (iii) A discrete analog of the geometric Hamiltonian structure needs to be established to ensure conservation properties of the system. In particular, it would guarantee preservation of wave amplitude and phase space volume, such that long-time calculations remain stable and relevant over many wave periods [44]. The use of stable dissipative, time integrators would destroy the carefully preserved geometric structure of the spatial discretisation designed for Hamiltonians in classical mechanics. Hence, symplectic time integrators are required.

The need to deal with local fine scales and the presence of strong gradients led to our choice for discontinuous Galerkin finite element methods in the first place. Furthermore, DGFEM permits large gradients and *hp*-refinement. The computational linear algebra demands are handled by using PETSc [10, 91] in our versatile DGFEM software environment hpGEM [80].

The outline of the paper is as follows and concerns all four challenges. In Section 2, we review the equations of motion for the linear compressible and incompressible Euler equations and their Hamiltonian formulations. It also includes an exposition of Dirac's method of constraints for the linear compressible Euler equations, with zero perturbation density as primary constraint [16, 93]. Concerning challenges (i)–(ii), in Section 3, we derive the general Hamiltonian DGFEM for an incompressible flow from the Hamiltonian structure for a compressible flow via Dirac's theory. Concerning challenge (iii), in Section 4, we present a proper time integrator for the presented Hamiltonian dynamics and discuss some of the properties of the resulting time and space discrete numerical schemes. Numerical verifications are given in Section 5, where DGFEM simulations are compared with exact solutions of incompressible flow in a rotating triple-periodic domain and a partially closed cuboid with periodicity in one direction, and with semi-analytical series solutions for incompressible flow in closed cuboids. Additionally, numerical results on chaotic wave attractors are presented. Conclusions are drawn in Section 6.

2.2 Continuum theory for (in)compressible fluid

2.2.1 Governing equations

Compressible fluid flow in a domain D is governed by the non-linear compressible Euler equations in a rotating frame with angular velocity

$\mathbf{\Omega} = (\Omega_1, \Omega_2, \Omega_3)^T$:

$$\frac{\partial \hat{\mathbf{u}}}{\partial t} = -\mathbf{2}\Omega \times \hat{\mathbf{u}} - (\hat{\mathbf{u}} \cdot \nabla)\hat{\mathbf{u}} - \hat{\rho}^{-1}\nabla \hat{P}(\hat{\rho}), \tag{2.1a}$$

$$\frac{\partial \hat{\rho}}{\partial t} = -\nabla \cdot (\hat{\rho}\hat{\mathbf{u}}), \tag{2.1b}$$

where $\hat{\mathbf{u}} = \hat{\mathbf{u}}(x, y, z, t) = (\hat{u}, \hat{v}, \hat{w})^T$ is the three-dimensional velocity field, $\hat{\rho} = \hat{\rho}(x, y, z, t)$ a scalar density field, and $\hat{P} = \hat{P}(\hat{\rho})$ the barotropic pressure. Cartesian coordinates $\mathbf{x} = (x, y, z)$ and time t are used; the three-dimensional differential operator is given by $\nabla = (\partial/\partial x, \partial/\partial y, \partial/\partial z)^T$. The boundaries of the domain D are denoted by $\partial D = \cup_i \partial D_i$.

We linearise the compressible Euler equations (2.1) around a rest state with $\mathbf{u}_0 = \mathbf{0}$ and $\rho_0 = const.$, such that $\hat{\mathbf{u}} = \mathbf{0} + \epsilon\mathbf{u}$ and $\hat{\rho} = \rho_0 + \epsilon\rho$, where $\mathbf{u}$ and ρ are the perturbation velocity and density fields, respectively. Linearisation yields the linear compressible Euler equations in a rotating domain

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla(\frac{c_0^2}{\rho_0}\rho) - 2\mathbf{\Omega} \times \mathbf{u}, \tag{2.2a}$$

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho_0 \mathbf{u}), \tag{2.2b}$$

where $c_0 = \sqrt{\partial \hat{P}/\partial \rho|_{\rho=\rho_0}}$ is the constant, acoustic wave speed. Two types of boundary conditions will be discussed: periodic and solid-wall boundary conditions. For fixed, solid-wall boundary conditions the normal component of the velocity field at the boundaries is zero $\mathbf{u} \cdot \hat{\mathbf{n}} = 0$, with $\hat{\mathbf{n}}$ the outward normal vector at the boundary. If we multiply both sides of the momentum equations (2.2a), restricted to the domain boundary, with the normal vector $\hat{\mathbf{n}}$,

$$\frac{\partial(\mathbf{u} \cdot \hat{\mathbf{n}})}{\partial t} = -\nabla(\frac{c_0^2}{\rho_0}\rho) \cdot \hat{\mathbf{n}} - (2\mathbf{\Omega} \times \mathbf{u}) \cdot \hat{\mathbf{n}}, \tag{2.3}$$

and apply the no-normal flow condition $\mathbf{u} \cdot \hat{\mathbf{n}} = 0$, we obtain a restriction on the density gradient

$$\frac{c_0^2}{\rho_0}\nabla\rho \cdot \hat{\mathbf{n}} = -(2\mathbf{\Omega} \times \mathbf{u}) \cdot \hat{\mathbf{n}}. \tag{2.4}$$

In the absence of domain rotation, the right side of (2.4) is zero at the boundary, which indicates that the normal component of the density gradient is also zero at the boundary. In contrast, with rotation the normal component of the density gradient is balanced by the projected components of the velocity field. This balance between the density/pressure gradient force and the Coriolis force is called geostrophic balance. Implementation of the boundary condition therefore becomes more challenging due to the mandatory satisfaction of geostrophic balance.

In the limit of zero Mach number, $M_0 = V_0/c_0 \to 0$, with V_0 a reference velocity of the fluid, the linear incompressible Euler equations arise from (2.2) as

$$\frac{\partial \mathbf{u}}{\partial t} = -2\boldsymbol{\Omega} \times \mathbf{u} - \nabla P, \tag{2.5a}$$

$$\nabla \cdot \mathbf{u} = 0, \;\; \rho = 0, \tag{2.5b}$$

where P is the pressure. Note that the constraint on the perturbation density ρ ensures that the total density is constant for all time.

2.2.2 Hamiltonian framework

In the following sections we introduce the Hamiltonian framework for linear compressible and incompressible fluid flows, including the connection with the corresponding partial differential equations (PDEs).

Bracket for linearised compressible flow

Hamiltonian dynamics of compressible fluid flow, cf., [19, 77] governed by the linear equations (2.2) in $D \subset \mathbb{R}^3$ is given by

$$\frac{d\mathcal{F}}{dt} = \{\mathcal{F}, \mathcal{H}\} = \int_D \left(\frac{\delta \mathcal{H}}{\delta \rho} \nabla \cdot \frac{\delta \mathcal{F}}{\delta \mathbf{u}} - \frac{\delta \mathcal{F}}{\delta \rho} \nabla \cdot \frac{\delta \mathcal{H}}{\delta \mathbf{u}} - \frac{\mathbf{2}\Omega}{\rho_0} \times \frac{\delta \mathcal{H}}{\delta \mathbf{u}} \cdot \frac{\delta \mathcal{F}}{\delta \mathbf{u}} \right) d\mathbf{x}, \tag{2.6}$$

with Hamiltonian energy functional

$$\mathcal{H} = \mathcal{H}[\mathbf{u}, \rho] \equiv \int_D \frac{1}{2} \Big(\rho_0 |\mathbf{u}|^2 + \frac{c_0^2}{\rho_0} \rho^2 \Big) d\mathbf{x}. \tag{2.7}$$

The definition of the functional derivative is

$$\delta \mathcal{H} \equiv \lim_{\epsilon \to 0} \frac{\mathcal{H}[\mathbf{u} + \epsilon \delta \mathbf{u}, \rho + \epsilon \delta \rho] - \mathcal{H}[\mathbf{u}, \rho]}{\epsilon} = \int_D \left[\frac{\delta \mathcal{H}}{\delta \mathbf{u}} \cdot \delta \mathbf{u} + \frac{\delta \mathcal{H}}{\delta \rho} \delta \rho \right] d\mathbf{x}. \tag{2.8}$$

The functional derivatives of $\mathcal{H}$ follow from (2.7) and (2.8), and are

$$\frac{\delta \mathcal{H}}{\delta \mathbf{u}} = \rho_0 \mathbf{u}, \quad \frac{\delta \mathcal{H}}{\delta \rho} = \frac{c_0^2}{\rho_0} \rho. \tag{2.9}$$

The Poisson bracket $\{\ ,\ \}$ in (2.6) satisfies all properties: skew-symmetry is easy to spot from the structure of the bracket; the bracket is obviously bilinear, thus the linearity and Leibniz identity are automatically satisfied; and, the Jacobi identity can be checked directly, given suitable boundary conditions.

To specify Hamiltonian dynamics in the domain D one has to specify appropriate boundary conditions. Mathematical models based on PDEs usually specify boundary conditions on the relevant variables at the boundary. Similarly, in the Hamiltonian formulation boundary conditions can be imposed by choosing appropriate function spaces for the arbitrary functional $\mathcal{F}$.

As an example, we will show the equivalence between the Hamiltonian framework (2.6)–(2.7) and the PDE representation (2.2) of compressible fluid flow in a rotating domain D bounded by solid walls. The momentum and continuity equations can be obtained if the following functionals are chosen as follows

$$\mathcal{F}_{\mathbf{u}} \equiv \int_D \mathbf{u}(\mathbf{x}, t) \cdot \mathbf{\Phi}(\mathbf{x}) d\mathbf{x} \tag{2.10a}$$

$$\mathcal{F}_{\rho} \equiv \int_D \rho(\mathbf{x}, t) \phi(\mathbf{x}) d\mathbf{x}, \tag{2.10b}$$

with $\phi \in \mathcal{Q}$ and $\mathbf{\Phi} \in \mathcal{Y}$ arbitrary test functions, where

$$\mathcal{Q} = \{\phi \in L^2(D)\} \tag{2.11}$$

$$\mathcal{Y} = \{\mathbf{\Phi} \in (L^2(D))^3 \text{ and } \nabla \cdot \mathbf{\Phi} \in L^2(D) :\ \hat{\mathbf{n}} \cdot \mathbf{\Phi} = 0 \text{ at } \partial D\}, \tag{2.12}$$

and $L^2(D)$ is the space of square integrable functions on D. To incorporate slip flow boundary conditions at ∂D we restrict the space for the test functions $\mathbf{\Phi}$ at the boundary. Corresponding functional derivatives of (2.10a) and (2.10b) thus become

$$\frac{\delta \mathcal{F}_\rho}{\delta \rho} = \phi(\mathbf{x}) \text{ and } \frac{\delta \mathcal{F}_{\mathbf{u}}}{\delta \mathbf{u}} = \mathbf{\Phi}(\mathbf{x}), \text{ with } \frac{\delta \mathcal{F}_{\mathbf{u}}}{\delta \mathbf{u}} \cdot \hat{\mathbf{n}} = 0 \text{ at } \partial D. \tag{2.13}$$

Using functionals (2.10a) and (2.10b), with corresponding functional derivatives (2.13) and (2.9), in the bracket formulation (2.6) yields the momentum (2.2a) and continuity (2.2b) equations for linearised compressible flow, respectively. We also used Gauss' law combined with (2.13). The restricted test function arising from functional $\mathcal{F}_{\mathbf{u}}$ ensures the satisfaction of the boundary conditions at the PDE level.

Construction of a Dirac-bracket for linearised incompressible flow

Dirac's theory of constrained Hamiltonian systems ([28, 93, 105]) is used to derive the linearised incompressible Euler equations as the limit of the Hamiltonian structure of the linearised compressible Euler equations. Basically, Dirac's theory enforces a constant density constraint via Lagrange multipliers onto the derived compressible Hamiltonian framework [16, 19].

Due to linearisation, the constant total density constraint $\hat{\rho} = const$ transforms into the perturbation density constraint

$$\rho(\mathbf{x}) = 0. \tag{2.14}$$

It will act as a *primary* constraint, to be incorporated into the compressible Hamiltonian dynamics (2.6) via a Lagrange multiplier field. In a consistent theory, the constraint must be preserved by the evolution of the system. This leads to several possible outcomes: (i) the consistency requirement results into, modulo constraints, an equation of essentially the form $1 = 0$; (ii) it leads to an equation of the form $0 = 0$; (iii) we obtain an equation which resolves the unknown Lagrange multiplier, or (iv) it yields a *secondary* constraint. Case (i) implies inconsistent equations of motion; they do not posses any solution. Case (ii) is the desired outcome. Case (iv) introduces new *secondary* constraints, preservation of which must be checked by repeating the procedure until either we encounter case (i) or all constraints lead to case (ii). This is the main idea of Dirac's algorithm.

A Lagrange multiplier $\lambda_\rho(\mathbf{x}, t)$ is introduced to enforce the primary constraint. This constraint, or any arbitrary functional $\mathcal{F}[\rho]$ thereof, must be preserved in time. Hence, the evolution of such a functional must remain naught, i.e.,

$$\frac{d\mathcal{F}[\rho]}{dt} = 0 = \{\mathcal{F}[\rho], \mathcal{H}\} + \int_D \lambda_\rho(\mathbf{x}')\{\mathcal{F}[\rho], \rho(\mathbf{x}')\} d\mathbf{x}'. \tag{2.15}$$

From Poisson bracket (2.6), we deduce that $\{\mathcal{F}[\rho], \rho(\mathbf{x}')\} = 0$ and, therefore, the Lagrange multiplier remains undetermined. It gives, however, rise to a secondary constraint

$$0 = \{\mathcal{F}[\rho], \mathcal{H}\} = -\int_D \frac{\delta\mathcal{F}[\rho]}{\delta\rho} \nabla\cdot(\rho_0\, \mathbf{u}(\mathbf{x}))\, d\mathbf{x}. \tag{2.16}$$

Since the functional $\mathcal{F}[\rho]$ is arbitrary in (2.16), it follows that

$$\nabla\cdot\mathbf{u} = 0 \tag{2.17}$$

should hold as well. Note that $\delta\mathcal{F}[\rho]/\delta\rho$ serves as arbitrary test function and that the secondary constraint implies that the velocity is divergence-free. Next, both constraints

$$\rho(\mathbf{x}) = 0 \quad \text{and} \quad \Delta(\mathbf{x}) = \nabla\cdot\mathbf{u}(\mathbf{x}) = 0 \tag{2.18}$$

will be enforced simultaneously as primary constraints, also in time.

For this reason, we introduce Lagrange multipliers $\lambda_\rho = \lambda_\rho(\mathbf{x},t)$ and $\lambda_\Delta = \lambda_\Delta(\mathbf{x},t)$. The two consistency requirements are stated in weak form by using two (different) arbitrary functionals $\mathcal{F}[\rho]$ and $\mathcal{F}[\Delta]$, as follows

$$\frac{d\mathcal{F}[\rho]}{dt} = 0 = \{\mathcal{F}[\rho],\mathcal{H}\} + \int_D \lambda_\Delta(\mathbf{x}')\{\mathcal{F}[\rho],\Delta(\mathbf{x}')\}d\mathbf{x}', \tag{2.19a}$$

$$\begin{aligned}\frac{d\mathcal{F}[\Delta]}{dt} = 0 = \{\mathcal{F}[\Delta],\mathcal{H}\} &+ \int_D \lambda_\rho(\mathbf{x}')\{\mathcal{F}[\Delta],\rho(\mathbf{x}')\}d\mathbf{x}' \\ &+ \int_D \lambda_\Delta(\mathbf{x}')\{\mathcal{F}[\Delta],\Delta(\mathbf{x}')\}d\mathbf{x}',\end{aligned} \tag{2.19b}$$

where we omitted stating the explicit time dependence. An elaborate calculation of the brackets in (2.19a) yields

$$0 = \int_D \frac{\delta\mathcal{F}}{\delta\rho}\big(-\nabla\cdot(\rho_0\mathbf{u}) + \nabla^2\lambda_\Delta\big)d\mathbf{x} - \int_{\partial D}\frac{\delta\mathcal{F}}{\delta\rho}\hat{\mathbf{n}}\cdot\nabla\lambda_\Delta dS \tag{2.20}$$

with surface element dS. By using the secondary constraint in (2.20), and the arbitrariness of the functional $\mathcal{F}[\rho]$ in the interior and at the boundary, we find that

$$\nabla^2\lambda_\Delta = 0 \qquad \text{with} \qquad \hat{\mathbf{n}}\cdot\nabla\lambda_\Delta = 0. \tag{2.21}$$

Its solution is $\lambda_\Delta = \text{cst}$.

To analyse (2.19b), we first relate the functional derivative of $\mathcal{F}[\Delta]$ with respect to Δ to the one with respect to $\mathbf{u}$, as follows

$$\delta\mathcal{F}[\Delta] = \int_D \frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}\delta\Delta\, d\mathbf{x} = -\int_D \nabla\frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}\cdot\delta\mathbf{u}\, d\mathbf{x}, \tag{2.22}$$

where we used that $\hat{\mathbf{n}}\cdot\delta\mathbf{u} = 0$. The last term in (2.19b) cancels after an integration by parts, by using the additional boundary conditions $\hat{\mathbf{n}}\cdot\nabla\lambda_\Delta = 0$ and $\hat{\mathbf{n}}\cdot\nabla(\delta\mathcal{F}[\Delta]/\delta\Delta) = 0$ at ∂D. We subsequently find that (2.19b) becomes

$$0 = \int_D \frac{\delta\mathcal{F}}{\delta\Delta}\big(\nabla\cdot(2\boldsymbol{\Omega}\times\mathbf{u}) + \nabla^2\lambda_\rho\big)d\mathbf{x} - \int_{\partial D}\frac{\delta\mathcal{F}}{\delta\Delta}\hat{\mathbf{n}}\cdot\big(2\boldsymbol{\Omega}\times\mathbf{u} + \nabla\lambda_\rho\big)\, dS. \tag{2.23}$$

The arbitrariness of $\mathcal{F}[\Delta]$ in (2.23), in the interior and at the boundary, then implies that

$$\nabla \cdot (2\boldsymbol{\Omega} \times \mathbf{u}) + \nabla^2 \lambda_\rho = 0 \quad \text{on } D \quad \text{with} \quad \big(2\boldsymbol{\Omega} \times \mathbf{u} + \nabla \lambda_\rho\big) \cdot \hat{\mathbf{n}} = 0 \quad \text{on } \partial D. \tag{2.24}$$

Details in the above calculations have been relegated to Appendix 2.7.1.

The bracket formulation for incompressible flow is now given by

$$\frac{d\mathcal{F}[\mathbf{u}]}{dt} = \{\mathcal{F}, \mathcal{H}\} + \int_D \lambda_\rho(\mathbf{x}')\{\mathcal{F}, \rho(\mathbf{x}')\} d\mathbf{x}'. \tag{2.25}$$

The dynamics is then obtained from (2.25) combined with (2.23) for the Lagrange multiplier ($\lambda = \lambda_\rho$)

$$\frac{d\mathcal{F}}{dt} = \{\mathcal{F}, \mathcal{H}\}_{inc} \equiv \int_D \left[-\frac{2\boldsymbol{\Omega}}{\rho_0} \times \frac{\delta \mathcal{H}}{\delta \mathbf{u}(\mathbf{x})} \cdot \frac{\delta \mathcal{F}}{\delta \mathbf{u}(\mathbf{x})} + \lambda(\mathbf{x}) \nabla \cdot \frac{\delta \mathcal{F}}{\delta \mathbf{u}(\mathbf{x})} \right] d\mathbf{x}, \tag{2.26a}$$

$$0 = \int_D \frac{\delta \mathcal{F}}{\delta \Delta} \big(\nabla \cdot (2\Omega \times \mathbf{u}) + \nabla^2 \lambda\big) d\mathbf{x} - \int_{\partial D} \big(2\Omega \times \mathbf{u} + \nabla \lambda\big) \cdot \hat{\mathbf{n}} \frac{\delta \mathcal{F}}{\delta \Delta} dS, \tag{2.26b}$$

with constrained energy functional

$$\mathcal{H} = \int_D \frac{1}{2} \rho_0 |\mathbf{u}|^2 d\mathbf{x}. \tag{2.26c}$$

It is obtained after application of the primary constraint $\rho = 0$. The incompressible, linear Euler equations can be derived from (2.26) by choosing functionals

$$\mathcal{F}_{\mathbf{u}} \equiv \int_D \mathbf{u}(\mathbf{x}, t) \cdot \boldsymbol{\Phi}(\mathbf{x}) d\mathbf{x} \quad \text{and} \quad \mathcal{F}_\Delta \equiv \int_D \Delta(\mathbf{x}, t) \widetilde{\phi}(\mathbf{x}) d\mathbf{x}, \tag{2.27}$$

where $\boldsymbol{\Phi}(\mathbf{x}) \in \mathcal{Y}$ and $\widetilde{\phi}(\mathbf{x}) \in \mathcal{Q}$ with the additional requirement that $\hat{\mathbf{n}} \cdot \nabla \widetilde{\phi} = 0$. The functionals in (2.27) lead to the system of equations

$$\frac{\partial \mathbf{u}}{\partial t} = -\nabla \lambda - 2\boldsymbol{\Omega} \times \mathbf{u} \quad \text{and} \quad \nabla^2 \lambda = -\nabla \cdot (2\boldsymbol{\Omega} \times \mathbf{u}), \tag{2.28}$$

with slip flow $\mathbf{u} \cdot \hat{\mathbf{n}} = 0$ and geostrophic balance $\big(2\Omega \times \mathbf{u} + \nabla \lambda\big) \cdot \hat{\mathbf{n}} = 0$ at the solid-wall boundary. Notice that the Lagrange multiplier $\lambda = P$ plays the role of the pressure P.

2.3 Discrete Hamiltonian formulation

Discretisations of the earlier derived compressible and incompressible continuous Hamiltonian formulations will be derived next. There are two possible choices for a derivation of discrete Hamiltonian dynamics for incompressible fluid flow: direct discretisation of the continuous bracket formulation (2.26) for incompressible fluid flow, or application of Dirac's theory on the discretised Hamiltonian formulation of compressible flow. The latter approach is preferable for several reasons: (i) a discretisation of the compressible Hamiltonian formulation is becoming an intermediate check point for the introduced discretisation algorithm; (ii) avoidance of dealing with discontinuities of unknown Lagrange multipliers simplifies the process; and, (iii) the relatively easy incorporation of boundary conditions which are set automatically by Dirac's theory given the proper boundary conditions for the compressible case. Before proceeding to a discontinuous Galerkin FEM discretisation, we demonstrate key aspects of the algorithm on a finite volume (FV) discretisation of the compressible Hamiltonian formulation with consecutive application of Dirac's theory on a discrete level. This FV discretisation is equivalent to a DG discretisation with constant basis functions.

2.3.1 Finite volume discretisation for linear Euler equations

Discrete compressible dynamics

The three-dimensional linear compressible Euler equations (2.2) are considered in a periodic rectangular parallelepiped, where an equidistant mesh is introduced. The equations are scaled for simplicity such that we effectively can take $\rho_0 = c_0 = 1$ in (2.2) (hereafter). A tessellation of this triple periodic domain results in a collection of elements K with (i, j, k) index numbering. A FV discretisation for the scaled version of compressible Euler equations (2.2), with a chosen "antisymmetric θ scheme" for the spatial derivatives, yields the following discrete equations

$$\frac{d}{dt}\begin{pmatrix} \bar{U}_{i,j,k} \\ \bar{V}_{i,j,k} \\ \bar{W}_{i,j,k} \\ \bar{R}_{i,j,k} \end{pmatrix} = -\begin{pmatrix} 2\boldsymbol{\Omega} \times \begin{pmatrix} \bar{U}_{i,j,k} \\ \bar{V}_{i,j,k} \\ \bar{W}_{i,j,k} \end{pmatrix} \\ 0 \end{pmatrix} - \begin{pmatrix} \bar{G}^1_{i,j,k} \\ \bar{G}^2_{i,j,k} \\ \bar{G}^3_{i,j,k} \\ \bar{G}^4_{i,j,k} \end{pmatrix}, \tag{2.29a}$$

where $\bar{U}_{i,j,k} = \bar{U}_{i,j,k}(t)$, $\bar{V}_{i,j,k} = \bar{V}_{i,j,k}(t)$, $\bar{W}_{i,j,k} = \bar{W}_{i,j,k}(t)$ and $\bar{R}_{i,j,k} = \bar{R}_{i,j,k}(t)$ are the time-dependent mean values of u, v, w, ρ in the (i, j, k)-th element and the flux functions are defined by

$$\bar{G}^1_{i,j,k} = -\frac{(\bar{R}_{i+1,j,k}(1-\theta) + \bar{R}_{i,j,k}\theta) - (\bar{R}_{i,j,k}(1-\theta) + \bar{R}_{i-1,j,k}\theta)}{\Delta x}, \quad (2.29b)$$

$$\bar{G}^2_{i,j,k} = -\frac{(\bar{R}_{i,j+1,k}(1-\theta) + \bar{R}_{i,j,k}\theta) - (\bar{R}_{i,j,k}(1-\theta) + \bar{R}_{i,j-1,k}\theta)}{\Delta y}, \quad (2.29c)$$

$$\bar{G}^3_{i,j,k} = -\frac{(\bar{R}_{i,j,k+1}(1-\theta) + \bar{R}_{i,j,k}\theta) - (\bar{R}_{i,j,k}(1-\theta) + \bar{R}_{i,j,k-1}\theta)}{\Delta z}, \quad (2.29d)$$

$$\begin{aligned}\bar{G}^4_{i,j,k} = &-\frac{(\bar{U}_{i,j,k}(1-\theta) + \bar{U}_{i+1,j,k}\theta) - (\bar{U}_{i-1,j,k}(1-\theta) + \bar{U}_{i,j,k}\theta)}{\Delta x} \\ &-\frac{(\bar{V}_{i,j,k}(1-\theta) + \bar{V}_{i,j+1,k}\theta) - (\bar{V}_{i,j-1,k}(1-\theta) + \bar{V}_{i,j,k}\theta)}{\Delta y} \\ &-\frac{(\bar{W}_{i,j,k}(1-\theta) + \bar{W}_{i,j,k+1}\theta) - (\bar{W}_{i,j,k-1}(1-\theta) + \bar{W}_{i,j,k}\theta)}{\Delta z},\end{aligned} \quad (2.29e)$$

with Δx, Δy, and Δz the respective mesh sizes, and $0 \leq \theta \leq 1$.

Energy conservation can be shown by a series of straightforward calculations: multiply equations (2.29a) by $(\bar{U}_{(i,j,k)}, \bar{V}_{(i,j,k)}, \bar{W}_{(i,j,k)}, \bar{R}_{(i,j,k)})$, and sum over all elements. Discretisation (2.29) then leads to energy conservation, i.e.,

$$\frac{dH}{dt} = 0, \quad \text{with} \quad H = \sum_{(i,j,k)} \frac{1}{2}\left(\bar{U}^2_{i,j,k} + \bar{V}^2_{i,j,k} + \bar{W}^2_{i,j,k} + \bar{R}^2_{i,j,k}\right). \quad (2.30)$$

The latter result suggests there is a discrete Hamiltonian formulation for (2.29). We first calculate the partial derivatives of the Hamiltonian

$$\frac{\partial H}{\partial \bar{U}_{i,j,k}} = \bar{U}_{i,j,k}, \quad \frac{\partial H}{\partial \bar{V}_{i,j,k}} = \bar{V}_{i,j,k}, \quad \frac{\partial H}{\partial \bar{W}_{i,j,k}} = \bar{W}_{i,j,k}, \quad \frac{\partial H}{\partial \bar{R}_{i,j,k}} = \bar{R}_{i,j,k} \quad (2.31)$$

and use these in the right hand side of (2.29a). Subsequently, we multiply the four equations in (2.29a) by $\partial F/\partial \bar{U}_{i,j,k}, \partial F/\partial \bar{V}_{i,j,k}, \partial F/\partial \bar{W}_{i,j,k}$, and $\partial F/\partial \bar{R}_{i,j,k}$, respectively, add them up, and sum over all cells. After some algebraic manipulations, it yields the Hamiltonian finite-dimensional dynamics

$$
\begin{aligned}
\frac{dF}{dt} =& [F,H]_{c_\theta} \\
\equiv& -\frac{\partial F}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} \cdot 2\boldsymbol{\Omega} \times \frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} \\
&+ \frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} \cdot \left((1-\theta) \begin{pmatrix} \frac{1}{\Delta x}\big(\frac{\partial F}{\partial \bar{R}_{i+1}} - \frac{\partial F}{\partial \bar{R}_{\mathbf{k}}}\big) \\ \frac{1}{\Delta y}\big(\frac{\partial F}{\partial \bar{R}_{j+1}} - \frac{\partial F}{\partial \bar{R}_{\mathbf{k}}}\big) \\ \frac{1}{\Delta z}\big(\frac{\partial F}{\partial \bar{R}_{k+1}} - \frac{\partial F}{\partial \bar{R}_{\mathbf{k}}}\big) \end{pmatrix} + \theta \begin{pmatrix} \frac{1}{\Delta x}\big(\frac{\partial F}{\partial \bar{R}_{\mathbf{k}}} - \frac{\partial F}{\partial \bar{R}_{i-1}}\big) \\ \frac{1}{\Delta y}\big(\frac{\partial F}{\partial \bar{R}_{\mathbf{k}}} - \frac{\partial F}{\partial \bar{R}_{j-1}}\big) \\ \frac{1}{\Delta z}\big(\frac{\partial F}{\partial \bar{R}_{\mathbf{k}}} - \frac{\partial F}{\partial \bar{R}_{k-1}}\big) \end{pmatrix} \right) \\
&- \frac{\partial F}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} \cdot \left((1-\theta) \begin{pmatrix} \frac{1}{\Delta x}\big(\frac{\partial H}{\partial \bar{R}_{i+1}} - \frac{\partial H}{\partial \bar{R}_{\mathbf{k}}}\big) \\ \frac{1}{\Delta y}\big(\frac{\partial H}{\partial \bar{R}_{j+1}} - \frac{\partial H}{\partial \bar{R}_{\mathbf{k}}}\big) \\ \frac{1}{\Delta z}\big(\frac{\partial H}{\partial \bar{R}_{k+1}} - \frac{\partial H}{\partial \bar{R}_{\mathbf{k}}}\big) \end{pmatrix} + \theta \begin{pmatrix} \frac{1}{\Delta x}\big(\frac{\partial H}{\partial \bar{R}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{R}_{i-1}}\big) \\ \frac{1}{\Delta y}\big(\frac{\partial H}{\partial \bar{R}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{R}_{j-1}}\big) \\ \frac{1}{\Delta z}\big(\frac{\partial H}{\partial \bar{R}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{R}_{k-1}}\big) \end{pmatrix} \right),
\end{aligned}
\tag{2.32}
$$

where $\bar{\mathbf{U}} = (\bar{U}, \bar{V}, \bar{W})^T$ and $\mathbf{k} = (i, j, k)$, and we used the shorthand notation $R_{i+1} = R_{i+1,j,k}$, et cetera. Repeated indices indicate summation, if not stated otherwise. By inspection, (2.32) is seen to be anti-symmetric, and independent of the variables involved. It therefore satisfies all requirement of a (noncanonical) Hamiltonian system. The c_θ subscript indicates the cosymplectic form of the Poisson bracket for a FV discretisation of the compressible Euler equations. Since several brackets appear below, we will use such subscripts to distinguish the different brackets used hereafter.

Discrete incompressible dynamics

The discrete compressible Hamiltonian dynamics (2.32) with (2.30) is our starting point for Dirac's theory of constraints. It will be used to enforce the density as a constraint into the compressible Hamiltonian formulation. For simplicity of illustration, we will only consider the case with $\theta = 0$ in

(2.32). With some minor index renumbering, we obtain

$$
\begin{aligned}
\frac{dF}{dt} = [F,H]_{c_0} \equiv &- \left(2\boldsymbol{\Omega} \times \frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}}\right) \cdot \frac{\partial F}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} \\
&- \left(\frac{\partial H}{\partial \bar{R}_{i+1,j,k}} - \frac{\partial H}{\partial \bar{R}_{\mathbf{k}}}\right) \frac{1}{\Delta x} \frac{\partial F}{\partial \bar{U}_{\mathbf{k}}} - \left(\frac{\partial H}{\partial \bar{R}_{i,j+1,k}} - \frac{\partial H}{\partial \bar{R}_{\mathbf{k}}}\right) \frac{1}{\Delta y} \frac{\partial F}{\partial \bar{V}_{\mathbf{k}}} \\
&- \left(\frac{\partial H}{\partial \bar{R}_{i,j,k+1}} - \frac{\partial H}{\partial \bar{R}_{\mathbf{k}}}\right) \frac{1}{\Delta z} \frac{\partial F}{\partial \bar{W}_{\mathbf{k}}} - \left(\frac{\partial H}{\partial \bar{U}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{U}_{i-1,j,k}}\right) \frac{1}{\Delta x} \frac{\partial F}{\partial \bar{R}_{\mathbf{k}}} \\
&- \left(\frac{\partial H}{\partial \bar{V}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{V}_{i,j-1,k}}\right) \frac{1}{\Delta y} \frac{\partial F}{\partial \bar{R}_{\mathbf{k}}} - \left(\frac{\partial H}{\partial \bar{W}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{W}_{i,j,k-1}}\right) \frac{1}{\Delta z} \frac{\partial F}{\partial \bar{R}_{\mathbf{k}}}.
\end{aligned}
\tag{2.33}
$$

For the continuous problem the primary constraint is the zero perturbation density, $\rho(x,y,z,t) = 0$. For the FV discretisation, it means that the mean value of the density is zero everyhwere,

$$
\bar{R}_{\mathbf{k}} = 0. \tag{2.34}
$$

The primary constraints should hold in time, as a consistency requirement,

$$
0 = \frac{d\bar{R}_{\mathbf{k}}}{dt} = [\bar{R}_{\mathbf{k}}, H]_{c_0} + \mu_{\mathbf{p}}[\bar{R}_{\mathbf{k}}, \bar{R}_{\mathbf{p}}]_{c_0}, \tag{2.35}
$$

with Lagrange multipliers $\mu_{\mathbf{p}}$ and $\mathbf{p} = (p,q,r)$. From (2.33), it follows that $[R_{\mathbf{k}}, R_{\mathbf{p}}]_{c_0} = 0$. The Lagrange multiplier thus remains undetermined and a secondary constraint arises from (2.35) as

$$
\begin{aligned}
E_{\mathbf{k}} \equiv - [\bar{R}_{\mathbf{k}}, H]_{c_0} = &\left(\frac{\partial H}{\partial \bar{U}_{\mathbf{k}}} + \frac{\partial H}{\partial \bar{U}_{i-1,j,k}}\right) \frac{1}{\Delta x} \\
&\left(\frac{\partial H}{\partial \bar{V}_{\mathbf{k}}} + \frac{\partial H}{\partial \bar{V}_{i,j-1,k}}\right) \frac{1}{\Delta y} + \left(\frac{\partial H}{\partial \bar{W}_{\mathbf{k}}} + \frac{\partial H}{\partial \bar{W}_{i,j,k-1}}\right) \frac{1}{\Delta z} \\
= &\frac{(\bar{U}_{\mathbf{k}} - \bar{U}_{i-1,j,k})}{\Delta x} + \frac{(\bar{V}_{\mathbf{k}} - \bar{V}_{i,j-1,k})}{\Delta y} + \frac{(\bar{W}_{\mathbf{k}} - \bar{W}_{i,j,k-1})}{\Delta z} = 0.
\end{aligned}
\tag{2.36}
$$

A closer look at the constraint (2.36) shows that it is a first-order discretisation of the divergence-free velocity condition (2.17). Subsequently, both constraints are enforced together, with mandatory preservation in time

$$0 = \frac{d\bar{R}_{\mathbf{k}}}{dt} = [\bar{R}_{\mathbf{k}}, H]_{c_0} + \lambda^U_{\mathbf{p}}[\bar{R}_{\mathbf{k}}, E_{\mathbf{p}}]_{c_0}, \tag{2.37a}$$

$$0 = \frac{dE_{\mathbf{k}}}{dt} = [E_{\mathbf{k}}, H]_{c_0} + \lambda^R_{\mathbf{p}}[E_{\mathbf{k}}, \bar{R}_{\mathbf{p}}]_{c_0} + \lambda^U_{\mathbf{p}}[E_{\mathbf{k}}, E_{\mathbf{p}}]_{c_0}. \tag{2.37b}$$

Application of primary constraint (2.34) in (2.37a) yields $\lambda^U_{\mathbf{p}}[\bar{R}_{\mathbf{k}}, E_{\mathbf{p}}]_{c_0} = 0$. The latter equation is a FV discretisation of the Laplacian, as follows shortly, and has as solution $\lambda^U_{\mathbf{p}} = 0$. That result simplifies (2.37b) to

$$0 = [E_{\mathbf{k}}, H]_{c_0} + \lambda^R_{\mathbf{p}}[E_{\mathbf{k}}, \bar{R}_{\mathbf{p}}]_{c_0}. \tag{2.38}$$

Further valuation of (2.38) using (2.32) gives an equation for $\lambda \equiv \lambda^R$

$$\begin{aligned} &\frac{\lambda_{i+1,j,k} - 2\lambda_{\mathbf{k}} + \lambda_{i-1,j,k}}{\Delta x^2} + \frac{\lambda_{i,j+1,k} - 2\lambda_{\mathbf{k}} + \lambda_{i,j-1,k}}{\Delta y^2} \\ &+ \frac{\lambda_{i,j,k+1} - 2\lambda_{\mathbf{k}} + \lambda_{i,j,k-1}}{\Delta z^2} = -\frac{\left[2\mathbf{\Omega} \times \left(\frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{\mathbf{U}}_{i-1,j,k}}\right)\right]_1}{\Delta x} \\ &- \frac{\left[2\mathbf{\Omega} \times \left(\frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{\mathbf{U}}_{i,j-1,k}}\right)\right]_2}{\Delta y} - \frac{\left[2\mathbf{\Omega} \times \left(\frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} - \frac{\partial H}{\partial \bar{\mathbf{U}}_{i,j,k-1}}\right)\right]_3}{\Delta z}. \end{aligned} \tag{2.39}$$

Note that (2.39) is a discretisation of the Laplacian acting on the Lagrange multiplier, $\nabla^2\lambda$, on the left-hand-side and the divergence of the rotational effects on the right-hand-side, corresponding to a discrete version of the continuous case (2.24). Finally, the bracket for the incompressible case becomes

$$\begin{aligned} \frac{dF}{dt} = [F, H]^{inc}_{c_0} \equiv -\left(2\mathbf{\Omega} \times \frac{\partial H}{\partial \bar{\mathbf{U}}_{\mathbf{k}}}\right) \cdot \frac{\partial F}{\partial \bar{\mathbf{U}}_{\mathbf{k}}} - \frac{\lambda_{i+1,j,k} - \lambda_{\mathbf{k}}}{\Delta x}\frac{\partial F}{\partial \bar{U}_{\mathbf{k}}} \\ - \frac{\lambda_{i,j+1,k} - \lambda_{i,j,k}}{\Delta y}\frac{\partial F}{\partial \bar{V}_{\mathbf{k}}} - \frac{\lambda^U_{i,j,k+1} - \lambda^U_{\mathbf{k}}}{\Delta z}\frac{\partial F}{\partial \bar{W}_{\mathbf{k}}}. \end{aligned} \tag{2.40}$$

Hence, the constrained dynamics for incompressible fluid flow results in the Dirac-bracket formulation (2.40) coupled with (2.39) for the Lagrange multiplier λ, and the discrete energy functional

$$H = \sum_{(i,j,k)} \frac{1}{2}\left(\bar{U}^2_{\mathbf{k}} + \bar{V}^2_{\mathbf{k}} + \bar{W}^2_{\mathbf{k}}\right). \tag{2.41}$$

Proofs of energy conservation and preservation of zero divergence in time will be presented later for the more general, DGFEM discretisation of incompressible Hamiltonian dynamics.

2.3.2 Discontinuous Galerkin FEM discretisation for the linearised Euler equations

In this section, we will introduce a discontinuous Galerkin FEM discretisation that preserves the Hamiltonian structure of linear, compressible and incompressible flows. The FV discretisation of the Hamiltonian system presented above is used as a guide in the choice of the numerical flux.

Finite element space

Let $\mathcal{I}_h$ denote a tessellation of the domain D with elements K. The set of all edges in the tessellation $\mathcal{I}_h$ is Γ, with Γ_i the set of interior edges and Γ_D the set of edges at the domain boundary ∂D. Additional notation is introduced for the numerical flux, to be introduced shortly. Let e be a face between "left" and "right" elements K_L and K_R, respectively, with corresponding outward normals $\mathbf{n}_L$ and $\mathbf{n}_R$. When f is a continuous function on K_L and K_R, but possibly discontinuous across the face e, let $f_L = (f|_{K_L})|_e$ and $f_R = (f|_{K_R})|_e$ denote the left and right traces, respectively. Let $\mathcal{P}^p(K)$ be the space of polynomials of at most degree p on $K \in \mathcal{I}_h$, with $p \geq 0$. The finite element spaces $\mathcal{Q}_h$ and $\mathcal{Y}_h$ required are

$$\mathcal{Q}_h = \{q \in L^2(D) : q|_K \in \mathcal{P}^p(K), \forall K \in \mathcal{I}_h\}, \tag{2.42a}$$

$$\mathcal{Y}_h = \{\boldsymbol{Y} \in (L^2(D))^3 : \boldsymbol{Y}|_K \in (\mathcal{P}^p(K))^3, \forall K \in \mathcal{I}_h\}. \tag{2.42b}$$

The number of degrees of freedom on an element is denoted by $N_K = \dim(\mathcal{P}^p(K))$.

The discrete energy on the tessellated domain, cf. (2.7), thus becomes

$$H = \frac{1}{2}\sum_K \int_K \left(|\mathbf{u}_h|^2 + \rho_h^2\right)\, dK, \tag{2.43}$$

where $\rho_h \in \mathcal{Q}_h$ and $\mathbf{u}_h \in \mathcal{Y}_h$. Corresponding variational derivatives are

$$\frac{\delta H}{\delta \mathbf{u_h}} = \mathbf{u}_h \quad \text{and} \quad \frac{\delta H}{\delta \rho_h} = \rho_h. \tag{2.44}$$

There is some abuse of notation here, because we use functions F and H for functionals. However, if approximations $\mathbf{u}_h$ and ρ_h are viewed as finite-dimensional expansions, then function derivatives with respect to the expansion coefficients emerge.

Hamiltonian DGFEM discretisation for linearised compressible flow

In this section, we derive a DGFEM discretisation of the Hamiltonian structure for linearised compressible flow (2.6). The specific functional $F[\mathbf{u}_h] \equiv \int_D \mathbf{u}_h \cdot \mathbf{\Phi} d\mathbf{x}$ is chosen to obtain the discretised momentum equations in a Hamiltonian framework, with $\mathbf{\Phi} \in \mathcal{Y}_h$ an arbitrary test function. The functional derivative with respect to the velocity thus equals

$$\frac{\delta F}{\delta \mathbf{u}_h} = \mathbf{\Phi}. \tag{2.45}$$

Likewise, a functional $F[\rho_h] \equiv \int_D \rho_h \phi \, d\mathbf{x}$ is needed, with $\phi \in \mathcal{Q}_h$ an arbitrary test function. Its functional derivative equals

$$\frac{\delta F}{\delta \rho_h} = \phi. \tag{2.46}$$

Our starting point is to simply limit functionals in the Poisson bracket (2.6) on tessellation $\mathcal{I}_h$ to ones on the approximate finite element space, as follows

$$\begin{aligned} \frac{dF}{dt} &= [F, H] \\ &\equiv \sum_K \int_K \left(\frac{\delta H}{\delta \rho_h} \nabla_h \cdot \frac{\delta F}{\delta \mathbf{u}_h} - \frac{\delta F}{\delta \rho_h} \nabla_h \cdot \frac{\delta H}{\delta \mathbf{u}_h} - 2\mathbf{\Omega} \times \frac{\delta H}{\delta \mathbf{u}_h} \cdot \frac{\delta F}{\delta \mathbf{u}_h} \right) d\mathbf{x} \end{aligned} \tag{2.47}$$

with element-wise differential operator ∇_h. After integration by parts of the first two terms on the right-hand-side of (2.47) and introduction of numerical fluxes, we obtain

$$\begin{aligned} \frac{dF}{dt} =& \sum_K \int_K \left(-\nabla_h \frac{\delta H}{\delta \rho_h} \cdot \frac{\delta F}{\delta \mathbf{u}_h} + \frac{\delta H}{\delta \mathbf{u}_h} \cdot \nabla_h \frac{\delta F}{\delta \rho_h} - 2\mathbf{\Omega} \times \frac{\delta H}{\delta \mathbf{u}_h} \cdot \frac{\delta F}{\delta \mathbf{u}_h} \right) dK + \\ & \sum_K \int_{\partial K} \left(\frac{\delta H}{\delta \rho_h} \mathbf{n} \cdot \widehat{\frac{\delta F}{\delta \mathbf{u}_h}} - \widehat{\frac{\delta H}{\delta \mathbf{u}_h}} \cdot \mathbf{n} \frac{\delta F}{\delta \rho_h} \right) d\Gamma, \end{aligned} \tag{2.48a}$$

with element boundaries ∂K. Wide hats on the expressions in the boundary integrals indicate numerical fluxes. We chose the following numerical fluxes

$$\widehat{\frac{\delta F}{\delta \mathbf{u}_h}} = (1-\theta) \frac{\delta F}{\delta \mathbf{u}_h^L} + \theta \frac{\delta F}{\delta \mathbf{u}_h^R} \quad \text{and} \quad \widehat{\frac{\delta H}{\delta \mathbf{u}_h}} = (1-\theta) \frac{\delta H}{\delta \mathbf{u}_h^L} + \theta \frac{\delta H}{\delta \mathbf{u}_h^R}, \tag{2.48b}$$

where L and R indicate the traces from the left and right elements connected to the faces, and $0 \leq \theta \leq 1$. We emphasise the equivalence of the numerical fluxes used in (2.48) and in the FV discretisation for the case with constant basis and test functions on each element.

We use numerical fluxes (2.48b) and rewrite the sum over element boundaries into a sum over all faces. Together with (2.43), it yields the following DGFEM discretisation for linear, compressible Hamiltonian dynamics

$$\begin{aligned}\frac{dF}{dt} = &\sum_K \int_K \left(-\nabla_h \frac{\delta H}{\delta \rho_h} \cdot \frac{\delta F}{\delta \mathbf{u}_h} + \frac{\delta H}{\delta \mathbf{u}_h} \cdot \nabla_h \frac{\delta F}{\delta \rho_h} - 2\boldsymbol{\Omega} \times \frac{\delta H}{\delta \mathbf{u}_h} \cdot \frac{\delta F}{\delta \mathbf{u}_h} \right) dK + \\ &\sum_{e \in \Gamma_i} \int_e \left(\frac{\delta H}{\delta \rho_h^L} - \frac{\delta H}{\delta \rho_h^R} \right) \mathbf{n} \cdot \left((1-\theta) \frac{\delta F}{\delta \mathbf{u}_h^L} + \theta \frac{\delta F}{\delta \mathbf{u}_h^R} \right) + \\ &\qquad \left(\frac{\delta F}{\delta \rho_h^R} - \frac{\delta F}{\delta \rho_h^L} \right) \mathbf{n} \cdot \left(\frac{\delta H}{\delta \mathbf{u}_h^R} \theta + \frac{\delta H}{\delta \mathbf{u}_h^L} (1-\theta) \right) d\Gamma. \end{aligned} \tag{2.49}$$

Here $\mathbf{n} = \mathbf{n}^L$ is the exterior normal vector connected with element K^L.

Technically speaking, periodic boundary conditions can be specified in ghost cells (denoted with subscript R), where values of the variables exactly coincide with the face-adjacent cell values (denoted with subscript L_p) at the other side of the periodic boundary

$$\frac{\delta H}{\delta \mathbf{U}^R} \cdot \mathbf{n} = \frac{\delta H}{\delta \mathbf{U}^{L_p}} \cdot \mathbf{n} \qquad \text{and} \qquad \frac{\delta H}{\delta \rho^R} = \frac{\delta H}{\delta \rho^{L_p}}, \tag{2.50}$$

with $\mathbf{n}$ the normal to the boundary face. Geometrically speaking, there are of course only internal cells in a periodic domain. In the case of a three-dimensional cuboid bounded by solid walls, the numerical fluxes on both the test functions and the Hamiltonian derivatives must vanish, cf. our specifications in (2.13). In terms of ghost cells, it implies that

$$(1-\theta) \frac{\delta F}{\delta \mathbf{u}_h^L} + \theta \frac{\delta F}{\delta \mathbf{u}_h^R} = 0 \quad \text{and} \quad (1-\theta) \frac{\delta H}{\delta \mathbf{u}_h^L} + \theta \frac{\delta H}{\delta \mathbf{u}_h^R} = 0 \quad \text{at} \quad \Gamma_D. \tag{2.51}$$

We will use, or rather assume, shortly that boundary conditions for incompressible flow should automatically satisfied by using Dirac's theory, given that those boundary conditions are satisfied for the discrete, compressible Hamiltonian discretisation.

By construction, the bracket (2.49) remains skew-symmetric. Unconventional is that the numerical flux is also acting on the test functions

$\delta\mathcal{F}/\delta\mathbf{u}_h$. We refer to [109] for a proof that the bracket (2.49) can be transformed to a classical, discontinuous Galerkin finite element weak formulation with alternating fluxes, provided $\theta = 1/2$ at boundary faces and for constant material parameters. When material parameters are a function of space, then the Hamiltonian formulation with its division between bracket and Hamiltonian becomes crucial. Not only the skew-symmetric or alternating fluxes matter then but also the dual, Hamiltonian projection. The DG discretisation with a polynomial approximation of order zero will exactly coincide with our FV discretisation. Note that for $\theta = 1/2$ the well-known image boundary condition emerges from (2.51). We emphasise, though, that for $\theta \neq 1/2$ our general condition (2.51) still applies, but that it seems no longer quite equivalent to the standard, alternating flux formulation applied directly to the PDEs.

Variables are expanded on each element K in terms of local basis functions such that: $\mathbf{u}_h = \phi_\beta \mathbf{u}_\beta$ and $\rho_h = \phi_\beta \rho_\beta$. Both coefficients and test functions require elemental superscripts, which we silently omit. Greek numerals are used locally on each element K and we apply the summation convention over repeated indices. Variational and function derivatives are then related as follows

$$\begin{aligned}
\delta\mathcal{F} =& \sum_K \int_K \frac{\delta\mathcal{F}}{\delta\mathbf{u}_h}\delta\mathbf{u}_h + \frac{\delta\mathcal{F}}{\delta\rho_h}\delta\rho_h dK && \text{(2.52a)}\\
=& \sum_K \big(\int_K \frac{\delta\mathcal{F}}{\delta\mathbf{u}_h}\phi_\beta dK\big)\delta\mathbf{u}_\beta + \big(\int_K \frac{\delta\mathcal{F}}{\delta\rho_h}\phi_\beta dK\big)\delta\rho_\beta && \text{(2.52b)}\\
=& \sum_K \frac{\partial F}{\partial\mathbf{u}_\beta}\delta\mathbf{u}_\beta + \frac{\partial F}{\partial\rho_\beta}\delta\rho_\beta. && \text{(2.52c)}
\end{aligned}$$

Similarly, by starting from (2.52c) and using the relation

$$M_{\alpha\beta}\mathbf{u}_\beta = \int_K \phi_\alpha \mathbf{u}_h dK, \tag{2.53}$$

with local mass matrix $M_{\alpha\beta} = M^K_{\alpha\beta}$, one can derive

$$\frac{\delta\mathcal{F}}{\delta\mathbf{u}_h} = M^{-1}_{\beta\gamma}\frac{\partial F}{\partial\mathbf{u}_\beta}\phi_\gamma \qquad \text{and} \qquad \frac{\delta\mathcal{F}}{\delta\rho_h} = M^{-1}_{\beta\gamma}\frac{\partial F}{\partial\rho_\beta}\phi_\gamma. \tag{2.54}$$

By substitution of (2.54) into (2.49), we immediately derive the desired form of the finite-dimensional Hamiltonian discretisation

$$\begin{aligned}\frac{dF}{dt} =&\sum_K \left(\frac{\partial H}{\partial \mathbf{u}_\beta}\frac{\partial F}{\partial \rho_\alpha} - \frac{\partial F}{\partial \mathbf{u}_\beta}\frac{\partial H}{\partial \rho_\alpha}\right)\cdot \mathbf{E}_{\gamma\mu} M^{-1}_{\beta\gamma} M^{-1}_{\alpha\mu} - 2\mathbf{\Omega} \times \frac{\partial H}{\partial \mathbf{u}_\alpha}\cdot\frac{\partial F}{\partial \mathbf{u}_\beta} M^{-1}_{\alpha\beta}\\
&+\sum_{e\in\Gamma_i}(1-\theta)\left(\frac{\partial H}{\partial \rho^L_\alpha}\frac{\partial F}{\partial \mathbf{u}^L_\beta} - \frac{\partial F}{\partial \rho^L_\alpha}\frac{\partial H}{\partial \mathbf{u}^L_\beta}\right)\cdot \mathbf{G}^{LL}_{\gamma\mu} M^{-L}_{\alpha\mu} M^{-L}_{\beta\gamma}\\
&\qquad+\theta\left(\frac{\partial H}{\partial \rho^L_\alpha}\frac{\partial F}{\partial \mathbf{u}^R_\beta} - \frac{\partial F}{\partial \rho^L_\alpha}\frac{\partial H}{\partial \mathbf{u}^R_\beta}\right)\cdot \mathbf{G}^{LR}_{\gamma\mu} M^{-L}_{\alpha\mu} M^{-R}_{\beta\gamma}\\
&\qquad-(1-\theta)\left(\frac{\partial H}{\partial \rho^R_\alpha}\frac{\partial F}{\partial \mathbf{u}^L_\beta} - \frac{\partial F}{\partial \rho^R_\alpha}\frac{\partial H}{\partial \mathbf{u}^L_\beta}\right)\cdot \mathbf{G}^{RL}_{\gamma\mu} M^{-R}_{\alpha\mu} M^{-L}_{\beta\gamma}\\
&\qquad-\theta\left(\frac{\partial H}{\partial \rho^R_\alpha}\frac{\partial F}{\partial \mathbf{u}^R_\beta} - \frac{\partial F}{\partial \rho^R_\alpha}\frac{\partial H}{\partial \mathbf{u}^R_\beta}\right)\cdot \mathbf{G}^{RR}_{\gamma\mu} M^{-R}_{\alpha\mu} M^{-R}_{\beta\gamma}\end{aligned} \tag{2.55}$$

with elemental (vector) matrices $\mathbf{E}_{\gamma\mu}$ and $\mathbf{G}^{LR}_{\gamma\mu}$ etc. These read

$$\mathbf{E}_{\gamma\mu} = \int_K \phi_\gamma \nabla_h \phi_\mu dK \qquad \text{and} \qquad \mathbf{G}^{LR}_{\gamma\mu} = \int_e \mathbf{n}\phi^L_\mu \phi^R_\gamma d\Gamma, \tag{2.56}$$

with similar relations for other terms. Finally, after substitution of (2.54) into Hamiltonian (2.43), it becomes

$$H = \frac{1}{2}\sum_K M_{\alpha\beta}\left(\mathbf{u}_\alpha\cdot\mathbf{u}_\beta + \rho_\alpha\rho_\beta\right). \tag{2.57}$$

A global formulation is useful for the incompressible case. We therefore introduce a reordering into global coefficients $\mathbf{U}_i = \mathbf{U}_i(t) = (U,V,W)^T_i$ and $R_k(t)$, instead of the elemental ones, in the finite element expansion of $\mathbf{u}_h$ and ρ_h with indices running over their respective, global ranges. It turns out that the local matrices $M_{\alpha\beta}$ and $\mathbf{E}_{\gamma\mu}$ in (2.55) and (2.57) readily extend to global matrices M_{ij} and $\mathbf{E}_{ij}$. These have a block structure in which each elemental matrix fits in separation from the others. The contribution of the numerical fluxes lead to coupling between the elements, which can be incorporated into a global matrix $\mathbf{G}_{ij}$. The latter is straightforwardly defined computationally by a loop over the faces, and we will therefore not provide an explicit expression. The resulting, global Hamiltonian formulation then becomes

$$\begin{aligned}\frac{dF}{dt} = [F,H]_d \equiv&\left(\frac{\partial H}{\partial \mathbf{U}_j}\frac{\partial F}{\partial R_i} - \frac{\partial F}{\partial \mathbf{U}_j}\frac{\partial H}{\partial R_i}\right)\cdot \mathbf{DIV}_{kl} M^{-1}_{ik} M^{-1}_{jl} \qquad (2.58a)\\
&- 2\mathbf{\Omega}\times\frac{\partial H}{\partial \mathbf{U}_i}\cdot\frac{\partial F}{\partial \mathbf{U}_j} M^{-1}_{ij}\end{aligned}$$

with the divergence vector operator $\mathbf{DIV}_{kl} \equiv \mathbf{E}_{kl} - \mathbf{G}_{kl}$ and global Hamiltonian

$$H = \frac{1}{2} M_{ij} \left(\mathbf{U}_i \cdot \mathbf{U}_j + R_i R_j \right). \tag{2.58b}$$

The resulting equations of motion arising from (2.58) are

$$\dot{\mathbf{U}}_j = - M_{jk}^{-1} R_l \mathbf{DIV}_{kl} - 2\mathbf{\Omega} \times \mathbf{U}_j \tag{2.59a}$$

$$M_{kl} \dot{R}_l = \mathbf{U}_j \cdot \mathbf{DIV}_{jk} \tag{2.59b}$$

with the dot denoting a time derivative.

Hamiltonian DGFEM discretisation for linearised incompressible flow

In close analogy with the continuous case and the FV-case, Dirac's theory is applied to the Hamiltonian dynamics (2.58). The density expansion coefficients are all restricted in every local element such that the resulting density in the element is zero. The following primary constraints are imposed on the discrete density field

$$D_k = M_{kl} R_l. \tag{2.60}$$

Preservation of the constraints in time leads to the following consistency relation

$$0 = \dot{D}_k = [D_k, H]_d + \lambda_l [D_k, D_l]_d. \tag{2.61}$$

Using (2.60) in the bracket (2.58a) shows that $[D_k, D_l]_d = 0$. The Lagrange multipliers λ_l in (2.61) thus remain undetermined, but the consistency requirement gives rise to secondary constraints $L_k = [D_k, H]_d = 0$. Analogous to the continuous and FV-case, the secondary constraint is the discrete version of the divergence-free velocity field property (2.17). To wit

$$L_k = [D_k, H] \equiv \mathbf{DIV}_{lk} \cdot \mathbf{U}_l, \tag{2.62}$$

with the discrete divergence operator $\mathbf{DIV}_{lk}$. We start again with both primary constraints and require these to remain preserved under the Hamiltonian dynamics. We obtain

$$0 = \dot{D}_k = [D_k, H]_d + \mu_l [D_k, L_l]_d, \tag{2.63a}$$

$$0 = \dot{L}_k = [L_k, H]_d + \lambda_l [L_k, D_l]_d + \mu_l [L_k, L_l]_d. \tag{2.63b}$$

Application of the primary constraint implies that $L_k = [D_k, H] = 0$ in (2.63a). Hence,

$$\mu_l [D_k, L_l] = \mu_l \mathbf{DIV}_{ml} M_{jm}^{-1} \cdot \mathbf{DIV}_{jk} = 0. \tag{2.64}$$

The matrix acting on μ_l is a discrete Laplacian. It is nonsingular, whence $\mu_l = 0$. Consequently, (2.63b) reduces to

$$\begin{aligned} 0 = \dot{L}_k =& [L_k, H]_d + \lambda_l [L_k, D_l]_d && (2.65a) \\ =& -\mathbf{DIV}_{jk} \cdot 2\boldsymbol{\Omega} \times \mathbf{U}_j - \lambda_l \mathbf{DIV}_{jk} M_{jm}^{-1} \cdot \mathbf{DIV}_{ml}, && (2.65b) \end{aligned}$$

which is the discrete equivalent of the Poisson equation in (2.28). Finally, the resulting discrete, linear, incompressible Hamiltonian dynamics is given by

$$\frac{dF}{dt} = [F, H]_{inc} \equiv -\frac{\partial F}{\partial \mathbf{U}_j} \cdot \left(2\boldsymbol{\Omega} \times \frac{\partial H}{\partial \mathbf{U}_i} M_{ij}^{-1} + \lambda_l M_{jk}^{-1} \mathbf{DIV}_{kl} \right) \tag{2.66a}$$

with energy function

$$H = \frac{1}{2} M_{ij} \mathbf{U}_i \cdot \mathbf{U}_j. \tag{2.66b}$$

The final system consists of the ordinary differential equations following directly from (2.66) after using $F = \mathbf{U}_j$, as follows

$$\dot{\mathbf{U}}_j = -2\boldsymbol{\Omega} \times \mathbf{U}_j - \lambda_l M_{jk}^{-1} \mathbf{DIV}_{kl}, \tag{2.67}$$

combined with (2.65):

$$\lambda_l \mathbf{DIV}_{jk} M_{jm}^{-1} \cdot \mathbf{DIV}_{ml} = -\mathbf{DIV}_{jk} \cdot 2\boldsymbol{\Omega} \times \mathbf{U}_j. \tag{2.68}$$

2.4 Time Integrator

We consider a symplectic time integrator for the time discretisation of linear compressible (2.59) and incompressible (2.67) Hamiltonian dynamics. Symplectic time integrators form the subclass of geometric integrators which, by definition, are canonical transformations. The modified midpoint time integrator was chosen amongst other symplectic schemes [44]. It is implicit, which requires more computation, but that pays off in dealing with the momentum and continuity equations in a rotating frame of reference.

2.4.1 Linear, compressible flow

Applying the modified midpoint scheme to the discrete compressible Hamiltonian dynamics (2.58) or (2.59), one gets

$$\frac{\mathbf{U}_j^{n+1}-\mathbf{U}_j^n}{\Delta t} = -M_{jk}^{-1}\frac{(R_l^{n+1}+R_l^n)}{2}\mathbf{DIV}_{kl} - \mathbf{\Omega}\times(\mathbf{U}_j^{n+1}+\mathbf{U}_j^n) \tag{2.69a}$$

$$M_{kl}\frac{(R_l^{n+1}-R_l^n)}{\Delta t} = \frac{\mathbf{U}_j^{n+1}+\mathbf{U}_j^n}{2}\cdot\mathbf{DIV}_{jk}. \tag{2.69b}$$

Proposition 1. *The numerical scheme for linear, compressible fluid flow given by* (2.69) *is exactly energy conserving, such that* $M_{ij}(\mathbf{U}_i^{n+1}\cdot\mathbf{U}_j^{n+1}+R_i^{n+1}R_j^{n+1}) = M_{ij}(\mathbf{U}_i^n\cdot\mathbf{U}_j^n+R_i^nR_j^n)$.

Proof. Multiply equations (2.69) with $M_{ij}\mathbf{U}_i^{n+1}$, R_k^{n+1} and $M_{ij}\mathbf{U}_i^n$, R_k^n. Thereafter, add them up. After some manipulation, the Hamiltonian on the $(n+1)$-th time level can be shown to equal the Hamiltonian on the n-th level. □

2.4.2 Incompressible flow

The midpoint time integrator is also applied to the incompressible discrete Hamiltonian dynamics (2.66) or (2.67), giving

$$\frac{(\mathbf{U}_j^{n+1}-\mathbf{U}_j^n)}{\Delta t} = -\mathbf{\Omega}\times(\mathbf{U}_j^{n+1}+\mathbf{U}_j^n) - \lambda_l^{n+1}M_{jk}^{-1}\mathbf{DIV}_{kl} \tag{2.70a}$$

$$\lambda_l^{n+1}\mathbf{DIV}_{jk}M_{jm}^{-1}\cdot\mathbf{DIV}_{ml} = -\mathbf{DIV}_{jk}\cdot\mathbf{\Omega}\times(\mathbf{U}_j^{n+1}+\mathbf{U}_j^n). \tag{2.70b}$$

Proposition 2. *The numerical scheme for linear, incompressible fluid flow given by* (2.70) *exactly conserves energy and the discrete zero-divergence property in time, such that* $\mathbf{DIV}_{jm}\cdot\mathbf{U}_j^{n+1}=\mathbf{0}$ *given that* $\mathbf{DIV}_{jm}\cdot\mathbf{U}_j^0=\mathbf{0}$ *and* $M_{ij}\mathbf{U}_i^{n+1}\cdot\mathbf{U}_j^{n+1} = M_{ij}\mathbf{U}_i^n\cdot\mathbf{U}_j^n$.

Proof. Firstly, we present the proof for the conservation of the discrete zero-divergence, under the assumption that the current velocity of the n^{th}–time level is discretely divergence free, i.e., $L_m = \mathbf{DIV}_{jm}\cdot\mathbf{U}_j^n = 0$. We apply the discrete divergence operator on both sides of (2.70a) and use that the present velocity is divergence free, to obtain

$$\begin{aligned}\mathbf{DIV}_{jm}\cdot\mathbf{U}_j^{n+1}/\Delta t = &-\mathbf{DIV}_{jm}\cdot\mathbf{\Omega}\times(\mathbf{U}_j^{n+1}+\mathbf{U}_j^n)\\ &-\mathbf{DIV}_{jm}\cdot\lambda_l^{n+1}M_{jk}^{-1}\mathbf{DIV}_{kl}.\end{aligned} \tag{2.71}$$

The right hand side of (2.71) exactly coincides with (2.70b) and therefore $\mathbf{DIV}_{jm} \cdot \mathbf{U}_j^{n+1} = \mathbf{0}$.

Secondly, energy conservation means that the discrete Hamiltonian energy functional (2.66b) stays unchanged in time. Multiplication of (2.70a) with $M_{ij}\mathbf{U}_i^{n+1}$ and $M_{ij}\mathbf{U}_i^n$, followed by summation of both equations yields

$$M_{ij}\frac{(\mathbf{U}_i^{n+1} \cdot \mathbf{U}_j^{n+1} - \mathbf{U}_i^n \cdot \mathbf{U}_j^n)}{\Delta t} = -\,\lambda_l^{n+1}\mathbf{DIV}_{il} \cdot (\mathbf{U}_i^{n+1} + \mathbf{U}_i^n) = 0, \quad (2.72)$$

since the terms involving rotational effects cancel and in the last step we use that present and future time velocities are divergence free, as shown in the first part of this proof. Hence, the difference of the energy at the $(n+1)^{\text{th}}$ and n^{th} level is zero. ☐

2.4.3 Initial conditions

As proven above, the discontinuous Galerkin discretisation for linearised incompressible fluid flow conserves energy and is divergence free at the discrete level. The proofs require a discrete, divergence-free initial condition, i.e., $\mathbf{DIV}_{jm} \cdot \mathbf{U}_j^0 = 0$. This condition is not guaranteed automatically, since the projection of the initial, divergence-free velocity field on the chosen discontinuous Galerkin finite element space only satisfies discrete zero-divergence up to the order of accuracy. We therefore require a preprocessing step on this projected velocity $\mathbf{U}$. We are looking for a $\mathbf{U}^*$ for which $\mathbf{DIV}_{jm} \cdot \mathbf{U}_j^* = 0$ exactly and $||\mathbf{U}^* - \mathbf{U}||$ is minimal. Note that the matrix $\mathbf{DIV}_{jm}$ is not square. Hereafter, we denote the associated, global matrix by DIV and the vector of velocity unknowns as $\mathbf{U}$ (so without indices). Basically, the latter problem transforms into a well-known problem in vector calculus: find a projection of the vector $\mathbf{U}$ on the space A: the null-space of discrete divergence matrix operator DIV, defined as

$$A = \{\mathbf{Q} \in \mathbb{R}^{3N_{dof}} : \quad DIV\,\mathbf{Q} = 0\} \quad (2.73)$$

with N_{dof} the degrees of freedom per velocity component (assuming them to be equal for simplicity).

From linear algebra [38, 37], we obtain that the closest vector from the A-space will be the projection of vector $\mathbf{U}$ on the space, which is

$$\mathbf{U}^* \equiv proj_A\mathbf{U} = \mathbf{U} + \mathbf{U}_\perp. \quad (2.74)$$

Applying the DIV–operator on (2.74), we find

$$0 = DIV\mathbf{U}^* = DIV\mathbf{U} + DIV\mathbf{U}_\perp, \quad (2.75)$$

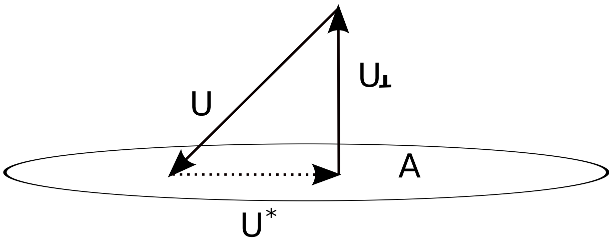


Figure 2.1: Projection of vector $\mathbf{U}$ on the null-space of matrix DIV.

which results in

$$DIV\mathbf{U}_\perp = -DIV\mathbf{U}. \tag{2.76}$$

The latter equation is solved for $\mathbf{U}_\perp$ via a least-square approximation [42] up to machine precision. Hence, the projected velocity is preprocessed using (2.74), and the resulting velocity field has become exactly discrete divergence free, as required.

2.4.4 Other properties of the algebraic system

A direct DGFEM discretisation of the incompressible Navier-Stokes equations (or the Euler equations as special case) generally requires the inf-sup condition to be satisfied to attain numerical stability [39, 71]. In order to get a stable pressure approximation, two different strategies are often pursued: either a pressure stabilisation term is used or the approximation spaces for velocity and pressure are chosen (differently) such that an inf-sup compatibility condition is fulfilled. Nonetheless, our numerical discretisation for linear, incompressible flow does not suffer from those drawbacks. The exact preservation of the Hamiltonian dynamics as well as the constraints makes the system unconditionally stable.

Furthermore, the three-dimensionality of the problem results in a large algebraic system, which we represent using the sparse matrix structures available in PETSc [10, 91]. Figure 2.2 shows the sparsity of a matrix, needed to determine $\mathbf{U}_j^{n+1}$ and λ_l^{n+1} in the discretisation for incompressible flow (2.69). We use a linear, iterative solver to converge to the desired tolerance. To improve the convergence rate of the iterative solver ILU preconditioners were used with controlled memory usage of the resulting algebraic system.

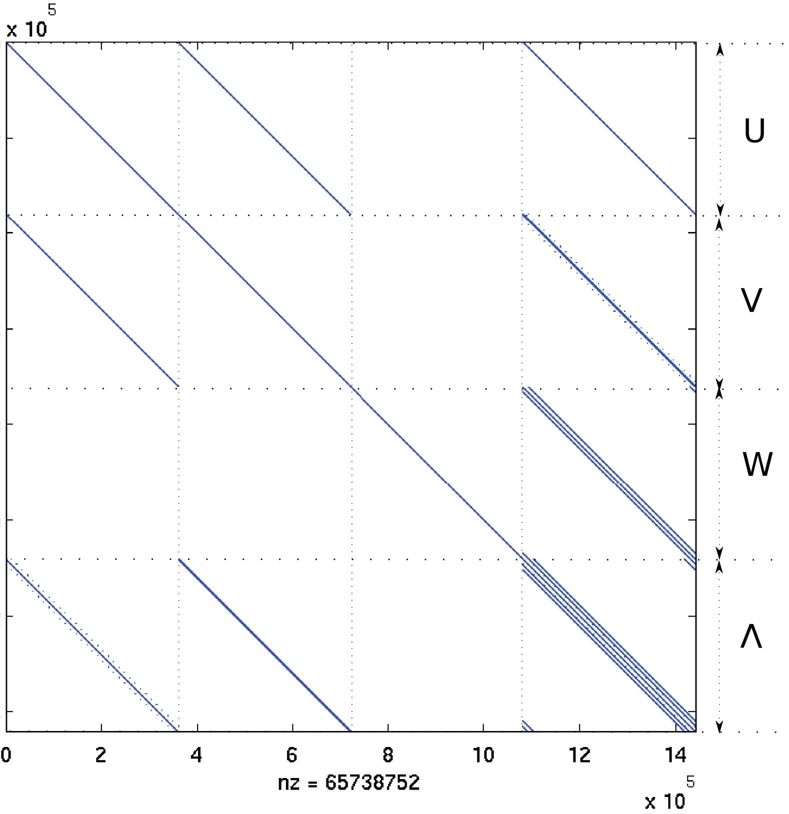


Figure 2.2: A structure of the resulting square sparse matrix with more than 10^8 non-zero elements and $\Omega_1 = \Omega_2 = 0$, and $\Omega_3 = 1$. Λ denotes the vector of unknown Lagrange multipliers.

2.5 Tests of numerical scheme

Discretisations for linear compressible and incompressible flows were implemented for discontinuous Galerkin methods in the hpGEM C++ software framework [80]. The developed applications are consequently highly object-oriented, easy to maintain and extend. Although the tests considered concern cuboids, the implementation can cope with general geometries and meshes. The three-dimensionality of the problem poses, however, significant requirements on speed and memory use. The sparse matrix data structures available in the PETSc library are therefore used. An ILU preconditioner is applied to the linear algebraic system before applying a GMRES linear solver [10, 91]. The number of iterations varies for the different test cases and strongly depends on the dimensions of the algebraic system, e.g., the grid size and the amount of basis functions. In the case of quadratic polynomial basis functions with a grid of 64 × 64 × 64 elements, which is the computationally most demanding case, one needs approximately forty GMRES iterations to reach the tolerance 10^{-14} in solving the algebraic equation for incompressible flow.

Although the main goal is to simulate inertial waves in a rectangular box, several extra test cases were performed to verify the approaches and techniques used. Two text cases (periodic waves and waves in a domain with no-slip boundaries) for the compressible fluid were implemented, tested and validated by comparison with an available exact solutions. Additionally, an attempt has been made to attain energy attractors in the domain with a geometrical asymmetry. In all tests presented, $\theta = 1/2$ was used in the numerical flux. Other values of $0 \leq \theta \leq 1$ were also used for various test cases with similar results.

2.5.1 Compressible harmonic waves in a periodic domain

Consider linear, compressible fluid flow with zero rotation $\mathbf{\Omega} = (0, 0, 0)$ in a rectangular triple periodic domain $D = [0, 1]^3$. The following expressions satisfy the linear Euler equations

$$u = A_1 \cos(2k_x(t - x)), \tag{2.77a}$$

$$v = A_2 \cos(2k_y(t - y)), \tag{2.77b}$$

$$w = A_3 \cos(2k_z(t - z)), \tag{2.77c}$$

$$\rho = A_1 \cos(2k_x(t - x)) + A_2 \cos(2k_y(t - y)) + A_3 \cos 2k_z(t - z)). \tag{2.77d}$$

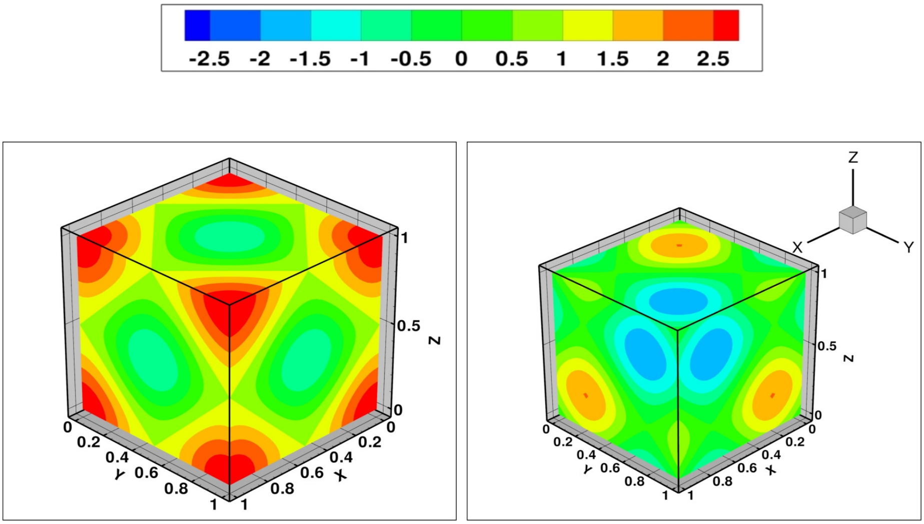
Each component of the velocity vector is a traveling wave in the direction of the corresponding axes. The numerical discretisation is initialised with (2.77) at $t = 0$, $k_x = k_y = k_z = 2\pi$ and $A_1 = A_2 = A_3 = 1$. Numerical and exact solutions were compared during several periods. Figure 2.3 presents the numerical density profile during one time period of the traveling waves. Discrete energy is conserved up to machine precision even after one hundred periods. The results of a convergence analysis, presented in Table 2.1, show that the numerical scheme is first, second and third order accurate in space for, respectively constant, linear and quadratic polynomial approximations.

Table 2.1: Convergence of the error for compressible traveling waves in a three-dimensional periodic domain. Due to symmetry all velocity components have the same error.

		$p = 0$		$p = 1$		$p = 2$	
Grid		l^2-error	order	l^2-error	order	l^2-error	order
4x4x4	**u**	1.011e+0	–	4.3024E-1	–	1.8532E-2	–
	ρ	1.751e+0	–	7.4520E-1	–	3.2098E-2	–
8x8x8	**u**	8.249E-1	0.3	1.3087E-1	1.7	3.0944E-3	2.6
	ρ	1.428e+0	0.3	2.2667E-1	1.7	5.3598E-3	2.6
16x16x16	**u**	2.405E-1	1.8	2.9192E-2	2.1	3.4723E-4	3.2
	ρ	4.166E-1	1.8	5.0554E-2	2.1	6.0142E-4	3.2
32x32x32	**u**	7.193E-2	1.7	5.0432E-3	2.5	2.9366E-5	3.6
	ρ	1.245E-1	1.7	8.7321E-3	2.5	5.0864E-5	3.6
64x64x64	**u**	2.638E-2	1.5	1.0813E-3	2.2	1.8266E-6	4.0
	ρ	4.579E-2	1.4	2.3901E-3	1.8	3.1638E-6	4.0

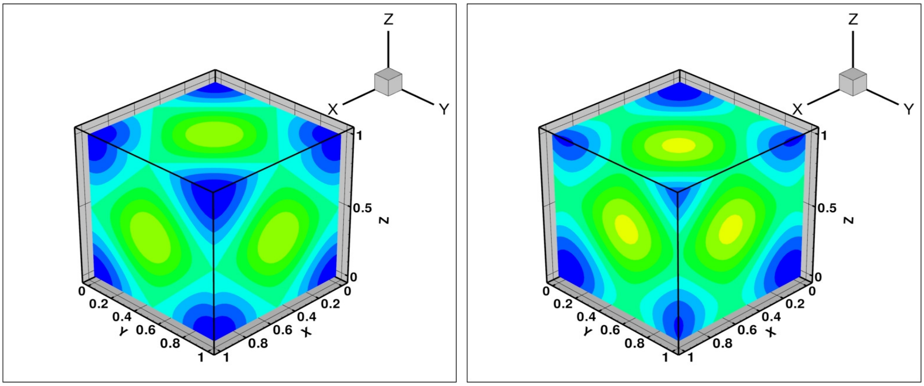
2.5.2 Compressible waves with slip-flow boundary conditions

Next, linear, acoustic fluid flow is considered in domain D, but now slip-flow boundary conditions are used with zero normal component of the velocity field at domain boundaries. Boundary conditions are effectively implemented with the help of ghost cells, where the values of velocity and density fields are specified to satisfy the boundary conditions. One can



(a) Density profile at t=0

(b) Density profile at t=T/4



(c) Density profile at t=T/2

(d) Density profile at t=3T/4

Figure 2.3: Plots of the density field computed with the discretised compressible Hamiltonian formulation. A 32 x 32 x 32 grid with time step $\Delta t = T/20$, where T is the time period of the harmonic waves, was used in a periodic domain.

check that

$$u = -A_1 \sin(2k_x x)\cos(\omega t), \tag{2.78a}$$
$$v = -A_2 \sin(2k_y y)\cos(\omega t), \tag{2.78b}$$
$$w = -A_3 \sin(2k_z z)\cos(\omega t), \tag{2.78c}$$
$$\rho = A_1 \sin(2k_x x)\cos(\omega t) + A_2 \sin(2k_y y)\cos(\omega t) + A_3 \sin(2k_z z)\cos(\omega t). \tag{2.78d}$$

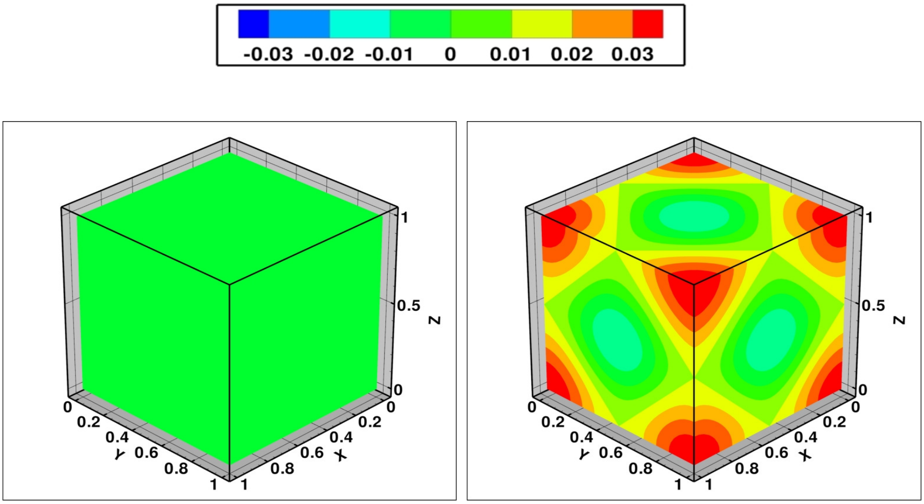
exactly satisfy the linearised compressible Euler equations with slip-flow boundary conditions. The numerical scheme is initialised using (2.78) at $t = 0$, where $\omega = k_x = k_y = k_z = 2\pi$ and $A_1 = A_2 = A_3 = 1$. Figure 2.4 shows the numerical density profile during a full time period. Discrete energy again is conserved up to machine precision, after one hundred wave periods. A convergence analysis is given in Table 2.2.

Table 2.2: Convergence of the error for compressible standing waves in a cuboid with solid walls. Due to symmetry all velocity components have the same error.

		$p = 0$		$p = 1$		$p = 2$	
Grid		l^2-error	order	l^2-error	order	l^2-error	order
4x4x4	$\mathbf{u}$	7.909E-1	–	1.6871E-1	–	7.6351E-3	–
	ρ	2.391e+0	–	6.8551E-1	–	2.9242E-2	–
8x8x8	$\mathbf{u}$	5.012E-1	0.7	6.8455E-2	1.3	3.2262E-4	4.5
	ρ	1.134e+0	0.6	1.9319E-1	1.8	5.3301E-3	2.5
16x16x16	$\mathbf{u}$	8.794E-2	2.5	6.2392E-3	3.4	4.2721E-5	3.0
	ρ	3.877E-1	1.4	4.9391E-2	2.0	5.9395E-4	3.2
32x32x32	$\mathbf{u}$	4.013E-2	1.1	1.0702E-3	2.5	5.6751E-6	4.6
	ρ	1.033E-1	1.9	8.5331E-3	2.5	5.0721E-5	3.5
64x64x64	$\mathbf{u}$	2.003E-2	1.0	2.5982E-4	2.0	7.0557E-7	3.0
	ρ	2.973E-2	1.8	2.3531E-3	1.9	5.1638E-6	3.1

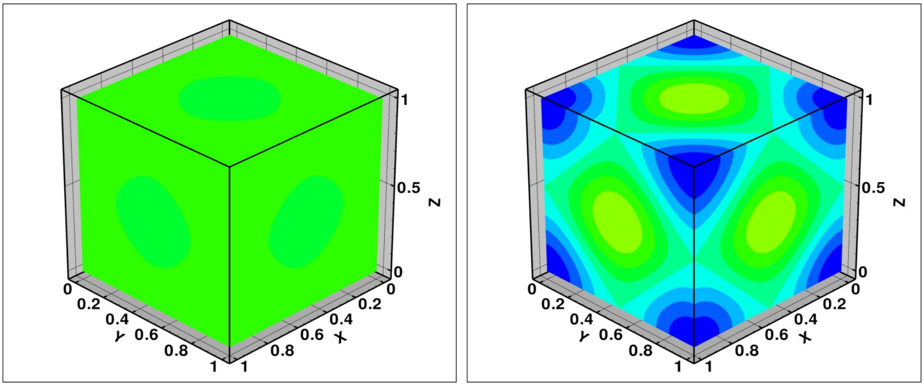
2.5.3 Incompressible waves in a periodic domain

The compressible test cases were mainly interesting as a quality assurance step for linearised incompressible fluid flow, which we consider next. An exact solution was found for the linear, incompressible, rotational Euler



(a) Density profile at t=0

(b) Density profile at t=T/4



(c) Density profile at t=T/2

(d) Density profile at t=3T/4

Figure 2.4: The results are obtained on a 32 x 32 x 32 grid with $\Delta t = T/20$, where the T is the time period of the standing, compressible waves in a closed cuboid.

equations (2.28) with periodic boundary conditions

$$u = \frac{1}{2\pi}\left[\sqrt{3}\cos\left(2\pi(x+y+z)+\frac{\sqrt{3}}{3}t\right)+3\sin\left(2\pi(x+y+z)+\frac{\sqrt{3}}{3}t\right)\right], \tag{2.79a}$$

$$v = \frac{1}{2\pi}\left[\sqrt{3}\cos\left(2\pi(x+y+z)+\frac{\sqrt{3}}{3}t\right)-3\sin\left(2\pi(x+y+z)+\frac{\sqrt{3}}{3}t\right)\right], \tag{2.79b}$$

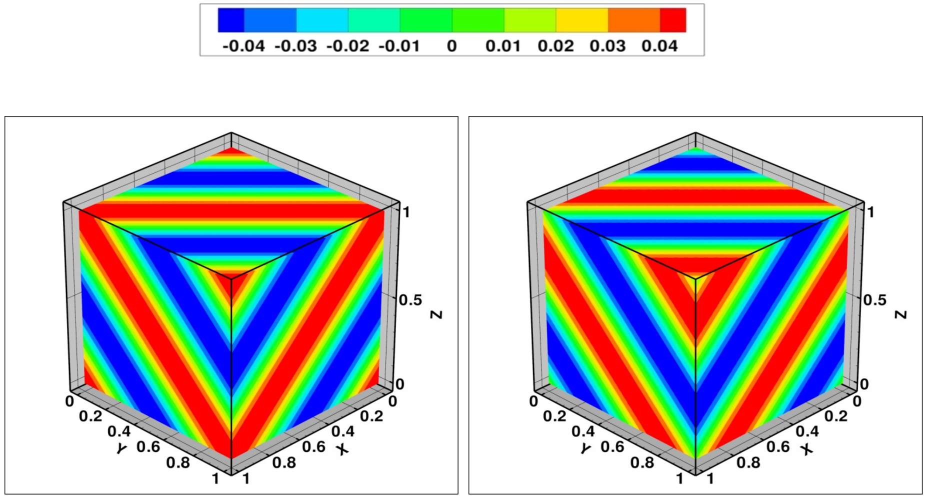
$$w = -\frac{1}{\pi}\sqrt{3}\cos\left(2\pi(x+y+z)+\frac{\sqrt{3}}{3}t\right), \tag{2.79c}$$

$$P = \frac{1}{2\pi^2}\cos\left(2\pi(x+y+z)+\frac{\sqrt{3}}{3}t\right), \tag{2.79d}$$

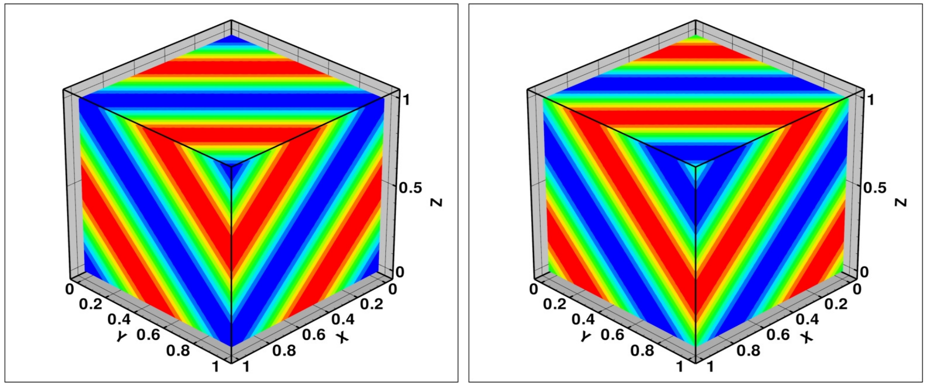
where the rotation vector is $\mathbf{\Omega} = (0,0,1)$ and P is the pressure. This exact solution is used for the initialisation in a periodic domain $D = [0,1]^3$. As was already discussed, the Lagrange multiplier $\lambda = P$ plays the role of the pressure in our incompressible Hamiltonian discretisation. The numerical velocity and pressure fields are compared against the exact solution during several wave periods. Figure 2.5 gives an example of the numerical solution during one period. Figure 2.6 shows that conservation of energy and discrete zero-divergence in time are maintained up to machine precision. To ensure that the velocity field has zero-divergence, one has to initialise the numerical scheme with an exact discrete divergence-free velocity field, see Section 4.3. Thus, adjustment of the initial projection of the velocity field onto the discontinuous Galerkin basis is required to satisfy this condition exactly (up to machine precision). The energy change observed at $t = 0$ in Figure 2.6, originates from this projection, and is within the order of accuracy of the numerical approximation. Table 2.3 presents the rate of convergence of the Hamiltonian DGFEM discretisation.

2.5.4 Poincaré waves in a channel

Poincaré waves in a channel for incompressible flow are considered next. The channel is periodic in the y-direction and closed with walls in the x- and z- directions. The angular rotation vector is equal to $\mathbf{\Omega} = (0,0,1)$. An



(a) Pressure profile at t=0 (b) Pressure profile at t=T/4

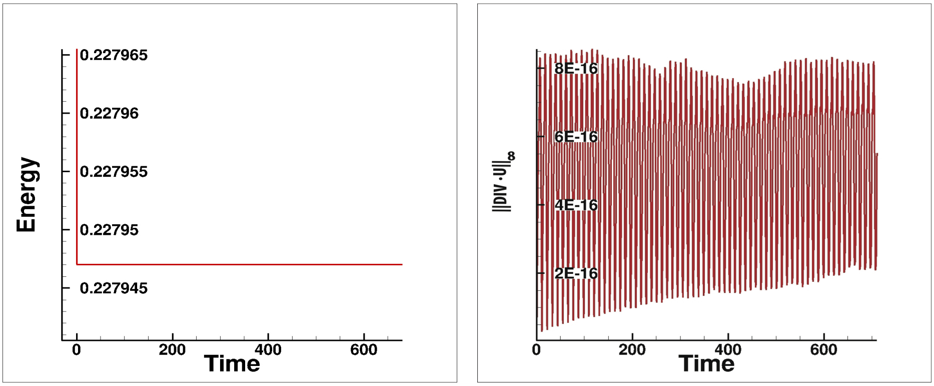


(c) Pressure profile at t=T/2 (d) Pressure profile at t=3T/4

Figure 2.5: Incompressible waves in periodic domain. The results concern an incompressible Hamiltonian discretisation on a $32 \times 32 \times 32$ grid with $\Delta t = T/20$ and period T. The implementation concerns a quadratic polynomial approximation on local elements.

Table 2.3: Convergence of the error in the Hamiltonian DGFEM discretisation for incompressible periodic waves in a cuboid. Due to symmetry all velocity components have the same error.

		$p = 0$		$p = 1$		$p = 2$	
Grid		l^2-error	order	l^2-error	order	l^2-error	order
4x4x4	**u**	2.665E-1	–	1.6340E-1	–	7.2122E-3	–
	p	2.872E-2	–	1.3876E-2	–	3.9242E-3	–
8x8x8	**u**	1.477E-1	0.9	5.3412E-2	1.6	9.6455E-4	2.9
	p	1.411E-2	1.0	4.6244E-3	1.6	6.0758E-4	2.7
16x16x16	**u**	7.587E-2	1.0	1.8100E-2	1.6	1.2843E-4	2.9
	p	7.141E-3	1.0	1.4475E-3	1.7	1.0251E-4	2.6
32x32x32	**u**	3.822E-2	1.0	5.7218E-3	1.7	1.6820E-5	2.9
	p	3.737E-3	0.9	4.5473E-4	1.7	1.443E-5	2.8
64x64x64	**u**	1.919E-2	1.0	1.6772E-3	1.8	2.2143E-6	2.9
	p	2.169E-3	0.8	1.3692E-4	1.7	2.0682E-6	2.8



(a) Energy function.

(b) Discrete divergence.

Figure 2.6: Energy and L_∞-norm of discrete divergence in the Hamiltonian DGFEM discretisation during 100 periods of periodic inertial waves in a cuboid.

exact solution for Poincaré waves in $D = [0,1]^3$ reads

$$u = -\frac{k\sigma^3}{1-\sigma^2}\left(1 + \frac{l^2}{(\sigma k)^2}\right)\sin(kx)\sin(ly - \sigma t)\cos(2\pi z), \tag{2.80a}$$

$$v = \left(-l\sigma\cos(kx) + \frac{1}{k}\sin(kx)\right)\cos(ly - \sigma t)\cos(2\pi z), \tag{2.80b}$$

$$w = \sigma\left(\cos(kx) + \frac{l}{\sigma k}\sin(kx)\right)\sin(ly - \sigma t)\sin(2\pi z), \tag{2.80c}$$

$$P = -\sigma^2\left(\cos(kx) + \frac{l}{\sigma k}\sin(kx)\right)\cos(ly - \sigma t)\cos(2\pi z), \tag{2.80d}$$

where $k = 2\pi$, $l = 2\pi$, and frequency $\sigma = 1/\sqrt{k^2 + l^2 + 1}$. At the solid walls in the x- and z-directions we need to satisfy geostrophic balance at the boundaries, due to the rotation of the domain. In Figures 2.7 and 2.8 we present a numerical solution (velocity vector and scalar pressure fields) during one period. Figure 2.9 demonstrates the discrete conservation of the energy and the zero-divergence of the discrete velocity. Convergence results are given in Table 2.4, which show that the convergence rates are close to $k+1$, with k the polynomial order.

2.5.5 Inertial waves

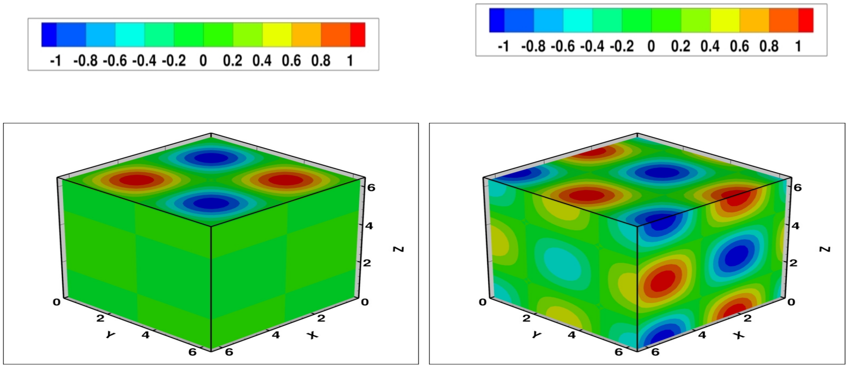
Next, we consider linear, incompressible, rotational fluid flow in a rectangular box with solid wall boundary conditions on all sides. Such kind of flow will lead to inertial waves in the interior of the domain, e.g., [62]. An extensive discussion of these waves as well as an improved semi-analytical solution of this problem can be found in [78]. The semi-analytical solution is used as a test solution for the verification of our incompressible Hamiltonian discretisation with slip-flow boundary conditions. Due to the slow convergence of the semi-analytical solutions, this comparison can, however, only be done for restricted mesh sizes.

Since the exact solution is unknown, we use solutions on a sequence of uniform meshes to obtain an estimate for the rate of convergence, which is called Richardson extrapolation (e.g., [103]). We take a uniformly refined sequence of meshes $h_4 = h_{\text{ref}} < h_3 < h_2 < h_1$, where the mesh-size h_i $(i = 1, 2, 3.4)$ is doubled for each finer mesh, and calculate the convergence rate s by numerically solving the following equation

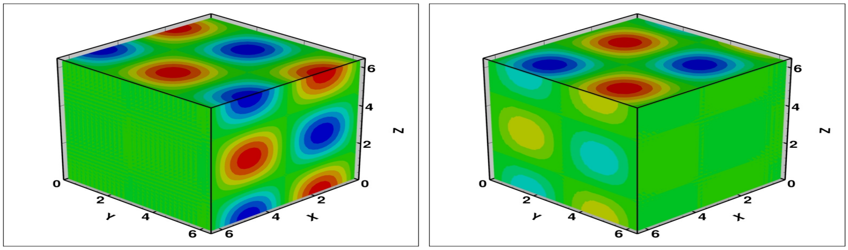
$$\frac{h_1^s - h_2^s}{h_2^s - h_3^s} = \frac{||\mathbf{U}_{\text{ref}} - \mathbf{U}_{h_1}|| - ||\mathbf{U}_{\text{ref}} - \mathbf{U}_{h_2}||}{||\mathbf{U}_{\text{ref}} - \mathbf{U}_{h_2}|| - ||\mathbf{U}_{\text{ref}} - \mathbf{U}_{h_3}||}. \tag{2.81}$$

Table 2.4: Convergence of the error in the Hamiltonian DGFEM discretisation for incompressible Poincaré waves in a channel.

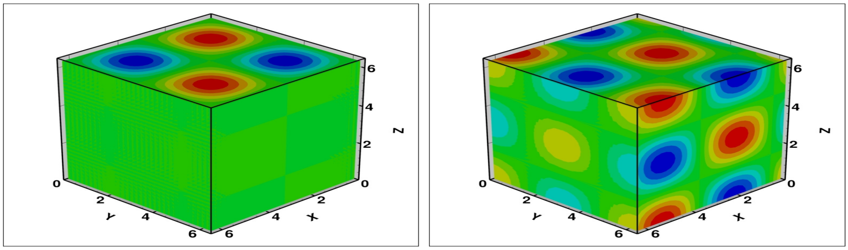
		$p = 0$		$p = 1$		$p = 2$	
Grid		l^2-error	order	l^2-error	order	l^2-error	order
4x4x4	u	7.834e+0	–	1.3599e+0	–	2.9126E-1	–
	v	7.492e+0	–	1.4793e+0	–	2.5108E-1	–
	w	8.935e+0	–	1.6662e+0	–	2.4933E-1	–
	p	5.819e+0	–	4.7072e+0	–	2.1281E-1	–
8x8x8	u	3.889e+0	1.0	5.7839E-1	1.3	2.8640E-2	3.3
	v	3.802e+0	1.0	6.1504E-1	1.3	2.0699E-2	3.6
	w	4.042e+0	1.1	6.7238E-1	1.3	2.0075E-2	3.6
	p	2.290e+0	1.3	9.1548E-1	2.3	2.9025E-2	2.9
16x16x16	u	2.192e+0	0.8	1.9858E-1	1.5	3.1792E-3	3.1
	v	2.229e+0	0.8	2.4017E-1	1.4	2.3963E-3	3.1
	w	2.015e+0	1.0	2.4244E-1	1.5	2.2733E-3	3.1
	p	1.179e+0	1.0	3.2671E-1	1.5	3.2460E-3	3.1
32x32x32	u	1.136e+0	0.9	6.3126E-2	1.6	4.1469E-4	2.9
	v	1.169e+0	0.9	8.4394E-2	1.5	3.2670E-4	2.9
	w	1.065e+0	0.9	8.5931E-2	1.5	3.1967E-4	2.8
	p	5.932E-1	1.0	1.0317E-1	1.7	4.3572E-4	2.9
64x64x64	u	5.726E-1	1.0	1.8031E-2	1.8	5.5452E-5	2.9
	v	5.899E-1	1.0	2.4461E-2	1.8	4.4548E-5	2.9
	w	5.258E-1	1.0	2.3687E-2	1.9	4.3963E-5	2.9
	p	2.961E-1	1.0	3.1627E-2	1.7	4.3572E-5	2.9



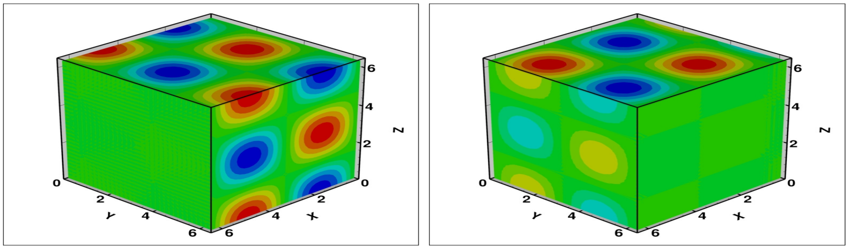
(a) u component at t=0 (b) v component at t=0



(c) u component at t=T/4 (d) v component at t=T/4

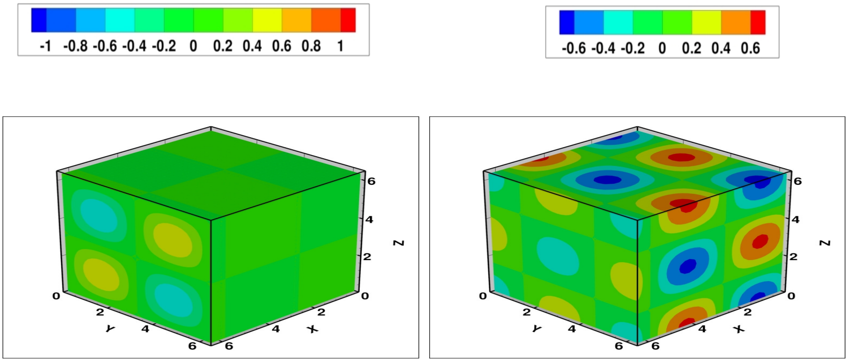


(e) u component at t=T/2 (f) v component at t=T/2

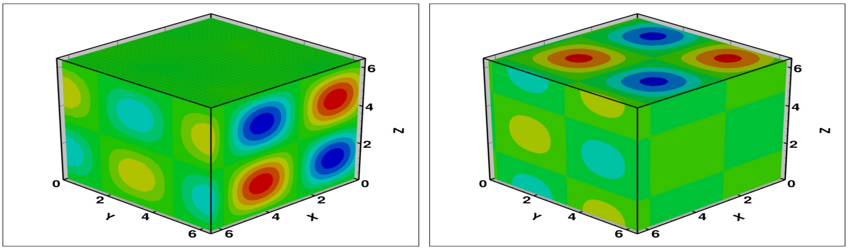


(g) u component at t=3T/4 (h) v component at t=3T/4

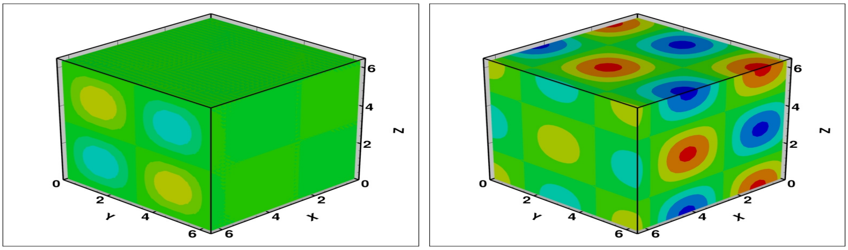
Figure 2.7: Velocity components u and v for Poincaré waves are computed. Numerical results concern the incompressible Hamiltonian DGFEM discretisation on a $32 \times 32 \times 32$ grid with $\Delta t = T/20$. We consider a quadratic polynomial approximation in local elements.



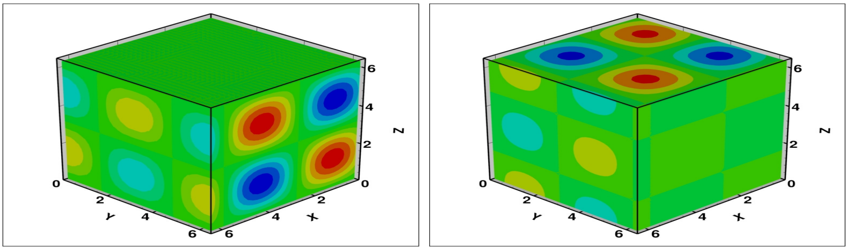
(a) w component at t=0 (b) p scalar pressure at t=0



(c) w component at t=T/4 (d) p scalar pressure at t=T/4

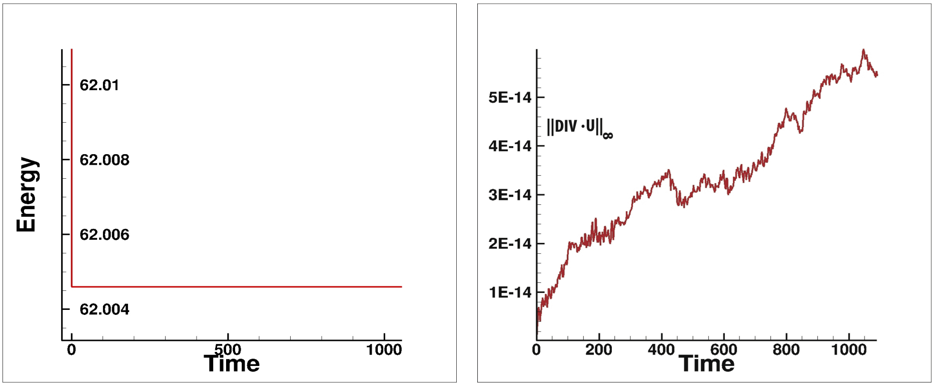


(e) w component at t=T/2 (f) p scalar pressure at t=T/2



(g) w component at t=3T/4 (h) p scalar pressure at t=3T/4

Figure 2.8: Vertical velocity component w and linearised scalar pressure fields during one period of a Poincaré-wave simulation. For details, see the caption of Figure 2.7.



(a) Energy function. (b) Discrete divergence.

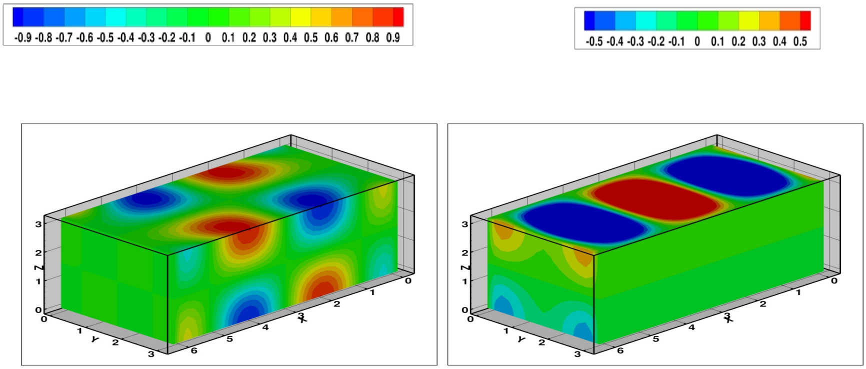
Figure 2.9: Energy and L_∞-norm of discrete divergence-free velocity field during 100 periods in Hamiltonian DGFEM computations of a Poincaré-wave.

The numerical velocity field and the mesh size for the different meshes are given with subscript notation, where $(\cdot)_{\text{ref}}$ denotes the finest mesh. By taking the sequence of meshes $64 \times 64 \times 64$, $32 \times 32 \times 32$, $16 \times 16 \times 16$, $8 \times 8 \times 8$, we numerically solve (2.81). The convergence rate in the L_∞-norm is roughly as expected, $s \approx 2.89$, for the implementation with quadratic polynomials.

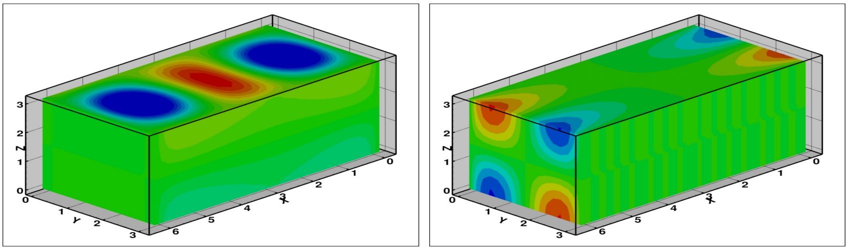
The extensive tests reported above convince us that the presented numerical scheme is actually more accurate than the slowly converging semi-analytical solutions. In Figures 2.10 and 2.11 we present all components of the numerical velocity vector and pressure fields produced by a simulation of incompressible fluid flow initialised with one of the eigenmodes of the semi-analytical solution. The domain is a rectangular box $D = [0, 2\pi] \times [0, \pi]^2$. Figure 2.12 shows conservation of energy and discrete zero-divergence.

2.5.6 Inertial waves in a ‘tilted’ box

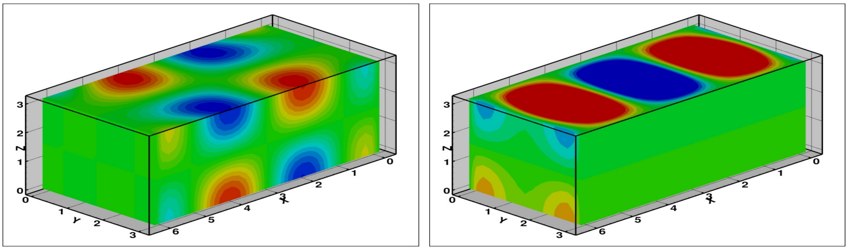
Finally, we attempt to observe wave focussing in our numerical wave tank, to demonstrate the capabilities of our novel numerical scheme. In the previous cases, the walls are either parallel or perpendicular to the rotation vector, possessing a “local reflectional symmetry”. Thus no mode breaking can be observed. However, a slight tilt in one of the walls results



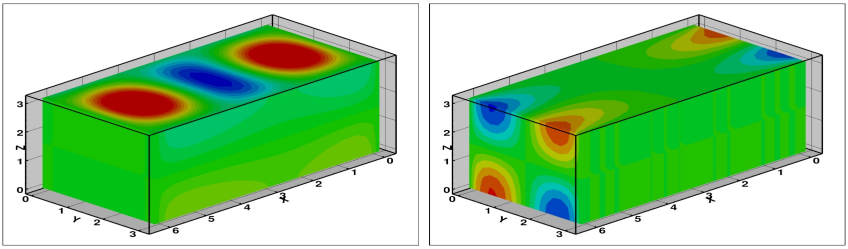
(a) u component at t=0 (b) v component at t=0



(c) u component at t=T/4 (d) v component at t=T/4

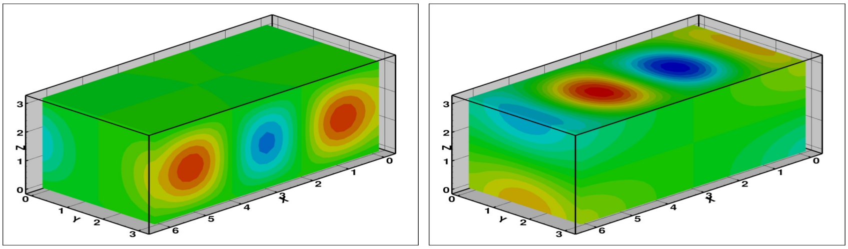


(e) u component at t=T/2 (f) v component at t=T/2

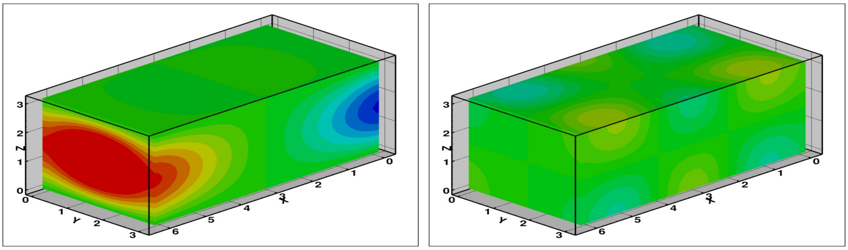


(g) u component at t=3T/4 (h) v component at t=3T/4

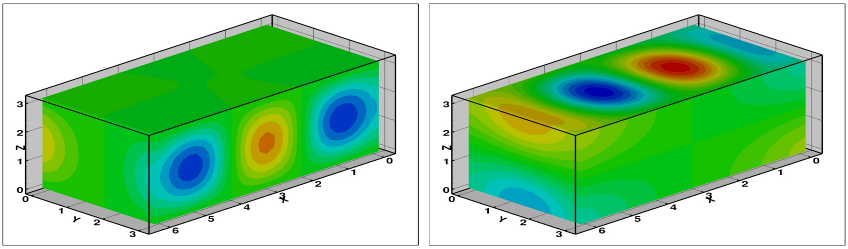
Figure 2.10: Horizontal velocity components u and v of an inertial wave with eigenfrequency $\sigma = 0.477$. The rotation vector is aligned with the z-direction. Quadratic basis functions are used on 32 x 16 x 16 mesh with time step $\Delta t = T/20$.



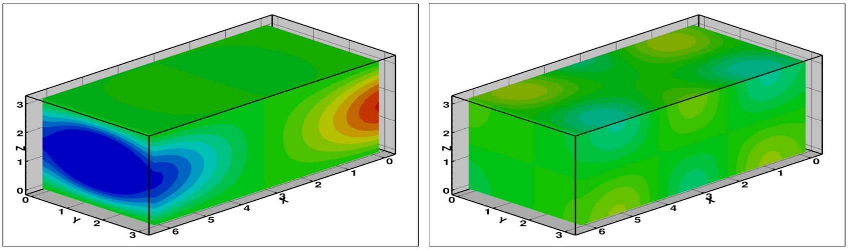
(a) w component at t=0 (b) p scalar pressure at t=0



(c) w component at t=T/4 (d) p scalar pressure at t=T/4

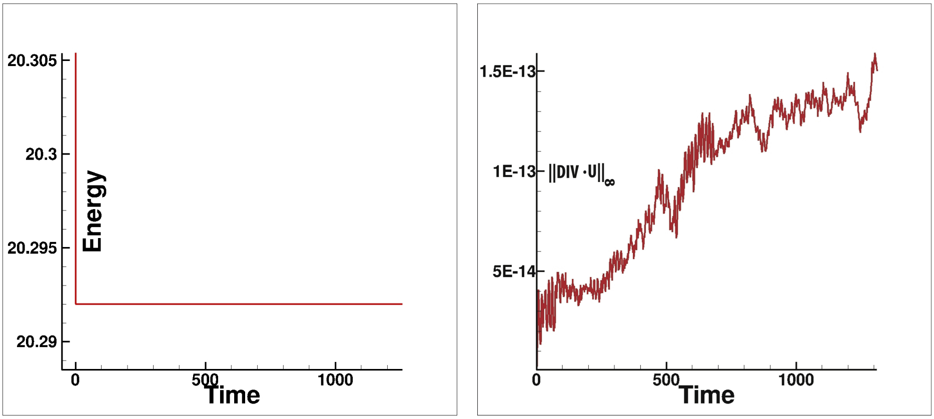


(e) w component at t=T/2 (f) p scalar pressure at t=T/2



(g) w component at t=3T/4 (h) p scalar pressure at t=3T/4

Figure 2.11: Vertical component w of velocity and scalar pressure p for the inertial wave simulation. For details, see the caption of Figure 2.10.



(a) Energy function.

(b) Discrete divergence.

Figure 2.12: Energy and L_∞-norm of discrete divergence-free velocity in Hamiltonian DGFEM discretisation during 100 time periods of inertial waves in a cuboid.

in symmetry breaking and hence in wave focussing and defocussing, such that, due to dominance of the former, wave attractors may appear [69]. Here we break the "local reflectional symmetry" by a small change in the background angular velocity vector.

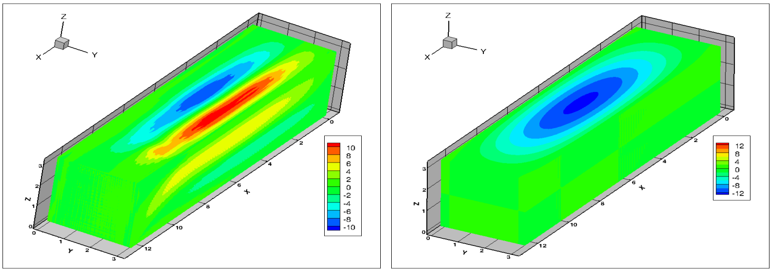
The domain is chosen to be a prolonged three-dimensional box with $D = [0, 4\pi] \times [0, \pi]^2$. The simulation is initialised with a particular eigenfrequency ($\sigma \approx 0.6946$) from the semi-analytical solution for a domain with a constant background rotation ($\Omega_* = (0, 0, 1)$) aligned along the z-direction available from [69, 78]. Next we introduce a small tilt in the angular velocity such that $\Omega = (0, 0.1, 1)$. After some time, the initial mode completely changes its structure and its frequency, due to the 'tilt'; see the plots of the 'tilted' and original flows in Figures 2.13 and 2.14. The distribution of the energy in the 'tilted' rectangular domain is given in Figure 2.15. In the vertical cross-section of the energy distribution plot (see Figure 2.15), a structure reminiscent of a wave attractor can be discerned. This rectangular region with a locally increased energy density is most evident in the middle of the tank, whereas in the rest of the tank there is a lower energy distribution. Furthermore, a similar rectangular structure can be seen in the pressure field, whereas the pressure field of the original 'untilted' mode has a regular structure (see Figures 2.13 and 2.14). The numerical solution is expanded in terms of quadratic polynomials defined on a mesh with $160 \times 40 \times 40$ elements. In the current implementation the size of the mesh is constrained by the available random-access memory. The latter obstacle can be overcome by applying local *hp*-refinement near the zones with high wave amplitude and implementing the numerical algorithm in a parallel environment.

2.6 Concluding remarks

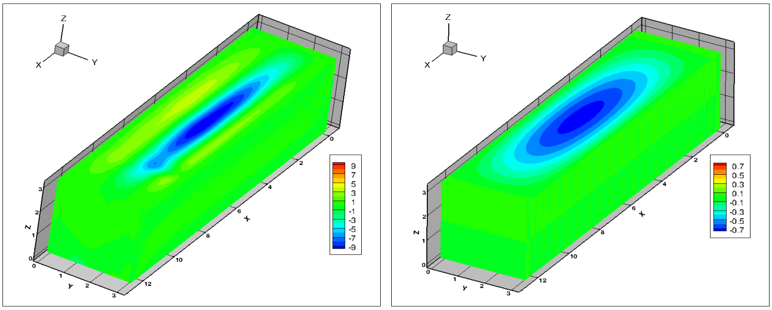
We have derived a DGFEM discretisation for Hamiltonian dynamics of linear, rotating incompressible fluid flow. The discretisation was obtained by applying Dirac's constrained Hamiltonian theory on a DGFEM formulation of compressible fluid flows. As an interim result a discretisation of Hamiltonian dynamics of compressible flow was derived, implemented and tested against exact solutions. The use of Dirac's theory is a novel approach to derive a Hamiltonian discontinuous Galerkin discretisation for incompressible flow using a related discretisation for compressible flow. The resulting system, as a consequence of the exact preservation of the constraints, does not require a stabilisation common to some direct DGFEM discretisations of incompressible fluid flows.

It was a challenge to derive and implement the boundary conditions for a discretisation preserving the Hamiltonian structure in a rotating frame, due to the mandatory satisfaction of geostrophic balance for the flow along fixed walls. Moreover, for exact preservation of energy and zero-divergence, the presented numerical scheme requires the projection of the initial velocity profile to be exactly divergence free at the discrete level. A preprocessing step was thus introduced to ensure that the initial velocity field in the DGFEM discretisation is divergence free up to machine precision.

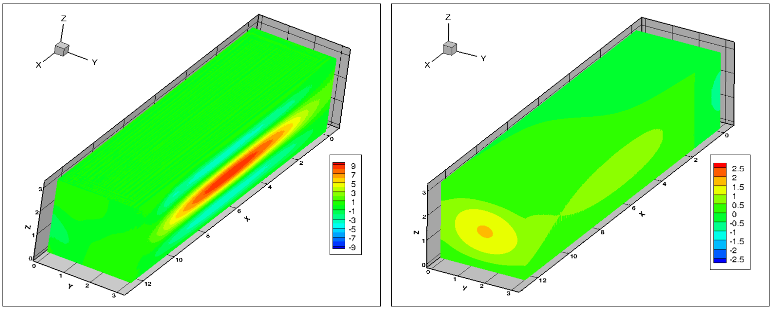
Several tests of inertial waves in rotating domains were presented. The simulation of Poincaré inertial waves in a channel assessed the proper implementation of no-normal flow boundary conditions in rotating domains. Next, an inertial-wave simulation in a cuboid with fixed solid walls showed agreement up to 10^{-2} with slowly converging semi-analytical solutions available from [62]. Richardson extrapolation with sequencing of meshes proved that our numerical solution is more accurate: the semi-analytical solutions available from [62] or [78] do either not satisfy the Euler equations or the solid-wall boundary conditions exactly. A DGFEM allows relatively easy *hp*-refinement of the system. Global *p*-refinement was already used in all presented numerical test cases and it appears that the quadratic polynomial approximation in the local elements provides sufficient order of



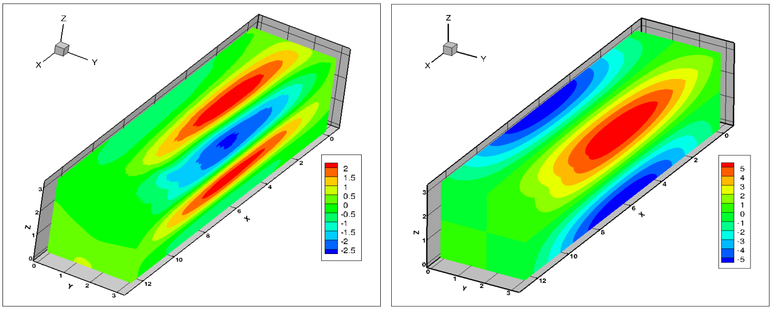
(a) u component (b) u component of exact mode



(c) v component (d) v component of exact mode

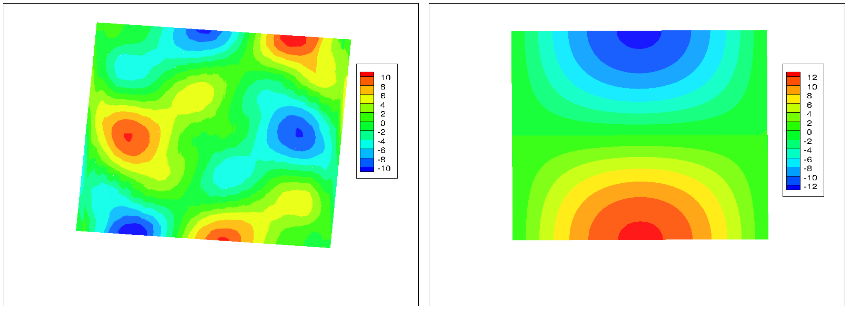


(e) w component (f) w component of exact mode

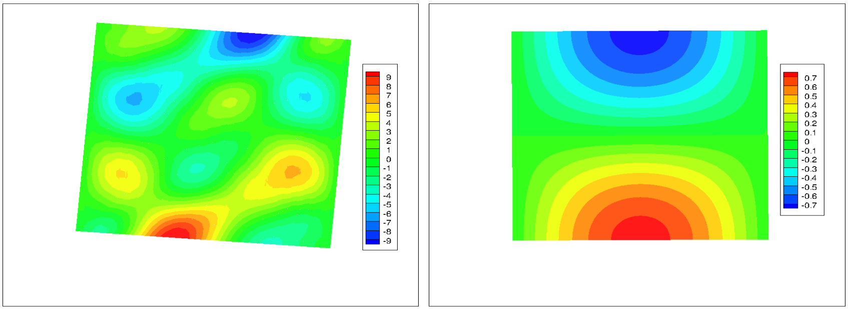


(g) p pressure (h) p pressure of exact mode

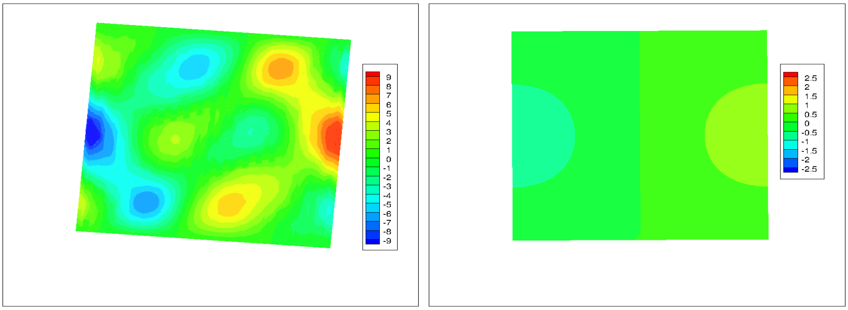
Figure 2.13: All components of the velocity field and the pressure field are given at time t=56.4. The first column concerns the 'tilted' simulation and the second column is the exact 'untilted' semi-analytical solution.



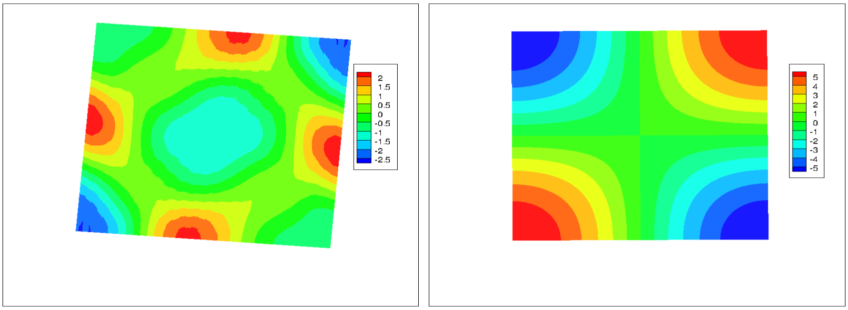
(a) u component (b) u component of the exact mode



(c) v component (d) v component of the exact mode

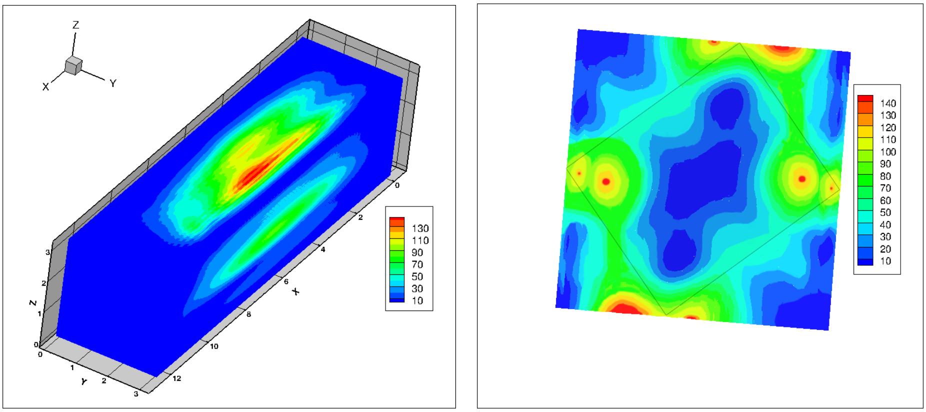


(e) w component (f) w component of the exact mode



(g) p component (h) p pressure of the exact mode

Figure 2.14: Vertical cross-sections of the fields given in Figure 2.13 in the middle of the tank.



(a) Three-dimensional energy plot. (b) Vertical cross-section in the middle.

Figure 2.15: The distribution of the numerical energy in the 'tilted' simulation at time t=56.4. In (b), we outlined the tentative attractor.

accuracy for capturing the phenomena of inertial waves in rotating domains.

Moreover, a simulation of inertial waves in a rotating domain with a small change in the direction and magnitude of the background angular velocity revealed a structure reminiscent of a wave attractor, due to the violation of the "local reflectional symmetry". To capture sufficient details of wave attractors, it is useful to introduce *hp*-refinement of the domain near the zones of attraction. This is one of the nice features of a DG scheme and will be addressed in future research. It will also allow more detailed studies of the wave focussing and defocussing in a container deprived of "local reflectional symmetry". Additional work will include the incorporation of a free surface in the presented numerical scheme for incompressible fluid flows.

2.7 Appendix

2.7.1 Constrained Hamiltonian continuum dynamics

The first calculation concerns the derivation from (2.19a) to (2.20)

$$\begin{aligned}0 =&\{\mathcal{F}[\rho],\mathcal{H}\} + \int_D \lambda_\Delta(\mathbf{x}')\{\mathcal{F}[\rho],\Delta(\mathbf{x}')\}d\mathbf{x}' \qquad (2.82)\\ =&-\int_D \frac{\delta\mathcal{F}}{\delta\rho}\nabla\cdot(\rho_0\mathbf{u})\,d\mathbf{x} - \iint_{D,D'}\lambda_\Delta(\mathbf{x})\frac{\delta\mathcal{F}}{\delta\rho(\mathbf{x}')}\nabla'\cdot\frac{\delta\nabla\cdot\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}'. \qquad (2.83)\end{aligned}$$

The last term in (2.83) can be reworked to

$$\begin{aligned}&-\iint_{D,D'}\lambda_\Delta(\mathbf{x})\frac{\delta\mathcal{F}}{\delta\rho(\mathbf{x}')}\nabla'\cdot\frac{\delta\nabla\cdot\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}'\\ =&\iint_{D,D'}\lambda_\Delta(\mathbf{x})\nabla'\frac{\delta\mathcal{F}}{\delta\rho(\mathbf{x}')}\cdot\frac{\delta\nabla\cdot\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}'\\ &\qquad -\int_D\int_{\partial D'}\lambda_\Delta(\mathbf{x})\frac{\delta\mathcal{F}}{\delta\rho(\mathbf{x}')}\hat{\mathbf{n}}'\cdot\frac{\delta\nabla\cdot\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}dS'd\mathbf{x}\\ =&-\int_D\nabla\lambda_\Delta\cdot\nabla\frac{\delta\mathcal{F}}{\delta\rho}\,d\mathbf{x} + \int_{\partial D}\int_{D'}\lambda_\Delta(\mathbf{x})\nabla'\frac{\delta\mathcal{F}}{\delta\rho(\mathbf{x}')}\hat{\mathbf{n}}\cdot\frac{\delta\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,dS\,d\mathbf{x}'\\ =&\int_D\frac{\delta\mathcal{F}}{\delta\rho}\nabla^2\lambda_\Delta\,d\mathbf{x} - \int_{\partial D}\frac{\delta\mathcal{F}}{\delta\rho}\nabla\lambda_\Delta\cdot\hat{\mathbf{n}}dS, \qquad (2.84)\end{aligned}$$

in which the first boundary term emerging is zero, and similarly the second boundary term emerging is zero, because

$$n_i'\partial_j\left(\frac{\delta u_j(\mathbf{x})}{\delta u_i(\mathbf{x}')}\right) = \delta_{ij}\partial_i\left(n_i'\frac{\delta u_i(\mathbf{x}')}{\delta u_i(\mathbf{x})}\right) = 0, \qquad (2.85)$$

since $\delta u_i(\mathbf{x})/\delta u_i(\mathbf{x}') = \delta u_i(\mathbf{x}')/\delta u_i(\mathbf{x}) = \delta(\mathbf{x}-\mathbf{x}')$, which follows by analysing $\delta u_i(\mathbf{x})$ and $\delta u_i(\mathbf{x}')$ as functionals, and $n_i\delta u_i = \delta(n_iu_i) = 0$ at ∂D. The index i denotes the velocity component and summation over repeated indices is understood. Recombining the above yields the result stated in the main text

$$0 = \int_D\frac{\delta\mathcal{F}}{\delta\rho}\big(-\nabla\cdot(\rho_0\mathbf{u}) + \nabla^2\lambda_\Delta\big)d\mathbf{x} - \int_{\partial D}\frac{\delta\mathcal{F}}{\delta\rho}\hat{\mathbf{n}}\cdot\nabla\lambda_\Delta d\mathbf{S}. \qquad (2.86)$$

Recall that the secondary constraint is $\nabla\cdot(\rho_0\mathbf{u}) = \rho_0\nabla\cdot\mathbf{u} = 0$. The solution of (2.86) is therefore

$$\lambda_\Delta = \text{constant}. \tag{2.87}$$

The second derivation concerns the step from (2.19b) to (2.23). From (2.22), it follows that

$$\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}} = -\nabla\frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}. \tag{2.88}$$

Detailed analysis of (2.19b) entails

$$\begin{aligned}
0 &= \{\mathcal{F}[\Delta],\mathcal{H}\} + \int_D \lambda_\rho(\mathbf{x}')\{\mathcal{F}[\Delta],\rho(\mathbf{x}')\}d\mathbf{x}' \\
&\qquad + \int_D \lambda_\Delta(\mathbf{x}')\{\mathcal{F}[\Delta],\Delta(\mathbf{x}')\}d\mathbf{x}' && (2.89)\\
&= \int_D 2\boldsymbol{\Omega}\times\mathbf{u}\cdot\nabla\frac{\delta\mathcal{F}}{\delta\Delta}\,d\mathbf{x} + \iint_{D,D'} \lambda_\rho(\mathbf{x})\nabla'\cdot\left(\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}(\mathbf{x}')}\right)\frac{\delta\rho(\mathbf{x})}{\delta\rho(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}' \\
&\qquad + \iint_{D,D'} \lambda_\Delta(\mathbf{x})\frac{2\boldsymbol{\Omega}}{\rho_0}\times\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}(\mathbf{x}')}\cdot\frac{\delta\nabla\cdot\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}'. && (2.90)
\end{aligned}$$

The third term in (2.90) can be shown to be zero as follows

$$\begin{aligned}
&\iint_{D,D'} \lambda_\Delta(\mathbf{x})\frac{2\boldsymbol{\Omega}}{\rho_0}\times\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}(\mathbf{x}')}\cdot\frac{\delta\nabla\cdot\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}' \\
&= -\iint_{D,D'} \nabla\lambda_\Delta(\mathbf{x})\cdot\frac{2\boldsymbol{\Omega}}{\rho_0}\times\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}(\mathbf{x}')}\cdot\frac{\delta\mathbf{u}(\mathbf{x})}{\delta\mathbf{u}(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}' \\
&+ \int_{D'}\int_{\partial D} \lambda_\Delta(\mathbf{x})\frac{2\boldsymbol{\Omega}}{\rho_0}\times\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}(\mathbf{x}'})\cdot\frac{\delta\left(\hat{\mathbf{n}}\cdot\mathbf{u}(\mathbf{x})\right)}{\delta\mathbf{u}(\mathbf{x}')}\,dS\,d\mathbf{x}' \\
&= 0, && (2.91)
\end{aligned}$$

since $\lambda_\Delta = \text{cst}$ (cf. (2.87)) and by using (2.85).

The second term in (2.90) can be reworked as follows

$$\begin{aligned}
&\iint_{D,D'} \lambda_\rho(\mathbf{x})\nabla'\cdot\left(\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}(\mathbf{x}')}\right)\frac{\delta\rho(\mathbf{x})}{\delta\rho(\mathbf{x}')}\,d\mathbf{x}\,d\mathbf{x}' \\
&= \int_D \lambda_\rho\nabla\cdot\left(\frac{\delta\mathcal{F}[\Delta]}{\delta\mathbf{u}}\right)d\mathbf{x} \\
&= \int_D \nabla\lambda_\rho\cdot\nabla\left(\frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}\right)d\mathbf{x} - \int_{\partial D}\lambda_\rho\hat{\mathbf{n}}\cdot\nabla\left(\frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}\right)dS \\
&= -\int_D \nabla^2\lambda_\rho\cdot\frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}\,d\mathbf{x} + \int_{\partial D}\nabla\lambda_\rho\cdot\hat{\mathbf{n}}\frac{\delta\mathcal{F}[\Delta]}{\delta\Delta}\,dS. && (2.92)
\end{aligned}$$

The second last boundary term $\hat{\mathbf{n}} \cdot \nabla(\delta\mathcal{F}/\delta\Delta)$ in (2.92) is zero because it is imposed as extra gauge or boundary condition, cf. an earlier remark.

When we combine (2.92) and (2.90), the final result (2.23) in the main text is reached

$$0 = -\int_D \frac{\delta\mathcal{F}}{\delta\Delta}\Big(\nabla^2\lambda_\rho + \nabla\cdot(2\boldsymbol{\Omega}\times\mathbf{u})\Big)\,d\mathbf{x} + \int_{\partial D}\frac{\delta\mathcal{F}}{\delta\Delta}\,\hat{\mathbf{n}}\cdot\big(\nabla\lambda_\rho + 2\boldsymbol{\Omega}\times\mathbf{u}\big)\,dS. \tag{2.93}$$

Chapter 3

New semi-analytical solution for inertial waves in a rectangular parallelepiped

A study of inertial gyroscopic waves in a rotating homogeneous fluid is undertaken both theoretically and numerically. A novel approach is presented to construct a semi-analytical solution of a linear three-dimensional fluid flow in a rotating rectangular parallelepiped bounded by solid walls. The three-dimensional solution is expanded in vertical modes to reduce the dynamics to the horizontal plane. On this horizontal plane the two dimensional solution is constructed via superposition of 'inertial' analogs of surface Poincaré and Kelvin waves reflecting from the walls. The infinite sum of inertial Poincaré waves has to cancel the normal flow of two inertial Kelvin waves near the boundaries. The wave system corresponding to every vertical mode results in an eigenvalue problem. Corresponding computations for rotationally modified surface gravity waves are in agreement with numerical values obtained by Taylor (1921), Rao (1966) and also, for inertial waves, by Maas (2003) upon truncation of an infinite matrix. The present approach enhances the currently available, structurally concise modal solution introduced by Maas (2003). In contrast to Maas' approach, our solution does not have any convergence issues in the interior and does not suffer from Gibbs phenomenon at the boundaries.

Additionally, an alternative finite element method is used to contrast these two semi-analytical solutions with a purely numerical one. The main differences are discussed for a particular example and one eigenfrequency.

3.1 Introduction

Fluid phenomena on Earth involve rotation to a greater or lesser extent. There are flows in which rotation is an absolutely essential factor. Waves that are appearing in a closed rotating container filled with a homogeneous fluid became a subject of interest to the scientific community at the end of the 19th and beginning of the 20th century. Taylor [101] derived and presented the first complete linear solutions (valid for any angular frequency) for free surface oscillations in a rotating rectangular parallelepiped. Before Taylor, Rayleigh [87] discussed the problem of the free tidal oscillations of a rectangular sea of uniform depth, when the vertical component of the Earth's rotation period is large compared with the periods of the oscillations. Later, the subject was studied by Proudman [84, 85], who also corrected some inaccuracies and errors in Rayleigh's works. A large amount of research has been focused on rotationally modified surface gravity waves as a major representative of the low-frequency waves in a homogeneous rotating fluid. In oceanography these are referred to as Poincaré and Kelvin waves, depending on whether the waves display a strictly sinusoidal or partially exponential spatial dependence (e.g.,[55]). However, there is a class of inertial (gyroscopic) waves that are possible within the interior of homogeneous rotating fluids. Unlike the surface gravity waves, they have their maximum displacement in the interior of the domain, vanish at the free or solid surface, and are not affected by gravity. The frequencies of these waves are below the inertial frequency, $f = 2\Omega$, of the rotating domain, and they exist solely due to restoring Coriolis forces. No gravity effects are present in contrast to the case for surface gravity waves. Pure inertial waves were initially discovered theoretically by Kelvin [51] in a cylindrical domain. The axial spheroid was the next geometry where the hyperbolic equation governing the flow was solved by exactly satisfying the no-normal flow boundary conditions; by Bryan [21]. Later, Maas [62] presented a semi-analytical structural solution in a rectangular parallelepiped with straight walls. Due to their symmetrical shape all three containers do not have any net focussing which inertial waves are otherwise prone to develop [61]. The axial spheroid has a symmetric structure, and thus compensates every reflected focussing wave with a reflected defocussing wave. In the case of an axial cylinder or a rectangular parallelepiped, the walls are either parallel or perpendicular to the rotation axis, therefore such walls possess a local reflectional symmetry. A simple tilt of one of the walls immediately results in symmetry breaking and hence in wave focussing and defocussing, such that due to dominance of the former, wave attractors may appear (e.g., [61]).

All the above mentioned theoretical solutions have also been observed experimentally. Inertial waves were identified in a rotating axial cylinder [34, 73, 65, 67, 53], in a slightly tilted free surface cylinder [102], in a sphere [4], in a tilted spheroid (tilt of its axis of rotation with respect to the axis of the cavity) [64, 106], in a (truncated) cone [12], in a rectangular parallelepiped [14, 54, 15] and in a trapezoid [61, 69].

From the theoretical point of view, the solutions for the inertial waves presented by Maas [62] have a precise structure (revealed by use of the so-called Proudman-Rao method). The no-normal flow boundary conditions are satisfied exactly, by construction. Nevertheless, the solution is practically unusable, due to its poor convergence and Gibbs phenomenon at the boundaries, as shown in Section 2. We substitute the solution back into the linear Euler equations governing the flow and calculate the residues. It appears that the residues of the momentum equations are not exactly zero, and moreover, their convergence to zero is very slow: with more than two hundred Fourier modes the residue still is only of order 10^{-1} of the maximum flow. The convergence of the residue is faster in the interior, in comparison to the boundaries, which is caused by the extra Gibbs phenomenon at the boundaries.

In this work, we present an enhanced solution for the free inertial waves of a rotating planar-rectangular parallelepiped, whose walls are parallel or perpendicular to the rotation axis. In Section 2, we are presenting a detailed description of a new algorithm for the construction of this solution. As in Maas [62], the three-dimensional solution is reduced to a two-dimensional one, by assuming a standing mode structure in the vertical direction. Thus, for every vertical mode, the problem reduces to the horizontal plane, where the techniques introduced by Taylor [101] for rotationally modified surface gravity waves are used. The solution is sought as a superposition of inertial Poincaré (IP) and inertial Kelvin (IK) waves in the rectangular parallelepiped, the strictly rotational internal counterparts of the rotationally modified external gravity waves [62]. Two IK waves are chosen as a 'base' or particular solutions for the flow in the horizontal plane. The IK waves are assumed to have no motion in the x-direction (u=0, v=$v(x, y)$), and thus they satisfy the no-normal flow boundary conditions at x-walls, x=constant, automatically. IK waves grow exponentially in the along-wall, y-direction and are thus useful only when two opposite walls exist, excluding the unbounded growth that appears on infinite planes. Near the y-walls the normal flow of these IK waves is compensated by addition of an infinite sum of IP waves. In other words, we are searching a superposition of IK and IP waves such that the flow governed by them will not be disturbed by the presence of the solid walls of the domain. The described algorithm (further called Taylor's method) results in an eigenvalue problem for an infinite matrix, the finite truncation of which identifies the eigenfrequencies. The present algorithm has better convergence behaviour in the interior than the Proudman-Rao approach given by Maas [62]: the residues of the momentum equations are exactly zero (up to machine precision). Results are discussed by comparing solutions of one particular eigenfrequency.

Nonetheless, the convergence of the solution is slow near the boundaries. Therefore, in Section 3 the problem is tackled purely from a numerical perspective, by implementing a FEM discretisation of the governing linear Euler equations on the horizontal plane. The calculated numerical eigenfrequencies exactly coincide with the results of the above discussed semi-analytical models. The differences between the numerical and the semi-analytical solutions are analysed for one particular eigenfrequency. The numerical results verify the new semi-analytical and FEM based methods. Conclusions are drawn in Section 4.

3.2 Semi-analytical inertial waves in a rectangular parallelepiped

3.2.1 3D-to-2D reduction of governing equations

We consider a wave-tank (rectangular parallelepiped) with solid body rotation. The wave-tank has fixed solid walls, is filled with an incompressible, homogenous fluid and is rotating about a vertical axis z_* with a constant angular velocity Ω_*, perpendicular to two of its side walls. Below, asterisks denote dimensional quantities. We closely follow the notation of [62]. Small-amplitude monochromatic waves appearing in the homogenous fluid on a rotating f_*-plane ($f_* = 2\Omega_*$) are governed by the linearised, inviscid equations of motion

$$\frac{\partial u_*}{\partial t_*} - f_* v_* = -\frac{1}{\rho_*}\frac{\partial p_*}{\partial x_*}, \tag{3.1a}$$

$$\frac{\partial v_*}{\partial t_*} + f_* u_* = -\frac{1}{\rho_*}\frac{\partial p_*}{\partial y_*}, \tag{3.1b}$$

$$\frac{\partial w_*}{\partial t_*} = -\frac{1}{\rho_*}\frac{\partial p_*}{\partial z_*}, \tag{3.1c}$$

$$\frac{\partial u_*}{\partial x_*} + \frac{\partial v_*}{\partial y_*} + \frac{\partial w_*}{\partial z_*} = 0, \tag{3.1d}$$

where (u_*, v_*, w_*) are the three-dimensional velocity components in the corresponding Cartesian directions (x_*, y_*, z_*), p_* is the linearised reduced pressure and ρ_* is the density.

As in Maas [62], we consider standing modes in the vertical direction to make w_* vanish at the rigid bottom $z_* = -H_*$ and top surface $z_* = 0$, i.e.,

$$w_* = \sum_{n=1}^{\infty} \frac{\partial \zeta_{n*}}{\partial t_*} \sin \frac{n\pi z_*}{H_*}, \tag{3.2a}$$

$$(u_*, v_*, p_*) = \sum_{n=1}^{\infty} (u_{n*}, v_{n*}, p_{n*}) \cos \frac{n\pi z_*}{H_*}, \tag{3.2b}$$

where subscript n refers to the nth vertical mode, and where we discarded the degenerate (geostrophic) mode with $n = 0$, and hence $w_* = 0$. The amplitude of the nth internal vertical elevation mode is denoted by ζ_{n*}. The presence of a solid wall at the top effectively eliminates the gravitational restoring forces and external gravity waves. Substitution of (3.2) into (3.1) modifies the governing equations in the following way

$$\frac{\partial u_{n*}}{\partial t_*} - f_* v_{n*} = -H_{n*} \frac{\partial^3 \zeta_{n*}}{\partial x_* \partial t_*^2}, \tag{3.3a}$$

$$\frac{\partial v_{n*}}{\partial t_*} + f_* u_{n*} = -H_{n*} \frac{\partial^3 \zeta_{n*}}{\partial y_* \partial t_*^2}, \tag{3.3b}$$

$$\frac{\partial \zeta_{n*}}{\partial t_*} + H_{n*} \left(\frac{\partial u_*}{\partial x_*} + \frac{\partial v_*}{\partial y_*} \right) = 0, \tag{3.3c}$$

with $H_{n*} \equiv H_*/(n\pi)$ and $p_{n*} = \rho_* H_{n*} (\partial^2 \zeta_{n*} / \partial t_*^2)$. For a specific nth vertical mode, (3.3) can be non-dimensionalised with H_{n*} and f_*^{-1} as length and time scales as follows $(u, v) = (u_{n*}, v_{n*})/(H_{n*} f_*)$: $\zeta = \zeta_{n*}/H_{n*}$, $t = t_* f_*$ and $(x, y) = (x_*, y_*)/H_{n*}$. The dimensionless version of system (3.1) is

$$\frac{\partial u}{\partial t} - v = -\frac{\partial^3 \zeta}{\partial x \partial t^2}, \tag{3.4a}$$

$$\frac{\partial v}{\partial t} + u = -\frac{\partial^3 \zeta}{\partial y \partial t^2}, \tag{3.4b}$$

$$\frac{\partial \zeta}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \tag{3.4c}$$

After substitution of $(u, v, \zeta) \propto \exp(-i\sigma t)$, system (3.4) becomes

$$(\Delta + \kappa^2)(u, v, \zeta) = 0, \tag{3.5}$$

where σ is the frequency of monochromatic waves, κ is defined by $\kappa^2 = 1/\sigma^2 - 1$ and the Laplacian is the following $\Delta = \partial_{xx} + \partial_{yy}$. Thus, the horizontal spatial structure of monochromatic waves is determined by the Helmholtz equation (3.5).

After choosing the Ansatz (3.2), the three-dimensional problem for resolving the flow (u, v, w, p) transforms into a two-dimensional (u, v, ζ) problem in the two-dimensional container, defined in the region $0 \leq x \leq L$, $-Y \leq y \leq Y$. Therefore, we are looking for a solution which will satisfy (3.5) with no-normal flow boundary conditions at the four walls: $u = 0$ at $x = 0$ and $x = L$, and $v = 0$ at $y = -Y$ and $y = Y$. Recall that, because of the mode (n) dependent scaling, the boundary sizes, $L = n\pi L_*/H_*$ and $Y = n\pi Y_*/H_*$ also depend on this mode number.

As was already mentioned before, the solution provided via the Proudman-Rao method in [62] satisfies the boundary conditions by construction, but suffers from poor convergence in the interior and Gibbs phenomenon at the boundaries. While some of the eigenfrequencies have indeed been obtained experimentally [14, 54], validating the precise shape of the corresponding eigenmodes has been more cumbersome [15]. This may of course be partially due to viscous boundary layers modifying the flow field near the boundaries, but may also partly be due to this convergence problem. The latter suggests that the development of a more precise method is desirable.

3.2.2 Taylor's method

We will present an alternative solution for determining the horizontal flow of the nth vertical mode in a rectangular parallelepiped. The method of Taylor is using a combination of results discussed in Maas [62] concerning inertial Poincaré waves and inertial Kelvin waves. Thus, the algorithm is applicable to semi-infinite as well as finite, rectangular regions. The idea is as follows: we search for analytic solutions of the Helmholtz equations (3.5), which do not or only partially satisfy impermeability conditions at the boundary, but the superposition of which will meet the requirements at the boundaries. To construct the final solution we use a combination of two IK and numerous IP waves defined in a finite meridional channel, available from [62]. The IK wave solution of (3.4) for a finite meridional channel is given by

$$u = 0, \tag{3.6a}$$

$$v = V\exp\left[-y + i\frac{x - L/2}{\sigma} - i\sigma t\right], \tag{3.6b}$$

$$\zeta = i\frac{v}{\sigma}. \tag{3.6c}$$

Obviously, the latter wave satisfies no-normal flow boundary condition only in the x-direction. Meanwhile, IP waves with quantised wave-numbers $\{k_m = m\pi/L\},\ m \in \{1, 2, 3, ...\}$ are given by

$$u = v_m i k_m \frac{\sigma^3}{1-\sigma^2}\left[1 + \left(\frac{l_m}{\sigma k_m}\right)^2\right] \sin k_m x\ \exp\ i(l_m y - \sigma t), \tag{3.7a}$$

$$v = v_m\left(-l_m \sigma \cos k_m x + \frac{1}{k_m}\sin k_m x\right)\ \exp\ i(l_m y - \sigma t), \tag{3.7b}$$

$$\zeta = v_m(\cos k_m x + \frac{l_m}{\sigma k_m}\sin k_m x)\ \exp\ i(l_m y - \sigma t), \tag{3.7c}$$

where $\mathbf{k_m} = (k_m, l_m)$ is the two-dimensional wave number vector and $l_m = \pm(\sigma^{-2} - \sigma_m^{-2})^{1/2}$ is determined by frequency σ, wave number k_m and $\sigma_m = (1 + k_m^2)^{-1/2}$. These waves, however, do not satisfy the boundary conditions $v = 0$ at $y = \pm Y$, but by contrast do satisfy $u = 0$ at $x = 0,\ L$.

After dropping the common factor $\exp(-i\sigma t)$, a solution of the Helmholtz equation (3.5), expressed in terms of the meridional velocity v, is now supposed to consist of two IK waves that are trapped at the walls $y = \pm Y$, and of an infinite sum of channel IP waves, a finite number of which are free to propagate in the y-direction:

$$\begin{aligned} 2v = {} & v_+\exp(i(x - L/2)/\sigma - y) + v_-\exp(-i(x - L/2)/\sigma + y) + \\ & \sum_{m=-\infty, m\neq 0}^{+\infty} v_m(is_m\sigma\cos k_m x + \frac{1}{k_m}\sin k_m x)\exp(s_m y). \end{aligned} \tag{3.8}$$

In view of the definition of σ_m, we find: $1 > \sigma_1 > \sigma_2 > \dots$. Now, if our frequency σ lies in between two of these eigenfrequencies: $\sigma_M > \sigma > \sigma_{M+1}$, then the s_m are imaginary for all $m \leq M$, while they are real for $m > M$. This implies that the first M modes have real l_m, with $l_m = -is_m$. Since l_m represent wave numbers, this implies M propagating modes, while all modes having $m > M$ are trapped (real s_m). For negative m we define $s_{-m} = -s_m$ and $k_{-m} = k_m$. Positive m refers either to energy propagation in positive y-direction, or trapping to $y = Y$ (depending on whether σ is larger or smaller than σ_m). Negative m refers to energy propagation in negative y-direction, or trapping to $y = -Y$. When we look for waves that are symmetric under reflection in the centre, $(x, y) \to (L - x, -y)$, the expression for v has to be invariant under this transformation. This requires $v_- = v_+ \equiv v_0$, $v_{2m} = -v_{-2m}$, $v_{2m+1} = v_{-(2m+1)}$. When we look for antisymmetric velocity fields v, these relations should reverse parity ($v_- = -v_+$ etc). Consider now v-symmetric solutions and adopt a Cartesian coordinate frame ξ, y, whose origin is at the centre of the rectangle, where

$$\xi = \frac{\pi x}{L} - \frac{\pi}{2}. \tag{3.9}$$

The container is now restricted to $\xi \in [-\pi/2, \pi/2]$. Then, with

$$\alpha \equiv \frac{L}{\pi\sigma}, \tag{3.10}$$

the v-velocity can be expressed

$$\begin{aligned} v = {} & \cosh y \cos \alpha\xi - i \sinh y \sin \alpha\xi + \\ & \frac{L}{\pi} \sum_{m\ odd}^{\infty} (-1)^{\frac{m-1}{2}} v_m \left(-i\frac{s_m}{\alpha} \sin m\xi \sinh s_m y + \frac{1}{m} \cos m\xi \cosh s_m y\right) + \\ & \frac{L}{\pi} \sum_{m\ even}^{\infty} (-1)^{\frac{m}{2}} v_m \left(i\frac{s_m}{\alpha} \cos m\xi \cosh s_m y + \frac{1}{m} \sin m\xi \sinh s_m y\right), \end{aligned} \tag{3.11}$$

where the amplitude v_0 has been arbitrarily set equal to one. Here and in the following all integers occurring in summations are strictly positive. Notice the point symmetry of (3.11) as it is invariant under the transformation $(\xi, y) \to (-\xi, -y)$. The IK waves (the first two terms in (3.11)) provide a coupling between the IP waves, but numbers α should be non-integer, as the IK-waves would otherwise be exactly annihilated by one of the IP-waves (having $s_m = 1$). Eigenfrequencies and amplitudes of each of the terms in (3.11) follow by rewriting this expression and by application of the boundary condition at $y = \pm Y$.

The IK waves, possessing non-integer α, yield only the cosine of even and the sine of odd multiples of ξ in their Fourier representations, as in the case of reflecting Kelvin waves [101]:

$$\cos\alpha\xi = \frac{4\alpha}{\pi}\sin\left(\frac{\alpha\pi}{2}\right)\left[\frac{1}{2\alpha^2} + \sum_{m\ even}\frac{(-1)^{\frac{m}{2}}}{\alpha^2 - m^2}\cos m\xi\right], \tag{3.12a}$$

$$\sin\alpha\xi = \frac{-4\alpha}{\pi}\cos\left(\frac{\alpha\pi}{2}\right)\sum_{m\ odd}\frac{(-1)^{\frac{m-1}{2}}}{\alpha^2 - m^2}\sin m\xi. \tag{3.12b}$$

These are therefore unable to match the cosine of odd and sine of even multiples of ξ, also occurring in (3.11). Now we directly expand the terms $\cos\alpha\xi$ and $\sin\alpha\xi$ in (3.11) using (3.12) in cosine of even and sine of odd multiples of ξ respectively, and require the coefficients of each of the trigonometric terms to vanish separately at $y = Y$. The same conditions are obtained by application of the boundary condition at the opposing boundary $y = -Y$. However, this direct approach yields an unwieldy matrix equation. Probably for this reason, Taylor extended (3.12a) with odd and (3.12b) with even multiplies ξ. The cosine expansion in (3.12a), for instance, is extended with $\cos s\xi$ (s odd), while each such term is counterbalanced by subtracting its even Fourier expansion (in another application of (3.12a) to odd integers such that in effects zeroes are added). Each such odd multiple and its counterbalancing Fourier expansion has an undetermined magnitude β_s (s odd). For the sine expansion similar terms are added yielding undetermined magnitude γ_s (s even). These undetermined magnitudes β_s and γ_s can be obtained from the requirement that the total velocity field v vanishes at

$y = \pm Y$. The total velocity at $y = Y$ reads:

$$
\begin{aligned}
v(\xi, Y) = {} & \frac{4\alpha}{\pi}\cosh Y \sin\left(\frac{\alpha\pi}{2}\right)\left[\frac{1}{2\alpha^2} + \sum_{m\ even}^{\infty}\frac{(-1)^{\frac{m}{2}}}{\alpha^2 - m^2}\cos m\xi \right. \\
& \left. + \sum_{s\ odd}^{\infty}\beta_s\left((-1)^{\frac{s-1}{2}}\frac{\pi}{4s}\cos s\xi - \left(\frac{1}{2s^2} + \sum_{j\ even}^{\infty}\frac{(-1)^{\frac{j}{2}}}{s^2 - j^2}\cos j\xi\right)\right)\right] \\
& + i\frac{4\alpha}{\pi}\sinh Y\cos\left(\frac{\alpha\pi}{2}\right) \\
& \left[\sum_{m\ odd}^{\infty}\frac{(-1)^{\frac{m-1}{2}}}{\alpha^2 - m^2}\sin m\xi + \sum_{s\ even}^{\infty}\gamma_s\left((-1)^{\frac{s}{2}}\frac{\pi}{4s}\sin s\xi + \sum_{j\ odd}^{\infty}\frac{(-1)^{\frac{j-1}{2}}}{s^2 - j^2}\sin j\xi\right)\right] \\
& + \frac{L}{\pi}\sum_{m\,odd}^{\infty}(-1)^{\frac{m-1}{2}}v_m\left(-i\frac{s_m}{\alpha}\sin m\xi\sinh s_mY + \frac{1}{m}\cos m\xi\cosh s_mY\right) \\
& + \frac{L}{\pi}\sum_{m\ even}^{\infty}(-1)^{\frac{m}{2}}v_m\left(i\frac{s_m}{\alpha}\cos m\xi\cosh s_mY + \frac{1}{m}\sin m\xi\sinh s_mY\right). \qquad (3.13)
\end{aligned}
$$

Vanishing of this expression requires the separate vanishing of the coefficients of $\cos m\xi$ or $\sin m\xi$. For m odd, vanishing of the coefficient of $\cos m\xi$ requires

$$
v_m = -\frac{\alpha\pi}{L}\frac{\cosh Y}{\cosh s_mY}\sin\left(\frac{\alpha\pi}{2}\right)\beta_m. \qquad (3.14)
$$

On the other hand, vanishing of the coefficient of $\sin m\xi$ requires

$$
v_m = \frac{4\alpha^2}{s_mL}\frac{\sinh Y}{\sinh s_mY}\cos\left(\frac{\alpha\pi}{2}\right)\left[\frac{1}{\alpha^2 - m^2} - \sum_{s\ even}^{\infty}\frac{\gamma_s}{m^2 - s^2}\right]. \qquad (3.15)
$$

The equality of these two expressions for v_m determines β_m in terms of γ_s,

$$
\frac{1}{m^2 - \alpha^2} + \sum_{s\ even}^{\infty}\frac{\gamma_s}{m^2 - s^2} + \beta_m\lambda_m = 0, \qquad (3.16)
$$

where

$$
\lambda_m = -\frac{s_m\pi}{4\alpha}\coth Y\tan\left(\frac{\alpha\pi}{2}\right)\tanh(s_mY). \qquad (3.17)
$$

For m even, we similarly find that vanishing of the coefficient of $\sin m\xi$ requires

$$v_m = -i\frac{\alpha\pi}{L}\frac{\sinh Y}{\sinh s_m Y}\cos\left(\frac{\alpha\pi}{2}\right)\gamma_m. \tag{3.18}$$

Dividing this by the expression for v_m obtained from vanishing of the coefficient of $\cos m\xi$ one finds

$$-\frac{1}{m^2-\alpha^2} + \sum_{s\ odd}^{\infty}\frac{\beta_s}{m^2-s^2} + \gamma_m\mu_m = 0, \tag{3.19}$$

where

$$\mu_m = \frac{s_m\pi}{4\alpha}\tanh Y\cot\left(\frac{\alpha\pi}{2}\right)\coth(s_m Y). \tag{3.20}$$

If we define $\mu_0 = 0$, (3.19) also includes, for $m = 0$, the extra requirement that the constant term in (3.13) vanishes. Note that in the expressions for λ_m and μ_m, s_m occurs in product with either $\tanh s_m y$ or $\coth s_m y$. The replacement $s_m = il_m$ when $\sigma < \sigma_m$, and the property $\tanh iz = i\tanh z$ (for real z), therefore leaves these expressions real, regardless of whether s_m is real or imaginary, making a cumbersome sign-distinction (as in Taylor [101]) unnecessary. It also guarantees λ_m and μ_m (and therefore the eigenfrequencies) to be real. Interpreting each first term of (3.16) and (3.19) as multiplying a quantity $\gamma_0(=1)$, the set of equations forms a system of infinite matrix equations $Ac = 0$, where A is as follows

$$A = \begin{pmatrix} \frac{1}{\alpha^2} & \frac{-1}{1^2} & 0 & \frac{-1}{3^2} & 0 & \cdots \\ \frac{1}{1^2-\alpha^2} & \lambda_1 & \frac{1}{1^2-2^2} & 0 & \frac{1}{1^2-4^2} & \cdots \\ \frac{-1}{2^2-\alpha^2} & \frac{1}{2^2-1^2} & \mu_2 & \frac{1}{2^2-3^2} & 0 & \cdots \\ \frac{1}{3^2-\alpha^2} & 0 & \frac{1}{3^2-2^2} & \lambda_3 & \frac{1}{3^2-4^2} & \cdots \\ \cdot & & & & & \\ \cdot & & & & & \\ \cdot & & & & & \end{pmatrix}, \tag{3.21}$$

and column vector $c = (\gamma_0, \beta_1, \gamma_2, \beta_3, ...)^T$, where superscript T means transpose. Matrix A is twice as big as the matrix obtained when directly expanding the terms in (3.11), alluded to above, but is much simpler to handle. Apart from differences in the definitions of the diagonal terms λ_m and μ_m, together with the fact that our α is a function of frequency, (3.10), the matrix equation exactly conforms with Taylor's. Nontrivial solutions result only when its determinant vanishes, $detA = 0$, which (through α) can be regarded as an equation establishing eigenfrequencies σ. Amplitudes β_m (m is odd) and γ_m (m is even), for any particular σ_j, $j = 1, 2...$ can be determined by the inversion of the reduced matrix that can be obtained from (3.16) and (3.19), where we now exclude the first row, corresponding to $m = 0$, and bring the first column to the right of the equation, which can now be regarded as known. With these amplitudes, from (3.15) and (3.18), also the amplitudes v_m in the expansion of v, Eqn. 3.11, are determined, and the solution is in essence complete. We also note that similar expressions can be obtained for antisymmetric solutions, where

$$\lambda_m = -\frac{s_m\pi}{4\alpha}\tanh Y \tan\left(\frac{\alpha\pi}{2}\right)\coth(s_mY), \tag{3.22a}$$

$$\mu_m = \frac{s_m\pi}{4\alpha}\coth Y \cot\left(\frac{\alpha\pi}{2}\right)\tanh(s_mY), \tag{3.22b}$$

and making similar replacements, $\sinh Y \iff -\cosh Y$ and $\sinh s_mY \iff \cosh s_mY$, in coefficients v_m and again, but with regards to the y-dependence, in fields u, v and ζ.

For each eigenvalue and corresponding set of amplitudes v_m, the velocity and elevation fields are now determined. Hence, the u velocity component of infinite sum of IP waves (3.7), with their corresponding eigenvalue and set of amplitudes v_m, reads

$$\begin{aligned} u = &\sum_{m\ odd}^{\infty} i(-1)^{\frac{m-1}{2}} v_m \left(\frac{\alpha}{m} - \frac{m}{\alpha}\right)\cos m\xi \cosh s_my \\ &+ \sum_{m\ even}^{\infty} i(-1)^{\frac{m}{2}} v_m \left(\frac{\alpha}{m} - \frac{m}{\alpha}\right)\sin m\xi \sinh s_my. \end{aligned} \tag{3.23}$$

The vertical elevation field then follows from the continuity equation

$$\begin{aligned} \zeta = &-\sigma^{-1}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = -i\sigma^{-1}(\pi L^{-1}u_\xi + v_y) = \\ &-\sigma^{-1}(i\sinh y\cos\alpha\xi + \cosh y \sin\alpha\xi) \\ &- \sum_{m\ odd}^{\infty} i(-1)^{\frac{m-1}{2}} v_m \left(\sin m\xi\cosh s_my + \frac{i\alpha s_m}{m}\cos m\xi \sinh s_my\right) \\ &+ \sum_{m\ even}^{\infty} i(-1)^{\frac{m}{2}} v_m \left(\cos m\xi \sinh s_my - \frac{i\alpha s_m}{m}\sin m\xi\cosh s_my\right). \end{aligned} \tag{3.24}$$

The eigenfrequencies are determined by truncating the infinite matrix to include just N rows and columns. Finding the roots of the resulting determinant numerically, and observing convergence of these roots upon increase of the number of rows, the set of eigenfrequencies σ_j, $j = 1, 2...$ is determined (approximately).

3.2.3 Comparison of two methods: Taylor's method vs. Proudman-Rao method

Next, this novel Taylor's method for a construction of semi-analytical solutions for the linearised Euler equations in a rectangular parallelepiped is tested and verified from a numerical perspective. First a few eigenfrequencies are determined in a given $[\pi \times 2\pi]$ rectangle, see Figure 3.1. Also, an independent verification is performed by comparing the latter eigenfrequencies to the frequencies for a $[2\pi \times \pi]$ rectangle in Figure 3.1. The eigenfrequency of the first symmetric mode is $\sigma_1^* = L/\pi\alpha_1 \approx 0.657$, which corresponds to the first root in Figure 3.1a, at $\alpha_1 \approx 1.522$. The next two eigenfrequencies of symmetric velocity modes are $\sigma_2^* = L/\pi\alpha_1 \approx 0.477$, $\alpha_2 \approx 2.095$ and $\sigma_3^* = L/\pi\alpha_1 \approx 0.398$, $\alpha_3 \approx 2.513$. Numerical computations for the frequencies and modal structures are assessed for several antisymmetric and symmetric modes. The eigenfrequencies and eigenmodes computed in this manner for the rotationally modified surface gravity modes are in agreement with numerical values obtained many years ago by Taylor [101] and later by Rao [86] and for the inertial modes with those computed in [62]. It is also worth to mention that the Proudman-Rao method is much faster in converging to the modal eigenfrequencies compared to Taylor's method.

Furthermore, close agreement in the eigenfrequencies of the modes enables a comparison between corresponding velocity fields of semi-analytical inertial waves constructed by the present Taylor's method to the previously employed Proudman-Rao method. The semi-analytical solutions are assessed according to the following criteria: (i) satisfaction of governing equations; (ii) satisfaction of the boundary conditions; and, (iii) speed of convergence. In both cases, solutions have the same standing mode structure in the vertical z-direction. Thus, the comparison is performed only on the horizontal plane. For a fair comparison in the calculations the same number of 'basis functions' are used: IP waves in Taylor's method and Fourier modes in the Proudman-Rao method.

The comparison is performed on a $[2\pi \times \pi]$ domain for the highest mode $\sigma \approx 0.657$ of the antisymmetric velocity field. In Figure 3.2 the horizon-

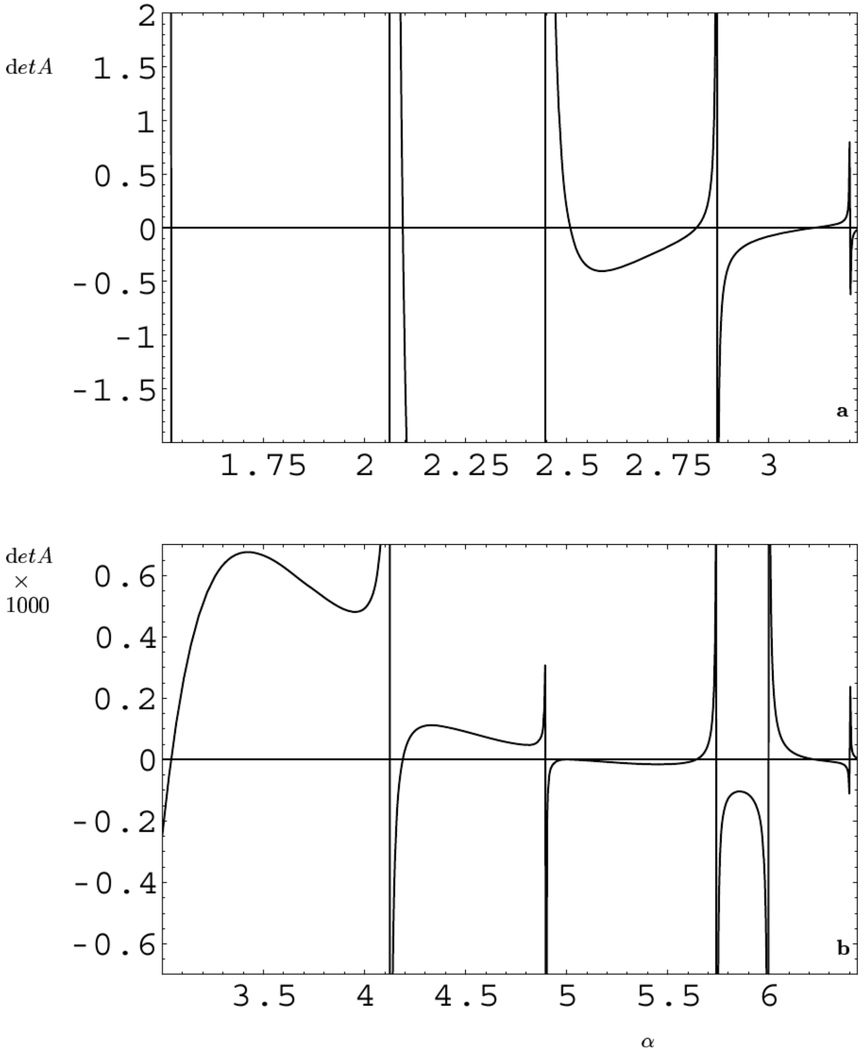
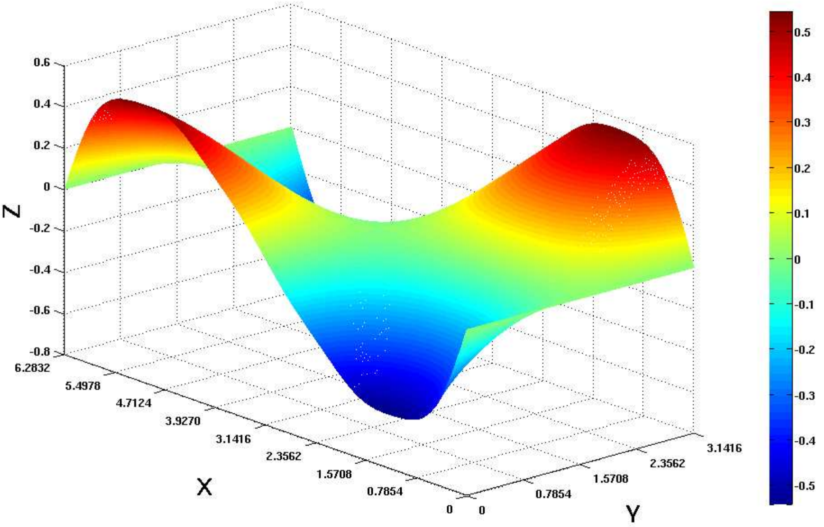
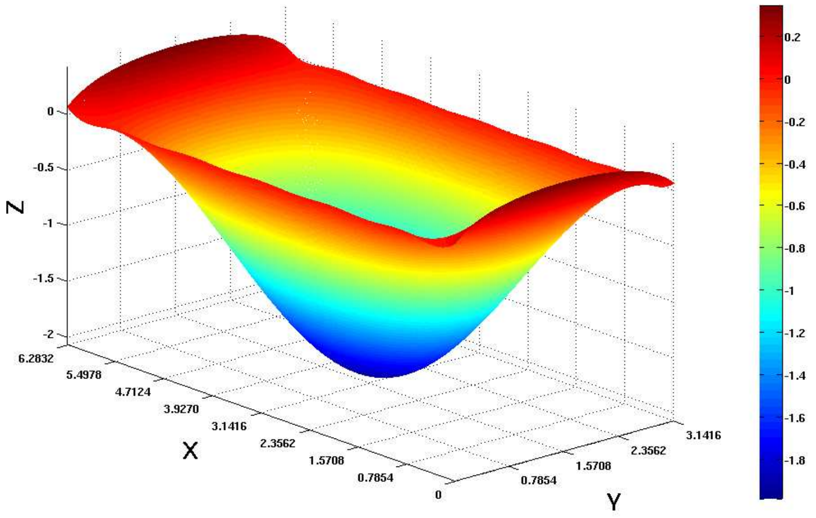


Figure 3.1: (a) Det A as a function of α for $L = Y = \pi$ for a mode whose v-velocity is symmetric. Hence, $L \times 2Y = \pi \times 2\pi$. Intersections with the horizontal line give the eigenfrequencies. The vertical lines represent asymptotes whose intersections with the horizontal axis should be disregarded. (b) Det A (for clarity multiplied by 10^3) as a function of α for $L = 2\pi$, $Y = \pi/2$, which represents a $2\pi \times \pi$ rectangle. Since the present rectangle has a width L twice that in (a), and $\sigma = L/\pi\alpha$, eigenfrequencies will be the same when the zeros are now found at α's twice that obtained in (a), which we verify by inspection.

tal u and v velocity fields constructed via Taylor's method are presented. For the first comparison the constructed modal solutions were substituted into the linear Euler equations and the residues are calculated. Obviously, due to the construction, in both algorithms the continuity equation and third momentum equation are satisfied exactly. Thus, in Figure 3.3 and Figure 3.4 only the residues for the u and v momentum equations are presented. It is apparent from Figure 3.3 that the Proudman-Rao method performs very poorly near the boundaries, which is explained by the Gibbs phenomenon. Also, the convergence is poor in the interior of the domain. By contrast, Figure 3.4 shows that in Taylor's method the rotating Euler equations are satisfied up to machine precision. The latter indicates that Taylor's method is preferable to Proudman-Rao, in this context. The differences between the constructed solutions are given in Figure 3.5. According to Figure 3.5 the main differences between the solutions are near the boundaries (y-wall for the u and x-wall for the v-component of the velocity). Near these boundaries the Proudman-Rao method suffers from a Gibbs phenomenon. In spite of its advantages, Taylor's method still has its flaws. The essence of the method is in 'filling' the solution with numerous IP-waves, to compensate the IK-waves motion near the boundaries in order to satisfy the no-normal flow boundary conditions. Thus, the more IP-waves are taken, the less no-normal flow will be registered near the appropriate boundaries. As can be noticed from Figure 3.2, the convergence is poor: non-zero flow is present near the x-boundary for the v-component of the velocity field and similarly near the y-boundary for the u-component. The nature of the convergence near the boundaries can be observed more closely in Figure 3.6. It shows that when the number of IP waves exceeds 100, the curve nearly stops decreasing. The latter suggests that despite the satisfaction of the Euler equations up to machine precision, the velocities constructed by Taylor's method, are not satisfying the boundary conditions exactly. Therefore, Taylor's method fails the second comparison test, namely satisfaction of the boundary conditions; whereas the satisfaction of the boundary conditions in the Proudman-Rao method was embedded in the construction. Due to the problems mentioned both methods globally fail the third assessment criterion. Nevertheless, it is worth to mention that the Proudman-Rao method is faster in convergence of the eigenfrequencies, whereas convergence of the velocity field of Taylor's method is high in the interior of the domain, but quite slow at the boundaries. Finally, if one of these two semi-analytical solutions has to be chosen, it remains which property is preferred: satisfaction of boundary conditions (Proudman-Rao method) or satisfaction of governing equations (Taylor's method) up-to machine precision.

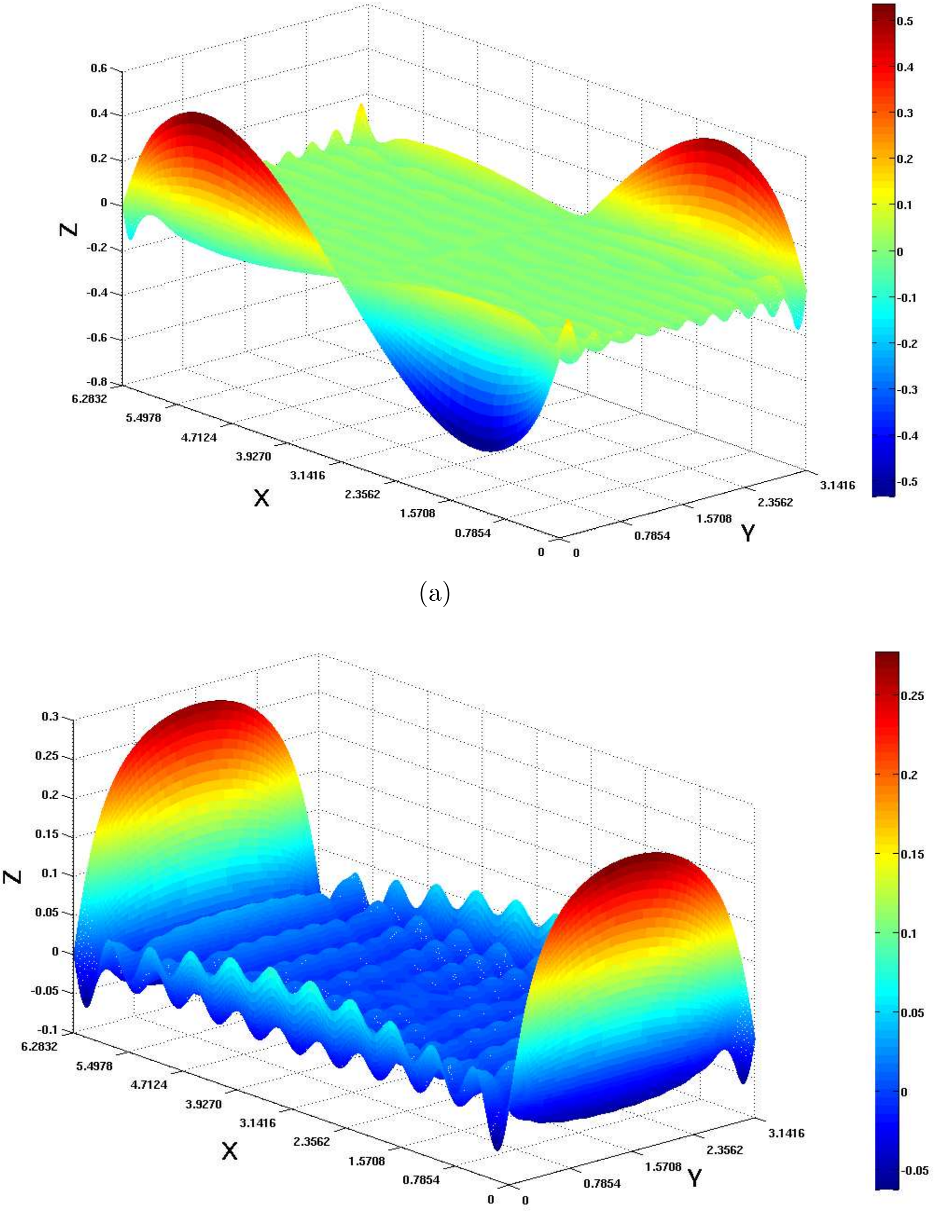


(a) u profile



(b) v profile

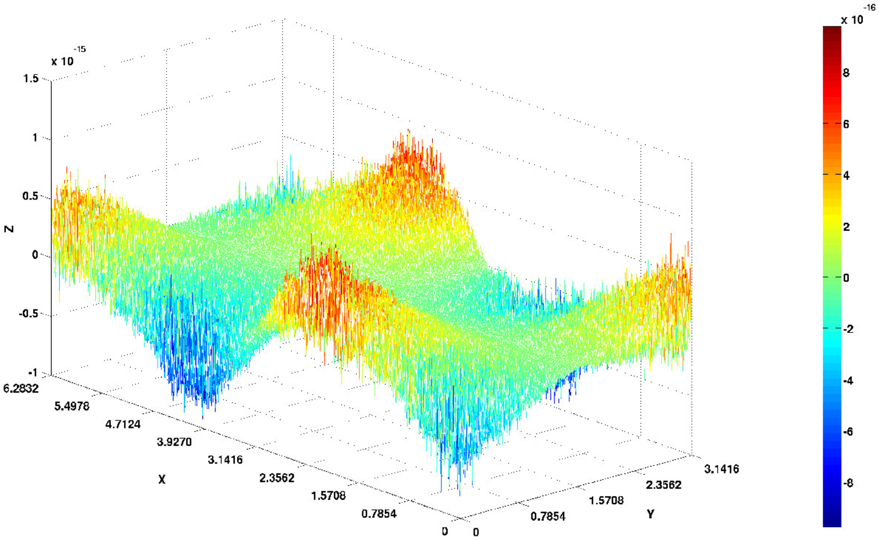
Figure 3.2: Antisymmetric horizontal velocity fields on the $x - y$ plane for mode $\sigma \approx 0.657$ constructed via Taylor's method, (a) and (b) subfigures are $u(x, y)$ and $v(x, y)$, respectively. The domain is rotating anti-clockwise and 20 IP waves were used.



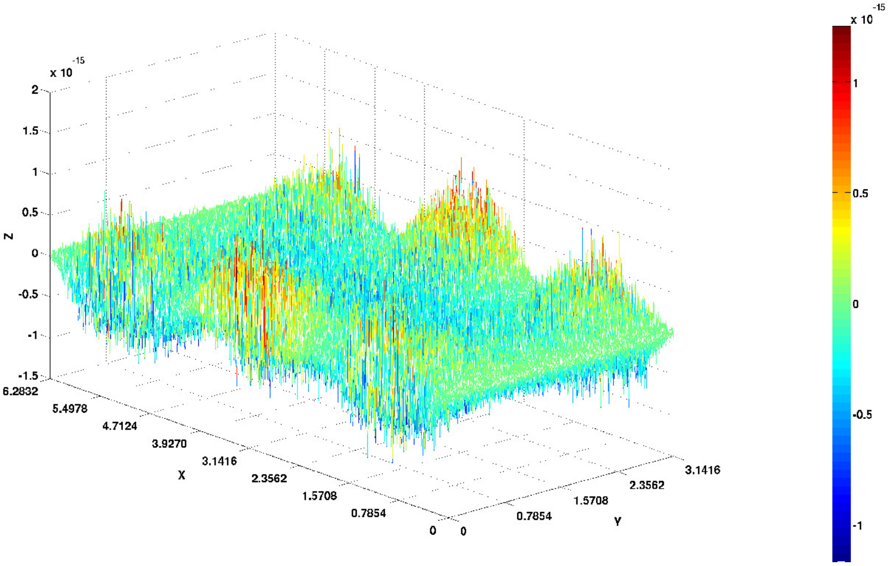
(a)

(b)

Figure 3.3: Antisymmetric horizontal velocity fields in the x–y plane for a mode with frequency sigma $\sigma \approx 0.657$ were substituted in the linearised Euler equations. (a) and (b) subfigures are the residues for velocities u and v in their respective momentum equations, as produced by the Proudman-Rao method.

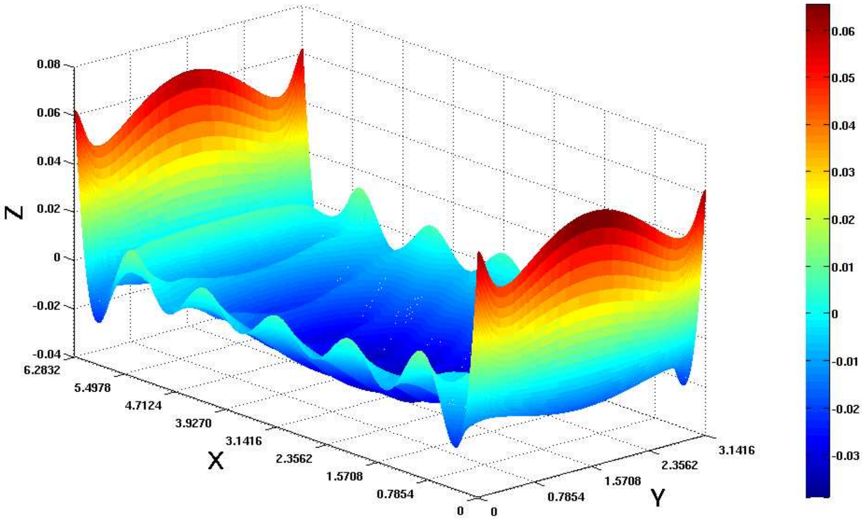


(a)

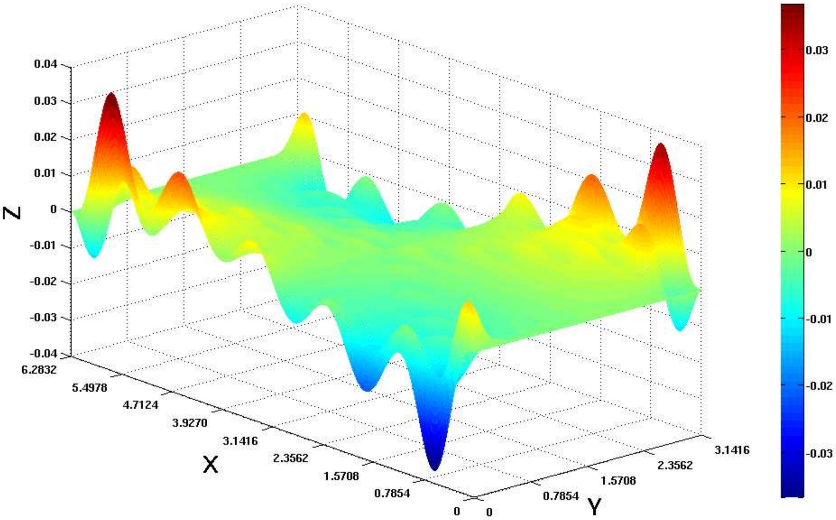


(b)

Figure 3.4: As Figure 3.3 using Taylor's method.



(a)



(b)

Figure 3.5: Antisymmetric horizontal velocity fields in the x–y plane for a mode with frequency sigma $\sigma \approx 0.657$ are constructed with two alternative algorithms. (a) and (b) subfigures give the differences between the results produced by the Proudman-Rao and Taylor's methods for the u and v velocity components, respectively.

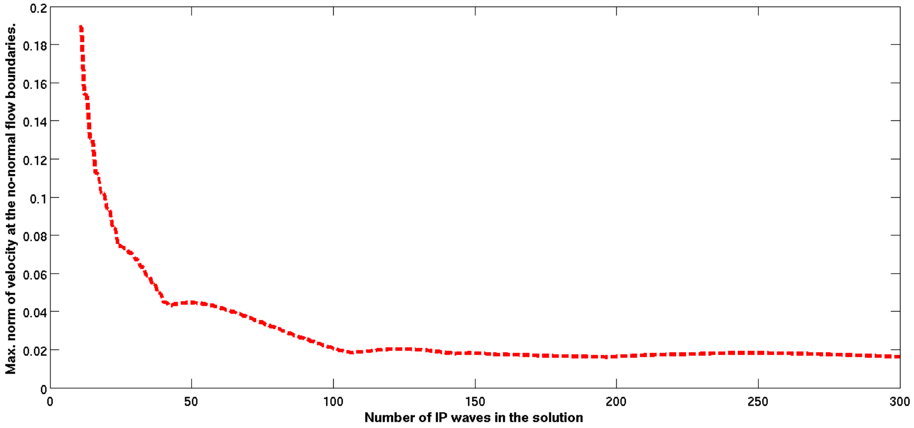


Figure 3.6: The maximum (numerical) speed v normal to the East-West boundaries is plotted against the number of IP waves used in Taylor's method.

3.3 FEM solution of linear inertial waves

In the previous section, it has been shown that both semi-analytical solutions for linear inertial waves have at least one major disadvantage. Therefore, next we present a purely numerical FEM solution for the inertial wave problem. First, the same vertical standing mode decomposition is considered, to reduce the inertial wave problem from 3D to 2D. Second, a FEM discretisation of the 2D boundary value problem given by the system (3.4) is constructed, with the mandatory satisfaction of the no-normal flow boundary conditions.

3.3.1 Weak formulation and resulting eigenvalue problem

A 2D boundary value problem is given on a rectangular domain $\Omega = \{0 \leq x \leq L; -Y \leq y \leq Y\}$ by the following partial differential

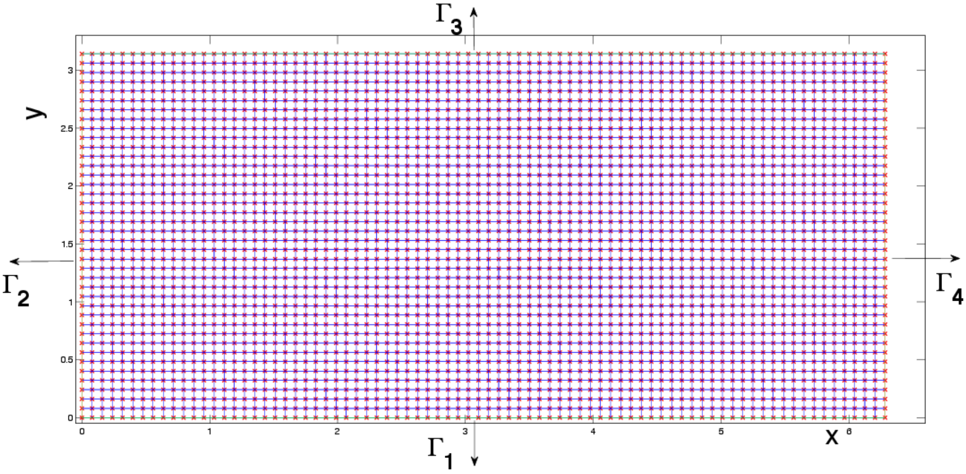


Figure 3.7: Tessellation $\mathcal{I}_h$ of the domain Ω with its four boundaries $\Gamma_1, \ldots, \Gamma_4$.

equations

$$\frac{\partial u}{\partial t} - v = -\partial_x p, \tag{3.25a}$$

$$\frac{\partial v}{\partial t} + u = -\partial_y p, \tag{3.25b}$$

$$-q + \partial_x u + \partial_y v = 0 \tag{3.25c}$$

$$\frac{\partial q}{\partial t} = -p \tag{3.25d}$$

and no-flow boundary conditions at the walls $\partial\Omega = \cup\Gamma_i$. The latter system is derived from (3.4), when the pressure is taken to be $p = \partial^2\zeta/\partial t^2$. The system introduced above is energy conserving, where the energy functional is given by

$$\mathcal{H} = \frac{1}{2}\int_\Omega \left(u^2 + v^2 + q^2\right)\, d\Omega. \tag{3.26}$$

Multiplication of (3.25) with u, v, p and q, respectively, integration over the domain followed by a summation and use of the boundary condition $\mathbf{u} \cdot \mathbf{n} = 0$ with outward normal $\mathbf{n}$, results in energy conservation of the system: $\dot{\mathcal{H}} \equiv d\mathcal{H}/dt = 0$, where the dot represents a time derivative.

A weak formulation of (3.25) is given by

$$\int_\Omega \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}^\perp\right)\phi \; d\Omega = -\int_\Omega (\nabla p)\phi \; d\Omega, \tag{3.27a}$$

$$\int_\Omega (-q)\phi \; d\Omega - \int_\Omega \nabla\phi\cdot\mathbf{u} \; d\Omega \; = -\int_{\partial\Omega} \mathbf{u}\cdot\mathbf{n}\phi \; \mathbf{dS}, \tag{3.27b}$$

$$\int_\Omega \frac{\partial q}{\partial t}\phi \; d\Omega = -\int_\Omega p\phi \; d\Omega, \tag{3.27c}$$

where the velocity field is given in terms of the two-dimensional vector $\mathbf{u} = (u, v)^T$, the perpendicular velocity vector is $\mathbf{u}^\perp = (-v, u)^T$, the two-dimensional differential operator is $\nabla = (\partial/\partial x, \partial/\partial y)^T$and ϕ is a test function taken from $H_0^1(\Omega)$. Hence, the function spaces are chosen to be

$$H_0^1(\Omega) = \{v \; \in \; H^1(\Omega) : v = 0 \text{ on } \partial\Omega\}, \text{ with} \tag{3.28a}$$

$$H^1(\Omega) = \{v \; \in \; L^2(\Omega) : \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y} \; \in L^2(\Omega)\}, \tag{3.28b}$$

where L^2 is a space of square integrable functions.

Note that after multiplication with a test function ϕ, we only do integration by parts in the continuity equation, and skip the integration by parts in the momentum equation, which is a slight modification of a classical FEM weak formulation. This way we ensure the conservation of the energy on the discrete level, as shown later. The latter is crucial for an accurate and robust numerical scheme.

After taking into an account no flow boundary conditions $\mathbf{u}\cdot\mathbf{n} = \mathbf{0}$ at $\partial\Omega$, (3.27) becomes

$$\int_\Omega \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}^\perp\right)\phi \; d\Omega = -\int_\Omega (\nabla p)\phi \; d\Omega, \tag{3.29a}$$

$$\int_\Omega (-q)\phi \; d\Omega - \int_\Omega \nabla\phi\cdot\mathbf{u} \; d\Omega = 0, \tag{3.29b}$$

$$\int_\Omega \frac{\partial q}{\partial t}\phi \; d\Omega = -\int_\Omega p\phi \; d\Omega. \tag{3.29c}$$

Given a tessellation $\mathcal{I}_h$ of the domain Ω, we search for a weak solution of system (3.29) in

$$V_h = \{v : v \text{ is continuous on } \Omega, v|_K \; \in \mathcal{P}^d(K) \; \forall K \; \in \; \mathcal{I}_h\} \subset H^1(\Omega), \tag{3.30}$$

with $\mathcal{P}^d(K)$ the space of polynomials of at most degree d on $K \in \mathcal{I}_h$, where $d \geq 0$. Variables u, v, p, q are represented via their expansions in terms of basis functions

$$\mathbf{u} = \mathbf{u}_i\phi_i, \quad p = p_i\phi_i, \quad q = q_i\phi_i, \tag{3.31}$$

where $\phi_i \ \in V_h$ and $\mathbf{u}_i$, p_i, q_i are expansion coefficients. We note that, here and hereafter, we use the Einstein convention implying summation over repeated indices. Incorporation of (3.31) into the weak formulation (3.29) results in

$$M_{ij}\dot{\mathbf{u}}_j + M_{ij}\mathbf{u}_{\mathbf{j}}^{\perp} = -\mathbf{S}_{ij}p_j, \tag{3.32a}$$

$$M_{ij}q_j + \mathbf{S}_{ji} \cdot \mathbf{u}_j = 0, \tag{3.32b}$$

$$M_{ij}\dot{q}_j = -M_{ij}p_j, \tag{3.32c}$$

where $M_{ij} = \int_{\mathcal{I}_h} \phi_i\phi_j \ d\mathbf{x}$, $\mathbf{S}_{ij} = (S_{ij}^x, S_{ij}^y)^T = \int_{\mathcal{I}_h} \phi_i \nabla \phi_j \ d\mathbf{x}$. If we multiply the discrete momentum equation with $\mathbf{u}_i$, the continuity equation with p_i, (3.32c) with q_i and sum over all nodes, we will obtain the following equation

$$\frac{d}{dt}\Big(\frac{1}{2}\left(M_{ij}\mathbf{u}_i \cdot \mathbf{u}_j + M_{ij}q_iq_j\right)\Big) = 0, \tag{3.33}$$

which ensures the conservation of a discrete energy functional in time, in addition to the conservation at the continuous level.

The unknowns p_j and q_j can be eliminated from (3.32): when a time derivative of (3.32b) is taken and the results are substituted into (3.32a) while using (3.32c), we obtain

$$M_{ij}\ddot{u}_j + S_{il}^x M_{lk}^{-1} S_{jk}^x \ddot{u}_j + S_{il}^x M_{lk}^{-1} S_{jk}^y \ddot{v}_j = M_{ij}\dot{v}_j, \tag{3.34a}$$

$$M_{ij}\ddot{v}_j + S_{il}^y M_{lk}^{-1} S_{jk}^x \ddot{u}_j + S_{il}^y M_{lk}^{-1} S_{jk}^y \ddot{v}_j = -M_{ij}\dot{u}_j. \tag{3.34b}$$

After incorporation of the Ansatz $(u, v) \propto \exp(-i\sigma t)$ into (3.34), we arrive at the following global generalised eigenvalue problem

$$i\sigma \begin{bmatrix} M_{ij} + S_{il}^x M_{lk}^{-1} S_{jk}^x & S_{il}^x M_{lk}^{-1} S_{jk}^y \\ S_{il}^y M_{lk}^{-1} S_{jk}^y & M_{ij} + S_{il}^y M_{lk}^{-1} S_{jk}^x \end{bmatrix} \begin{bmatrix} u_j \\ v_j \end{bmatrix} = \begin{bmatrix} 0 & -M_{ij} \\ M_{ij} & 0 \end{bmatrix} \begin{bmatrix} u_j \\ v_j \end{bmatrix}. \tag{3.35}$$

3.3.2 Numerical eigenfrequencies and tests against semi-analytical solutions

In the following subsection a numerical solution of the generalised eigenvalue problem (3.35) is discussed. For the particular simulation two-dimensional linear Bernstein polynomials were chosen as a set of basis functions for the FEM problem given on the rectangular tessellation $\mathcal{I}_h$ of the continuous domain Ω, see Figure 3.7.

For a given $n \in \mathbb{N}/\{0\}$, the corresponding Bernstein polynomials of degree n are defined by

$$B_k^n(x) = \binom{n}{k} x^k (1-x)^{n-k},\ \forall x \in [0,1], \qquad k = 0, 1, ..., n, \tag{3.36}$$

where $\binom{n}{k}$ is a binomial coefficient and $\mathbb{N}$ is the set of natural numbers. Bernstein polynomials are linearly independent and span the space of polynomials of degree n. Bernstein polynomials are invariant under affine transformations, and all the terms of the Bernstein basis are positive on the interval where they are defined, and their sum equals to one. Additionally, a Bernstein polynomial is always better conditioned than a polynomial of power form for the determination of simple roots on the unit interval $[0,1]$. The latter properties justify our choice of Bernstein polynomials to represent the polynomial space in the definition of V_h in (3.30).

The solution of the generalised eigenvalue problem (3.35) emerges in pairs of eigenvalues and corresponding eigenvectors. The set of eigenvalues is compared to the eigenvalues calculated from the semi-analytical solutions of the same problem, as discussed in the previous section. Unfortunately, alongside the acceptable eigenvalues, the method produces high frequency noise. Nevertheless, it appears that numerical, 'noisy' eigenvalues are not consistent for different runs, with different mesh sizes, which enables us to construct a simple algorithm for identifying the acceptable eigenvalues from the numerical noise. The numerical implementation of (3.35) is initialised consequently with four different meshes, four different tessellations of a domain $\Omega = [2\pi \times \pi]$ into 20×10, 40×20, 80×40 and 160×80 elements. The latter results in four sets of eigenvalues. From these four sets of eigenvalues, we notice that acceptable eigenvalues are converging. By contrast, the numerical noise eigenvalues are appearing and disappearing in the different runs. The latter suggests a simple 'decision algorithm': stationary (one has to allow convergence shifts, of course) eigenvalues for different runs and mesh-size are considered acceptable, and the rest is numerical noise. In other words, in the scale, which corresponds to the densest

mesh size, the presence of four converging eigenvalues (from different sets) is identifying the numerical eigenvalue as acceptable, otherwise it is considered numerical noise. In Figure 3.8 we present a graphical interpretation of the suggested 'decision algorithm'. The spots that are overlaid from all four sets of eigenvalues on the densest mesh scale ($\Delta x \approx 0.006$) are acceptable. We note that eigenvalues which are close and/or converge to 1 can be neglected, because the eigenfrequency we search for need to be smaller than one. Also, substantial numerical noise is noticeable near 0, but this behaviour is expected for any solution of a numerical eigenvalue problem. Thus, following the latter 'decision algorithm', we are able to reproduce every eigenfrequency found by the semi-analytical methods.
After the exact matching of numerical eigenfrequencies with semi-analytical eigenfrequencies, we proceed to compare the corresponding velocity fields, through the comparison of the eigenvectors. In the previous section, the discussion revolved around the comparison of two semi-analytical solutions for the velocity field of the highest eigenfrequency in a $[2\pi \times \pi]$ domain. Hence, here too, we consider the velocity field corresponding to the same $\sigma \approx 0.657$ eigenfrequency. In Figure 3.9 both components of the numerical two-dimensional velocity field corresponding to the eigenvalue $\sigma \approx 0.657$ are depicted. The numerical velocity field is a result of a simulation on a mesh with 80×40 elements. Flow near the normal boundaries is nearly absent. It is apparent that the equations and the corresponding boundary conditions are satisfied up-to FEM accuracy.

Results discussed in the previous section justify a comparison plot of the numerical solution against just Taylor's semi-analytical solution. Hence, in Figure 3.10 we introduce a comparison plot of the two velocity vectors. The comparison is fair, due to the comparability in sizes of the resulting eigenvalue problems in both the numerical and semi-analytical cases. As was expected, the difference is noticeable near the normal boundaries, where Taylor's method has slow convergence.

3.4 Summary and conclusions

We have shown numerically that the Proudman-Rao method for deriving modal solutions of the linear, rotating incompressible Euler equations in a (planar) rectangular parallelepiped bounded with solid walls suffers from poor convergence in the interior of the fluid domain as well as a Gibbs phenomenon at the boundaries. Despite the concise structural construction, the solution is practically unusable. Therefore, an alternative mode decomposition solution (Taylor's method) was presented. The three-dimensional

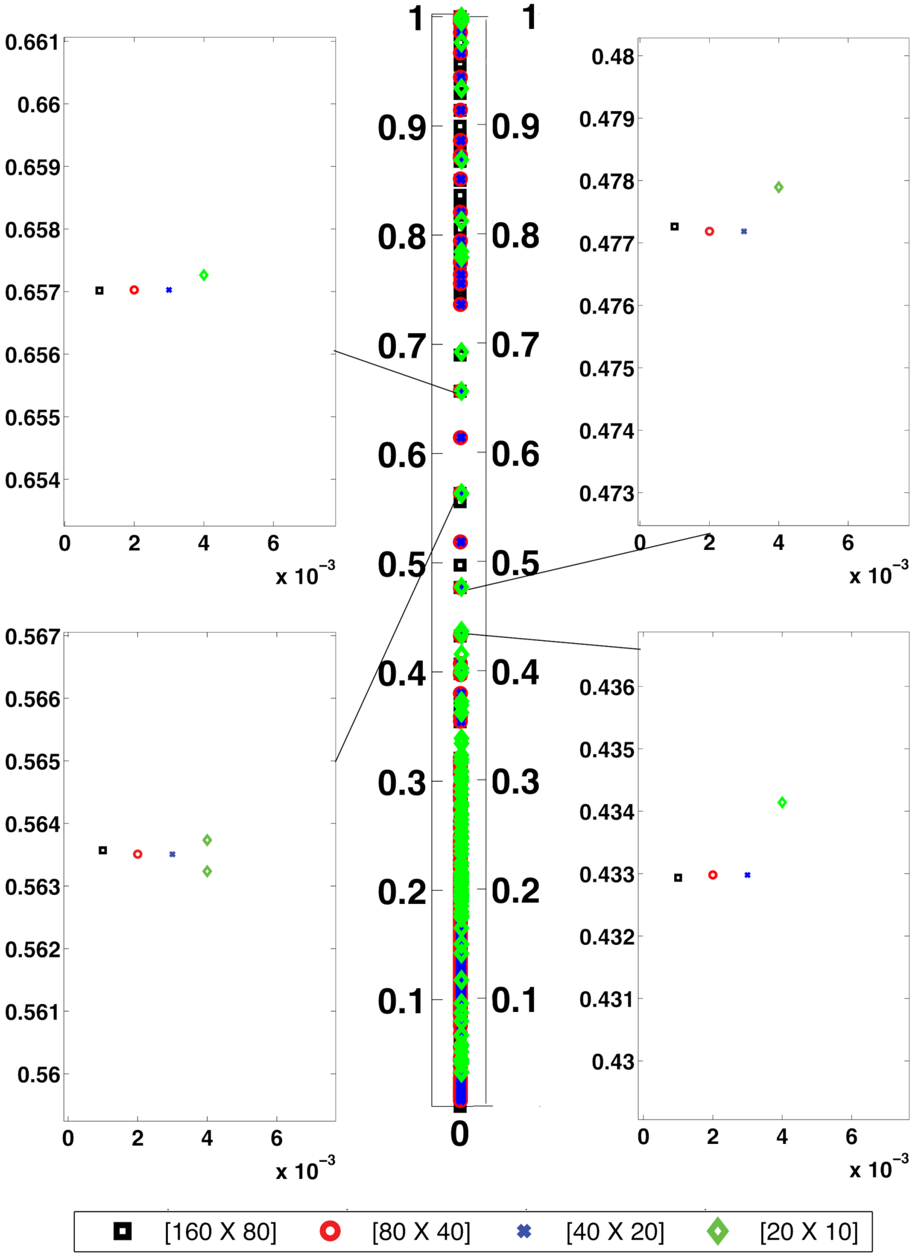
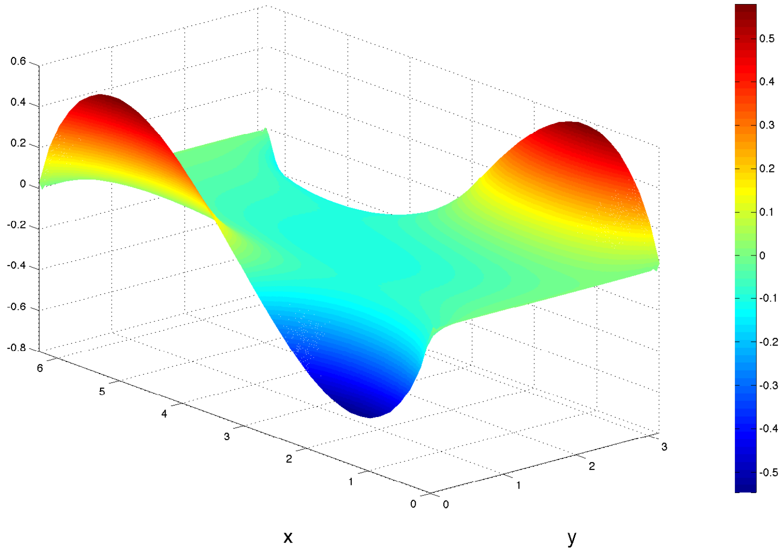
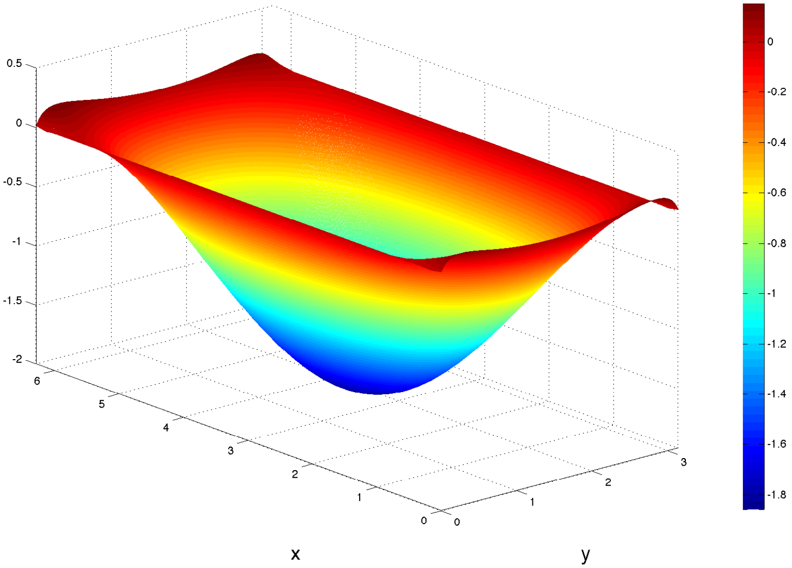


Figure 3.8: Four different sets of eigenvalues are plotted here. Black squares, red circles, blue crosses and green diamonds correspond to the set of eigenvalues from simulations with mesh-sizes 160 × 80, 80 × 40, 40 × 20 and 20 × 10, respectively. The first four highest eigenfrequencies are highlighted.



(a) u profile



(b) v profile

Figure 3.9: u and v components of the numerical horizontal velocity for mode $\sigma \approx 0.657$ are presented in (a) and (b) subplots, respectively. An 80×40-element mesh is used for the simulations in a domain which is rotating anti-clockwise.

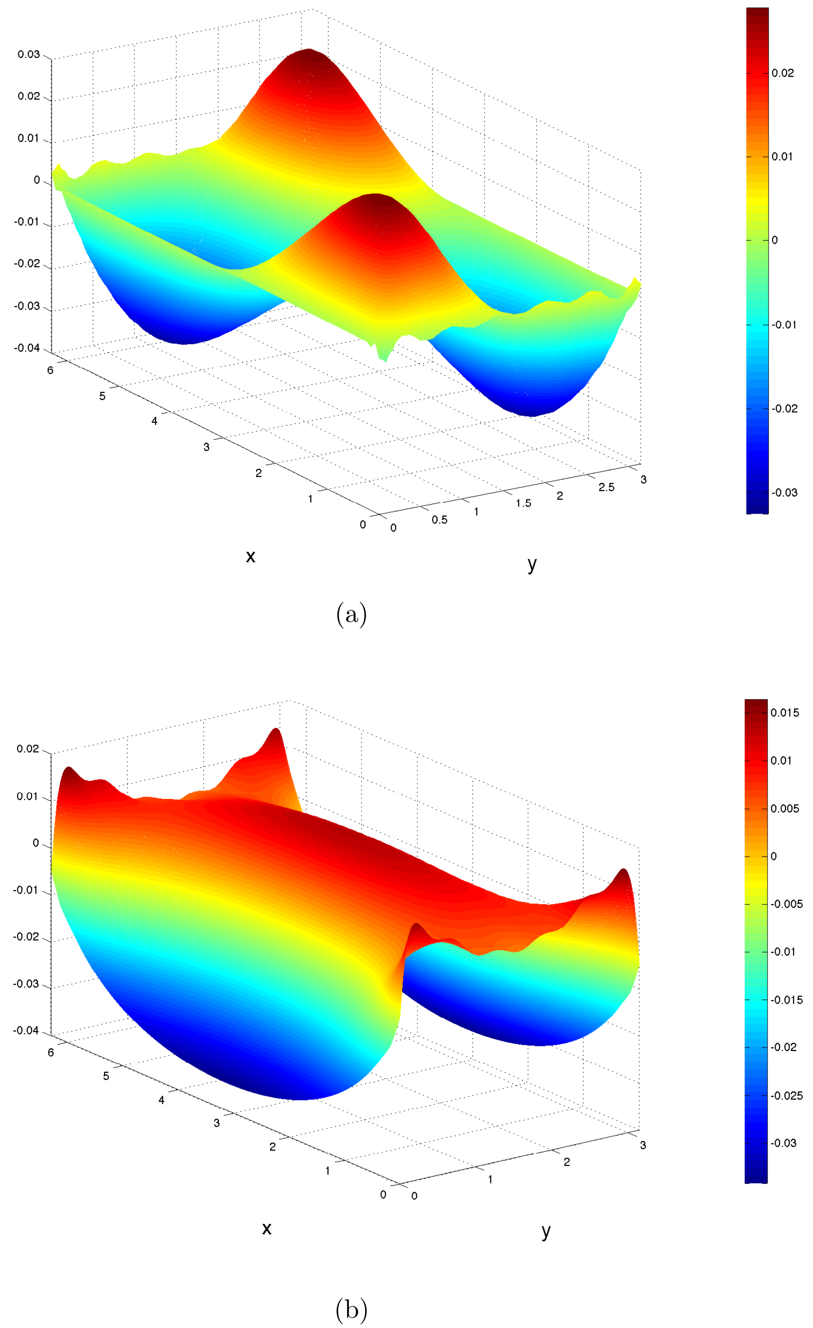


Figure 3.10: (a) and (b) subfigures give the difference between the numerical FEM solution and Taylor's semi-analytical solution in the u and v velocity components, respectively. An 80×40-element mesh is used for the FEM simulation.

problem was reduced to a two-dimensional problem by using the Ansatz of vertical modes in the z-direction, exactly repeating the arguments from [62]. By scaling the depth of the tank with $H_*/n\pi$ for the vertical mode n, we remove all reference to the vertical and the problem can be solved for each vertical mode strictly in the horizontal plane (whose size is fixed except for an n-dependent rescaling of the basin's size). The resulting two-dimensional problem in the horizontal plane was solved by employing ideas and results that Taylor [101] used to determine the rotational effects on long surface gravity waves. As in the Proudman-Rao method, Taylor's method also leads to an infinite matrix eigenvalue problem, whose solution upon truncation gives similar results. The novel mode solutions satisfy the linear Euler equations exactly, thus they are considered to be an improvement over those obtained with the Proudman-Rao method. Nevertheless, the novel semi-analytical solution has its own flaws. The mode solutions are, by construction, a superposition of inertial analogs of surface Kelvin and Poincaré wave solutions, which converge to solutions that satisfy the solid-wall boundary conditions. Unfortunately, the latter convergence is also slow. By contrast, the Proudman-Rao solution satisfies the no-normal flow boundary conditions exactly.

The latter motivated us to apply a continuous finite element (FEM) discretisation to the reduced two-dimensional problem (based on using a standing wave in the vertical of the original three-dimensional problem) in order to obtain numerical mode solutions that satisfy no-normal flow boundary conditions by construction. A modified FEM discretisation is proven to be symmetric and energy conserving on a discrete level, which plays an essential role in the stability and accuracy of this scheme (e.g., [18]). The resulting discrete system is solved via a generalised eigenvalue solver, which unfortunately produces a substantial amount of numerical noise. Nevertheless, a simple 'decision algorithm' is suggested to separate acceptable numerical eigenfrequencies from numerical noise. Finally, this numerical solution is tested against the semi-analytical solution.

Extensive comparison between the two semi-analytical and numerical solutions enables one to adopt the most appropriate method for resolving the inertial waves. The Proudman-Rao method facilitates fast convergence of eigenfrequencies and the determination of semi-analytical solutions that satisfy the boundary conditions exactly because this is embedded in the construction algorithm. However, this method displays a Gibbs phenomenon at the boundaries. Unlike the Proudman-Rao method, Taylor's method enables a semi-analytical solution exactly satisfying the governing equations, but with slow convergence near the boundaries. The numerical solution, based on a modified FEM discretisation, implements a very accurate but relatively slow method, which requires an extra step to separate the acceptable eigenfrequencies from numerical noise. Depending on one's needs one might choose one of the suggested methods.

The solutions we have presented have been used to verify the novel numerical technique developed in [79] for the initial-value problem of three-dimensional inertial waves in arbitrarily shaped domains. This method is geared to investigate whether wave attractors ([61, 69, 88]) and complex eigenmodes (such as in [17]) emerge in domain shapes of sufficient geometric complexity.

Additionally, we note that solutions of the Euler equations, by default inviscid, presented above are also relevant in the viscous case (Navier-Stokes equations), which is not intuitive at first glance. In experiments like those in [34, 73, 65, 67, 53, 102, 4, 64, 106, 12, 14, 54, 15, 69], a variety of forcing mechanisms and geometries have been applied. In some cases forcing occurs by pumping through viscous boundary layers, e.g. either on convex (outer) or concave (inner) parts of spherical shells; in other studies by means of the Euler force (using libration). Experiments performed in square domains [14, 54, 15], seem to show that inviscid eigenmodes derived above are relevant in the sense that energy spectra display peaks at inviscid eigenfrequencies of some of the larger scale eigenmodes. In two experiments [14, 54] forcing arises by pulling a grid through the fluid, leaving a set of waves behind. In another experiment with a libration ([15]), forcing occurs through Ekman-layer convergence, lending the spatial structure of the response much more of a beam-like character. Note, however, that due to the degeneracy discussed in [62] (near-degeneracy for numerical results) internal (gravity) and inertial waves have a much more flexible ('chameleonic') spatial structure than that of the eigenmodes of elliptic problems (like surface waves). Indeed, in general one observes that the inviscid response (in combination with any spatial structure in the forcing) is determining the field's spatial structure, slaving the viscous (boundary layer) response, and not the other way around.

Chapter 4

Nonlinear dynamics

A new Hamiltonian formulation for three-dimensional inviscid compressible fluid flow in a rotating domain containing both solid walls and a free surface is presented. Initially a Hamiltonian energy functional and related Poisson bracket are conjectured, which formulation results in the compressible Euler equations with a free surface. Subsequently, the same Hamiltonian formulation is derived starting from a Lagrangian functional expressed in terms of Eulerian variables. This derivation starts with a transformation of the Lagrangian functional in Eulerian variables to canonical Clebsch variables. Subsequently, the variational principle is applied to the transformed Lagrangian functional, which yields the Hamiltonian formulation in terms of the canonical Clebsch variables. A further transformation from Clebsch variables to Eulerian variables results in the same Hamiltonian functional and Poisson bracket that were originally conjectured and concludes the verification of the Hamiltonian formulation.

4.1 Introduction

Many wave phenomena in the ocean are governed by the (non)linear (in)compressible Euler equations, coupled with free surface boundary conditions. The Euler equations are obtained from the Navier-Stokes equations by setting the viscosity and heat conduction terms to zero. In the ocean, fluid flow can be considered incompressible due to the small changes in the density profile over space and time, and inviscid because of the relatively small influence of viscosity and surface tension on waves and currents. Due to the interaction of waves and currents vorticity is, however, important for many applications. This motivates the use of the Euler equations, since they allow the modelling and computation of free surface waves, currents and wave-current interactions in the deep ocean, including the effects of vorticity, which is frequently neglected in other mathematical models.

Over the past decades a wide variety of nonlinear wave models for deep water free surface and internal waves have been developed. Most of them were based on potential flow theory (e.g. [59, 29]), where the wave propagation is essentially considered irrotational and inviscid (until the point of wave breaking). The potential flow equations consist of the Laplace equation for the velocity potential in the interior of the fluid coupled to two nonlinear equations at the free surface: the kinematic boundary condition, describing the evolution of the free surface in time, and the dynamic boundary condition, which is essentially Bernoulli's equation. In this way, the dynamics of free surface waves in a non-stratified fluid can be described until wave breaking. The Euler equations provide an important extension of the potential flow model by including vorticity.

Instead of the Euler equations it is also possible to consider the full Navier-Stokes equations (e.g. [94] and references therein), but for long time simulations on a large domain, which is often necessary in free surface wave simulations, these methods are currently still too dissipative and not sufficiently accurate.

An important feature of both the potential flow and Euler models is that they have a variational and Hamiltonian structure. This can be employed to develop numerical discretisations based on a discrete variational and Hamiltonian structure with associated discrete conservation laws, with benefits such as no numerical dissipation (no energy drift), minimal dispersion error and excellent long time stability. An example of such numerical discretisation for the linear Euler equations for incompressible flow is discussed in Chapter 2. Additional benefit of having numerical discretisations based on a variational principle is that the complete dynamics is defined through one functional, a coordinate system-independent Lagrangian, that is often a good starting point for a variational finite element discretisation. In case of a proper choice for the numerical discretisation important conservation laws and the mathematical structure of the continuous mathematical model can than also be preserved in a discrete sense.

A further formalisation of the variational principle may, in some cases, result in a Hamiltonian system. The benefits of the Hamiltonian framework, presented in the introductory chapter of this thesis, motivated us to derive a Hamiltonian description of the three dimensional Euler equations for an inviscid, compressible fluid with a free surface boundary condition in a rotating frame. This is a first, important step in the development of an equivalent model for the incompressible Euler equations with a free surface using the Dirac theory, which is applied in Chapter 2 to obtain a variational discretisation of the linearised Euler equations for incompressible flow.

In this chapter the main focus will be on the Hamiltonian formulation for nonlinear compressible flow with a free surface. In Section 4.2 the Hamiltonian formulation for a nonlinear, inviscid compressible fluid in a domain with a free surface boundary and solid walls is conjectured. The correctness of this mathematical model is subsequently verified in Section 4.3 using an action principle applied to a Lagrangian expressed in Clebsch variables. This makes it much easier to verify satisfaction of the mandatory properties of the Poisson bracket (1.2) and, after a change of variables, we can derive the same Hamiltonian formulation as was conjectured in Section 4.2. In Section 4.4 we also discuss the special case of the linearised compressible Euler equations with a free surface. Conclusions are drawn in Section 4.5.

4.2 Nonlinear Hamiltonian framework

We consider a rectangular domain D with solid lateral boundaries on the sides, a rigid bed at the bottom and a free surface at the top (see Figure 4.1). The boundaries of the domain D are denoted with $\partial D = \cup_{i=1}^{5} \partial D_i \cup \Gamma_s$, where ∂D_1, ∂D_2, ∂D_4 and ∂D_5 are the vertical lateral boundaries, ∂D_3 is the bottom and Γ_s is the free surface. Cartesian coordinates and time are introduced via $\mathbf{x} = (x, y, z)$ and t, respectively; the three-dimensional and two-dimensional differential operators are denoted as $\boldsymbol{\nabla} = (\partial/\partial x, \partial/\partial y, \partial/\partial z)^T$ and $\nabla = (\partial/\partial x, \partial/\partial y)^T$. The three-dimensional velocity, scalar density and pressure fields are represented, respectively, as $\mathbf{u} = \mathbf{u}(x, y, z, t) = (u, v, w)^T$, $\rho = \rho(x, y, z, t)$, and $P = P(\rho)$. The free surface is described as $z = h(x, y, t) + b(x, y)$ relative to a horizontal reference plane $z = 0$, where $h = h(x, y, t)$ is the water depth and $b = b(x, y)$ a fixed bottom topography. The constant gravitational acceleration g is acting in

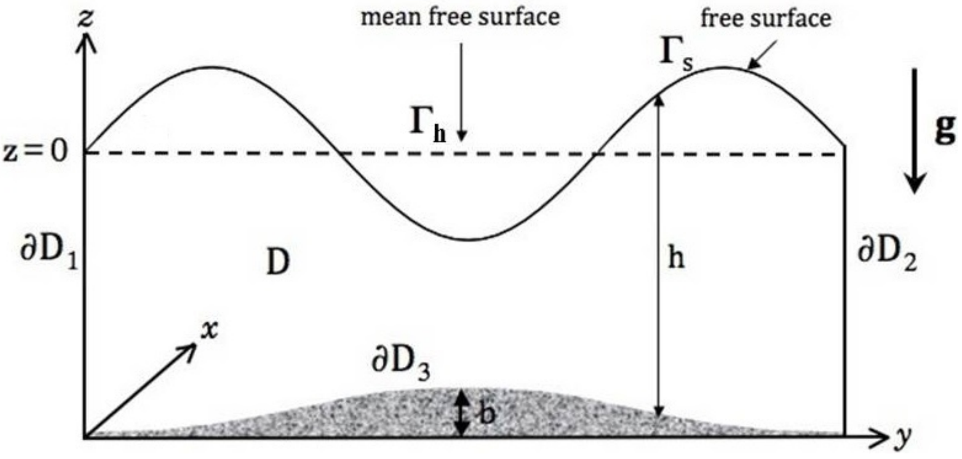


Figure 4.1: A vertical cut of a three-dimensional domain and its boundaries.

the negative z-direction. The background rotation of the domain is added via the angular velocity $\boldsymbol{\Omega} = (\Omega_1, \Omega_2, \Omega_3)^T$. The compressible fluid flow in the domain D is governed by the nonlinear, isentropic (or barotropic) compressible Euler equations, together with the kinematic and dynamic boundary conditions at the free surface

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u} - 2\boldsymbol{\Omega} \times \mathbf{u} - \rho^{-1}\boldsymbol{\nabla} P(\rho) - \boldsymbol{\nabla}(gz), \tag{4.1a}$$

$$\frac{\partial \rho}{\partial t} = -\boldsymbol{\nabla} \cdot (\rho \mathbf{u}), \tag{4.1b}$$

$$\frac{\partial h}{\partial t} = \mathbf{u} \cdot (-\nabla(h+b), 1)^T \quad \text{at the free surface } \Gamma_s, \tag{4.1c}$$

$$P = P_c \quad \text{at the free surface } \Gamma_s, \tag{4.1d}$$

where P_c is the constant atmospheric pressure at the free surface. The internal energy of the fluid $U(\rho)$ and the pressure are connected as $P = \rho^2 U_\rho$.

The Hamiltonian energy functional $\mathcal{H}$ is the total energy of the dynamical system. It consists of the integral of the densities of the kinetic energy $\frac{1}{2}\rho|\mathbf{u}|^2$, the internal energy $\rho U(\rho)$ and the potential energy $\rho g z$, integrated over the time dependent domain $D = D(t)$,

$$\mathcal{H} = \int_{D(t)} \rho \left(\frac{1}{2} |\mathbf{u}|^2 + U + gz \right) d\Omega. \tag{4.2}$$

Note, the domain is changing in time due to the time dependent free surface motion at the top. We limit ourselves to the case in which the free surface stays single-valued over time. Hence, the energy functional $\mathcal{H}$ can be rewritten as a double integral over the horizontal projection of the free surface and an integral from the bottom to the time-dependent free surface

$$\mathcal{H} = \int_{\Gamma_h} \int_b^{h(x,y,t)+b(x,y)} \left(\frac{1}{2} \rho |\mathbf{u}|^2 + \rho U(\rho) + \rho g z \right) dz\ dxdy, \tag{4.3}$$

where Γ_h is the projection of the free surface Γ_s onto the horizontal plane. In this case, variations of the Hamiltonian energy functional are as follows

$$\begin{aligned} \delta\mathcal{H} &\equiv \lim_{\epsilon \to 0} \frac{\mathcal{H}[\mathbf{u} + \epsilon\delta\mathbf{u}, \rho + \epsilon\delta\rho, h + \epsilon\delta h] - \mathcal{H}[\mathbf{u}, \rho, h]}{\epsilon} \\ &= \int_{\Gamma_h} \int_{b(x,y)}^{h(x,y,t)+b(x,y)} \rho\mathbf{u} \cdot \delta\mathbf{u} + \left(\frac{1}{2} |\mathbf{u}|^2 + (\rho U)_\rho + gz \right) \delta\rho dz\ dxdy + \\ &\quad \int_{\Gamma_h} \left(\frac{1}{2} \rho |\mathbf{u}|^2 + \rho U + \rho g z \right)_{z=h(x,y,t)+b(x,y)} \delta h\ dxdy. \end{aligned} \tag{4.4}$$

As a consequence of the variations $\delta\rho$, $\delta\mathbf{u}$ and δh being arbitrary, the functional derivatives of the Hamiltonian take the following form

$$\begin{aligned} \frac{\delta\mathcal{H}}{\delta\mathbf{u}} &= \rho\mathbf{u}, \quad \frac{\delta\mathcal{H}}{\delta\rho} = (\rho U)_\rho + \frac{1}{2} |\mathbf{u}|^2 + gz, \\ \frac{\delta\mathcal{H}}{\delta h} &= \left[\frac{1}{2} \rho |\mathbf{u}|^2 + \rho U + \rho g z \right]_{z=h(x,y,t)+b(x,y)}. \end{aligned} \tag{4.5}$$

To complete the introduction of the Hamiltonian framework for a nonlinear compressible fluid flow with a free surface we have to introduce a (generalised) Poisson bracket.

Traditionally, there are two approaches to obtain a Poisson bracket. Firstly, one can try to conjecture a complete set of (non)canonical variables and state a skew-symmetric bracket, which later should be proven to satisfy all the requirements of a Lie algebra. The bracket then is the required

Poisson bracket. Secondly, one could also start with a corresponding variational principle, apply an action principle to the Lagrangian functional in canonical variables, derive the Poisson bracket and transform it from canonical variables to Eulerian variables.

Here, we start with conjecturing a skew-symmetric Poisson bracket and show its correspondence to the nonlinear Euler equations with a free surface, leaving the indirect proof of the bracket properties (1.2), Jacobi identity in particular, till Section 4.3.

As starting point we use the Poisson bracket for a three-dimensional ideal fluid in non-canonical Eulerian variables presented in [76], with the addition of a background rotation of the domain,

$$\{\mathcal{F},\mathcal{H}\} = \int_D \frac{\delta\mathcal{H}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}} - \frac{\delta\mathcal{F}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}} + \frac{2\boldsymbol{\Omega}+\boldsymbol{\nabla}\times\mathbf{u}}{\rho}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\times\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\;d\Omega. \tag{4.6}$$

This addition allows the modelling of interior fluid flow in rotating domains, governed by the three-dimensional nonlinear Euler equations, but lacks any representation of the free-surface dynamics. The extension of the Poisson bracket to account for the free surface dynamics must be done with care, since it can easily interfere with other contributions in the Poisson bracket. We conjecture that the following bracket combines both the dynamics of the flow field described by the compressible Euler equations and the free surface

$$\begin{aligned}\{\mathcal{F},\mathcal{H}\} = &\int_D \frac{\delta\mathcal{H}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}} - \frac{\delta\mathcal{F}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}} + \frac{2\boldsymbol{\Omega}+\boldsymbol{\nabla}\times\mathbf{u}}{\rho}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\times\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\;d\Omega\\ &+\int_{\Gamma_s}\frac{1}{\rho}\hat{\mathbf{n}}_s\cdot\left[-\frac{\delta\mathcal{H}}{\delta h}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}+\frac{\delta\mathcal{F}}{\delta h}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right]_{z=h(x,y,t)+b(x,y)}\;dS,\end{aligned} \tag{4.7}$$

where $\hat{\boldsymbol{n}}_s = (-\nabla(h+b),1)/|(-\nabla(h+b),1)|$ is the unit outward normal vector at the free surface with $\hat{\boldsymbol{n}}_s dS = \boldsymbol{n}_s dxdy$ and $\boldsymbol{n}_s = (-\nabla(h+b),1)^T$.

Generally, in PDE-based models, boundary conditions are specified on the variables (e.g. $\mathbf{u}\cdot\hat{\boldsymbol{n}}_s = 0$ at solid walls), whereas, in Hamiltonian dynamics essential boundary conditions are frequently specified via a restriction on the function spaces on which the unknown functional is defined. In case of a solid wall boundary condition, the arbitrary functional $\mathcal{F}[\mathbf{u}]$ is defined as

$$\mathcal{F}[\mathbf{u}] = \int_D \boldsymbol{\Phi}_u(\mathbf{x})\cdot\mathbf{u}(\mathbf{x},\mathbf{t})\;d\Omega, \tag{4.8}$$

where $\boldsymbol{\Phi}_u \in \mathcal{Y}$ and

$$\mathcal{Y} = \{\boldsymbol{\Phi} \in (L^2(D))^3 \ : \ \boldsymbol{n}_i \cdot \boldsymbol{\Phi} = 0 \text{ at } \cup_{i=1}^5 \partial D_i\}, \tag{4.9}$$

with $\boldsymbol{n}_i$ the normal vector at the boundary ∂D_i and L^2 the space of square integrable functions. The condition in (4.9) implies that variations of the normal component of the velocity field at the solid wall boundaries $\cup_{i=1}^5 \partial D_i$ must be zero. Further, we employ an additional restriction on the arbitrary functional F to incorporate the free surface boundary condition

$$\frac{\partial \mathcal{F}}{\partial \rho} = \frac{1}{\rho}\frac{\partial \mathcal{F}}{\partial h} \quad \text{at } \Gamma_h. \tag{4.10}$$

Conditions (4.9) and (4.10) concern any functional present in the definition of the Hamiltonian dynamics, including the energy functional $\mathcal{H}$. The solid wall restriction (4.9) in the context of the Hamiltonian energy functional (4.3) is equivalent to $\mathbf{u} \cdot \boldsymbol{n}_i = 0$ at the boundary ∂D_i. Similarly, the free surface restriction (4.10) for the Hamiltonian energy functional is equivalent to (4.1d) at the free surface Γ_s. As it follows from the latter, both restrictions presented at the functional level are also natural at the PDE level.

The definition of the Poisson bracket (4.7), in combination with the Hamiltonian functional $\mathcal{H}$ (4.3) and the essential no-flow boundary conditions (4.9), complete our Hamiltonian description of a nonlinear, inviscid compressible flow with a free surface in a rotating reference frame.

Next, we verify that the conjectured Hamiltonian system, given by (4.3), (4.7) and (4.9)-(4.10), leads to the compressible Euler equations (4.1a)-(4.1b), with free surface boundary conditions (4.1c)-(4.1d).

Firstly, the momentum equations (4.1a) are obtained by taking the functional $\mathcal{F} = \mathcal{F}[\mathbf{u}]$, defined in (4.8), together with the functional derivatives (4.5) of the Hamiltonian $\mathcal{H}$,

$$\begin{aligned}\frac{d\mathcal{F}[\mathbf{u}]}{dt} = \int_D \Big[&\big(\frac{1}{2}|\mathbf{u}|^2 + (\rho U)_\rho + gz\big) \boldsymbol{\nabla} \cdot \frac{\delta \mathcal{F}[\mathbf{u}]}{\delta \mathbf{u}} \\ &-(2\boldsymbol{\Omega} + \boldsymbol{\nabla} \times \mathbf{u}) \times \mathbf{u} \cdot \frac{\delta \mathcal{F}[\mathbf{u}]}{\delta \mathbf{u}} \Big] \, d\Omega \\ &+ \int_{\Gamma s} -\frac{1}{\rho}\frac{\delta \mathcal{H}}{\delta h}\frac{\delta \mathcal{F}[\mathbf{u}]}{\delta \mathbf{u}} \cdot \hat{\mathbf{n}}_s \, dS. \end{aligned} \tag{4.11}$$

Integration by parts with subsequent restriction of the variations of the functional $\mathcal{F}[\mathbf{u}]$ at $\cup_{i=1}^{5}\partial D_i$ using (4.9) yields

$$\begin{aligned}\frac{d\mathcal{F}[\mathbf{u}]}{dt} = &- \int_D \left(\boldsymbol{\nabla}\left[\frac{1}{2}|\mathbf{u}|^2 + (\rho U)_\rho + gz\right] + (2\boldsymbol{\Omega} + \boldsymbol{\nabla}\times\mathbf{u})\times\mathbf{u}\right)\cdot\frac{\delta\mathcal{F}[\mathbf{u}]}{\delta\mathbf{u}}\, d\Omega \\ &+ \int_{\Gamma s} \frac{\delta\mathcal{H}}{\delta\rho}\frac{\delta\mathcal{F}[\mathbf{u}]}{\delta\mathbf{u}}\cdot\hat{\mathbf{n}}_s\, dS - \int_{\Gamma s}\frac{1}{\rho}\frac{\delta\mathcal{H}}{\delta h}\frac{\delta\mathcal{F}[\mathbf{u}]}{\delta\mathbf{u}}\cdot\hat{\mathbf{n}}_s\, dS. \qquad (4.12)\end{aligned}$$

This expression can be further evaluated using the identities $(\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{u} = \boldsymbol{\nabla}(1/2|\mathbf{u}|^2) - \mathbf{u}\times(\boldsymbol{\nabla}\times\mathbf{u})$, $P/\rho = \rho U_\rho$ and the boundary condition (4.10) at the free surface. The momentum equation (4.1a) is then obtained using the fact that the test function $\boldsymbol{\Phi}_u$ in the functional derivative of $\mathcal{F}[\mathbf{u}]$ is arbitrary.

Similarly, the continuity equation (4.1b) is obtained if the functional $\mathcal{F}$ is taken as

$$\mathcal{F}[\rho] = \int_D \Phi(\mathbf{x},t)\rho(\mathbf{x},t)\, d\Omega, \qquad (4.13)$$

where $\Phi \in L^2(D)$.

Finally, the free surface equation (4.1c) is obtained if the functional $\mathcal{F}$ is taken as

$$\mathcal{F}[h] = \int_{\Gamma_s} \Phi_s(x,y,t)h(x,y,t)dS, \qquad (4.14)$$

with $\Phi_s \in L^2(\Gamma_s)$. This shows that the inviscid compressible Euler equations in a rotating domain with a free surface and solid boundaries (4.1) are obtained from the Hamiltonian system (4.3),(4.7).

4.3 Derivation of Hamiltonian structure via Clebsch variables

In this section the Hamiltonian system (4.3),(4.7) for nonlinear free surface flow is derived from a variational principle. We start with a Lagrangian functional for the three-dimensional nonlinear Euler equations. After some lengthy calculations the Lagrangian functional is transformed from non-canonical Eulerian variables to a canonical Clebsch variable representation. Next, the Poisson bracket in canonical Clebsch variables is stated using an action principle applied to the Lagrangian functional (e.g. see [92]). Finally, a reduction from canonical Clebsch variables to non-canonical Eulerian variables (velocity $\mathbf{u}$, density ρ and free surface height

h) results in the Poisson bracket which was conjectured in (4.7). In order to reduce the complexity we do not include the background rotation in the derivations below, but the analysis can be extended to include also this term. We start with a Lagrangian variational principle for the three-dimensional nonlinear Euler equations with a free surface. As usual, the Lagrangian functional L is taken to be the following algebraic expression

$$L(\mathbf{u},\rho) = \int_{D(t)} \frac{1}{2}\rho|\mathbf{u}|^2 - \rho U(\rho) - \rho g z \; dxdydz \tag{4.15}$$

consisting of the kinetic, internal and potential energy. Here, the density ρ is the Jacobian between the Eulerian coordinates $\mathbf{x} = (x, y, z)$ and the Lagrangian label coordinates $\boldsymbol{l} = (l_1, l_2, l_3)^T$ for a fluid parcel, such that $\rho\, dx\, dy\, dz = dl_1\, dl_2\, dl_3$. Using Lin constraints [58] as in [24] the Lagrangian functional (4.15) is now extended to include a number of constraints, viz. the mass conservation equation, the label-particle displacement equation and the kinematic free surface boundary condition, which states that a fluid particle at the free surface stays at the free surface. The Lagrangian functional then is equal to

$$\begin{aligned} L(\mathbf{u},\boldsymbol{l},\boldsymbol{\pi},\rho,\phi,h) = \int_{D(t)} & \frac{1}{2}\rho|\mathbf{u}|^2 - \rho U(\rho) - g\rho z + \phi(\partial_t\rho + \boldsymbol{\nabla}\cdot(\rho\mathbf{u})) \\ & - \rho\boldsymbol{\pi}\cdot(\partial_t\boldsymbol{l} + (\mathbf{u}\cdot\boldsymbol{\nabla})\boldsymbol{l}) \; dxdydz \\ & + \int_{\Gamma_h} \lambda(\partial_t h - \mathbf{u}_s\cdot\mathbf{n}_s) \; dxdy, \end{aligned} \tag{4.16}$$

with $\boldsymbol{l} = \boldsymbol{l}(x, y, z, t)$ the Lagrangian fluid labels in the Eulerian framework. The variables $\phi = \phi(\mathbf{x}, t)$, $\boldsymbol{\pi} = \boldsymbol{\pi}(\mathbf{x}, t)$ and $\lambda = \lambda(x, y, t)$ are the Lagrange multipliers to enforce the constraints given by the mass conservation equation, the label-particle displacement equation, and the kinematic free surface boundary condition. We also introduce the notation $\mathbf{u}_s(x, y, t) = \mathbf{u}(x, y, h(x, y, t) + b(x, y))$, with a similar relation for ρ_s, to express variables at the free surface.

The action principle now states that the variations of the Lagrangian functional have to be zero

$$0 = \delta \int_0^T L(\mathbf{u},\boldsymbol{l},\boldsymbol{\pi},\rho,\lambda,\phi,h) \; dt. \tag{4.17}$$

After calculating the variations of the Lagrangian functional with respect to $\mathbf{u}$, using the arbitrariness of $\delta\mathbf{u}$ and $(\delta\mathbf{u})_s$, the variations of $\mathbf{u}$ at the free surface, and the fact that $\delta\mathbf{u}\cdot\mathbf{n}_i = 0$ at $\cup_{i=1}^5 \partial D_i$, we obtain the following expressions for the velocity $\mathbf{u}$ and the Lagrange multiplier λ

$$\mathbf{u} = \boldsymbol{\nabla}\phi + (\boldsymbol{\nabla}\boldsymbol{l})^T\boldsymbol{\pi} \equiv \boldsymbol{\nabla}\phi + \mathbf{v} \qquad \text{in } D\times(0,T), \tag{4.18a}$$

$$\lambda = \phi\rho \qquad \text{at } \Gamma_h\times(0,T). \tag{4.18b}$$

Here we introduced the rotational velocity $\mathbf{v}$, but note that $\mathbf{v}$ is not an independent variable in the calculation of the variations. Next, we eliminate the velocity $\mathbf{u}$ and Lagrange multiplier λ from (4.16) by using (4.18) in (4.17). The Lagrangian functional then is equal to

$$\begin{aligned} L(\boldsymbol{l},\boldsymbol{\pi},\rho,\phi,h) = \\ &\int_D \bigg[\frac{1}{2}\rho|\boldsymbol{\nabla}\phi+\mathbf{v}|^2 - \rho U(\rho) - g\rho z + \phi[\partial_t\rho + \boldsymbol{\nabla}\cdot(\rho(\boldsymbol{\nabla}\phi+\mathbf{v}))] \\ &\qquad - \rho\boldsymbol{\pi}\cdot[\partial_t\boldsymbol{l} + (\boldsymbol{\nabla}\phi+\mathbf{v})\cdot\boldsymbol{\nabla}\boldsymbol{l}]\bigg]\, dxdydz \\ &+ \int_{\Gamma_h} \rho\phi[\partial_t h - (\boldsymbol{\nabla}\phi+\mathbf{v})\cdot\mathbf{n}_s]\, dxdy. \end{aligned} \tag{4.19}$$

Application of the theorems of Gauss and Leibnitz and using the fact that $\mathbf{u}\cdot\mathbf{n}_i = 0$ at $\cup_{i=1}^5 \partial D_i$, we can transform the right hand side of (4.19) into

$$\begin{aligned} &\int_{\Gamma_h}\int_b^{h+b} \frac{1}{2}\rho|\boldsymbol{\nabla}\phi+\mathbf{v}|^2 - \rho U(\rho) - g\rho z - \rho\boldsymbol{\nabla}\phi\cdot(\boldsymbol{\nabla}\phi+\mathbf{v})\, dxdydz + \\ &\frac{d}{dt}\int_{\Gamma_h}\int_b^{h+b} (\phi\rho)\, dxdydz - \int_{\Gamma_h}\int_b^{h+b} \rho\partial_t\phi\, dxdydz - \\ &\int_{\Gamma_h} \phi_s\rho_s\partial_t h - \phi_s\rho_s(\boldsymbol{\nabla}\phi+\mathbf{v})_s\cdot\mathbf{n}_s\, dxdy - \\ &\int_{\Gamma_h}\int_b^{h+b} \rho\boldsymbol{\pi}\cdot(\partial_t\boldsymbol{l} + (\boldsymbol{\nabla}\phi+\mathbf{v})\cdot\boldsymbol{\nabla}\boldsymbol{l})\, dxdydz + \\ &\int_{\Gamma_h} \phi_s\rho_s(\partial_t h - (\boldsymbol{\nabla}\phi+\mathbf{v})_s\cdot\mathbf{n}_s)\, dxdy. \end{aligned} \tag{4.20}$$

Cancellation of terms with opposite sign in (4.20) results in the following action functional

$$\begin{aligned}\int_0^T L(\boldsymbol{l},\boldsymbol{\pi},\rho,\phi,h)\ dt\ =&\int_0^T\int_{\Gamma_h}\int_b^{h+b} -\frac{1}{2}\rho|\boldsymbol{\nabla}\phi+\mathbf{v}|^2-\rho U(\rho)-g\rho z\\ &\qquad\qquad -\rho\partial_t\phi-\rho\boldsymbol{\pi}\cdot(\partial_t\boldsymbol{l})\ dxdydz\ dt\\ &+\int_{\Gamma_h}\int_b^{h+b}\phi\rho|_0^T\ dxdydz. \end{aligned} \tag{4.21}$$

The second integral in (4.21) is zero by imposing that the variation of $\rho\phi$ is zero at the times $t=0$ and $t=T$.

The action functional can now be expressed as

$$\begin{aligned}\int_0^T L(\boldsymbol{l},\boldsymbol{\pi},\rho,\phi,h)\ dt=&\int_0^T\int_{\Gamma_h}\int_b^{h+b}\frac{1}{2}\rho|\boldsymbol{\nabla}\phi+\mathbf{v}|^2+\rho U(\rho)+g\rho z\\ &\qquad\qquad +\rho\partial_t\phi+\rho\boldsymbol{\pi}\cdot\partial_t\boldsymbol{l}\ dxdydz\ dt.\end{aligned} \tag{4.22}$$

From the latter, it follows that Hamiltonian energy functional has the following representation in Clebsch variables

$$\mathcal{H}=\int_{\Gamma_h}\int_b^{h+b}\frac{1}{2}\rho|\boldsymbol{\nabla}\phi+\mathbf{v}|^2+\rho U(\rho)+g\rho z\ dxdydz. \tag{4.23}$$

The Euler equations and the free surface dynamics in Clebsch variables are obtained by applying the action principle to (4.22). A detailed derivation is deferred to Appendix 4.6.1. The utilisation of the arbitrariness of variations in the resulting action principle (4.66) yields the corresponding PDEs. Variations with respect to the density ρ yield

$$\delta\rho:\quad \partial_t\phi=-\left(\frac{1}{2}|\boldsymbol{\nabla}\phi+\mathbf{v}|^2+(\rho U)_\rho+gz-\mathbf{v}\cdot\mathbf{u}\right), \tag{4.24a}$$

and the other variations give

$$\begin{aligned}
\delta\phi:&\quad \partial_t\rho=-\boldsymbol{\nabla}\cdot(\rho\mathbf{u}), && (4.24\text{b})\\
\delta(\rho\boldsymbol{\pi}):&\quad \partial_t\boldsymbol{l}=-(\mathbf{u}\cdot\boldsymbol{\nabla})\boldsymbol{l}, && (4.24\text{c})\\
\delta l_j:&\quad \partial_t(\rho\pi_j)=-\boldsymbol{\nabla}\cdot(\rho\pi_j\mathbf{u}), && (4.24\text{d})\\
(\delta\phi)_s:&\quad \rho_s\partial_t h=\rho_s\mathbf{u}_s\cdot\mathbf{n}_s, && (4.24\text{e})\\
(\delta h):&\quad \rho_s\partial_t(\phi)_s=-\rho_s\left[\frac{1}{2}|\mathbf{u}|^2+\frac{(\rho U)_\rho}{\rho_s}+gz-\mathbf{u}\cdot\mathbf{n}_s\boldsymbol{\pi}\cdot\partial_z\boldsymbol{l}\right]_s. && (4.24\text{f})
\end{aligned}$$

From (4.24) it is evident that the pairs of variables $\{\rho, \phi\}$, $\{\rho\boldsymbol{\pi}, \boldsymbol{l}\}$ and $\{\phi_s, h\}$ are canonically conjugated. The latter enables a new representation of the system (4.24) in terms of Hamilton's equations

$$\delta\rho : \quad \partial_t \phi = -\frac{\delta\mathcal{H}}{\delta\rho}, \tag{4.25a}$$

$$\delta\phi : \quad \partial_t \rho = \frac{\delta\mathcal{H}}{\delta\phi}, \tag{4.25b}$$

$$\delta(\rho\boldsymbol{\pi}) : \quad \partial_t \boldsymbol{l} = \frac{\delta\mathcal{H}}{\delta(\rho\boldsymbol{\pi})}, \tag{4.25c}$$

$$\delta l_j : \quad \partial_t(\rho\pi_j) = -\frac{\delta\mathcal{H}}{\delta l_j}, \tag{4.25d}$$

$$(\delta\phi)_s : \quad \partial_t h = \frac{1}{\rho_s}\left(\frac{\delta\mathcal{H}}{\delta\phi}\right)_s, \tag{4.25e}$$

$$(\delta h) : \quad \partial_t(\phi)_s = -\frac{1}{\rho_s}\frac{\delta\mathcal{H}}{\delta h}, \tag{4.25f}$$

where in the last equation the boundary condition (4.10) for $\mathcal{H}$ is used. Combining the corresponding canonical pairs we can form the canonical Poisson bracket

$$\begin{aligned}\{\mathcal{F}, \mathcal{H}\} = \int_D \frac{\delta\mathcal{F}}{\delta\rho}\frac{\delta\mathcal{H}}{\delta\phi} - \frac{\delta\mathcal{H}}{\delta\rho}\frac{\delta\mathcal{F}}{\delta\phi} + \frac{\delta\mathcal{H}}{\delta\boldsymbol{l}}\cdot\frac{\delta\mathcal{F}}{\delta(\rho\boldsymbol{\pi})} - \frac{\delta\mathcal{F}}{\delta\boldsymbol{l}}\cdot\frac{\delta\mathcal{H}}{\delta(\rho\boldsymbol{\pi})}\, d\Omega \\ + \int_{\Gamma_h} \frac{1}{\rho_s}\left[\left(\frac{\delta\mathcal{H}}{\delta\phi}\right)_s \frac{\delta\mathcal{F}}{\delta h} - \left(\frac{\delta\mathcal{F}}{\delta\phi}\right)_s \frac{\delta\mathcal{H}}{\delta h}\right] dx\, dy.\end{aligned} \tag{4.26}$$

Since the ρ_s multiplier is constant at the free-surface (density is constant at the free surface [40]), the bracket (4.26) is a canonical Poisson bracket, which automatically implies that all identities from (1.2) are satisfied, including the Jacobi identity.

The next step is to relate the Poisson bracket in Clebsch variables to the Poisson bracket in Eulerian variables (4.7), which we conjectured in Section 4.2. Since the Poisson bracket in Clebsch variables (4.26) is anti-symmetric and satisfies all identities from (1.2) we then have proven these relations also for the conjectured Poisson bracket (4.7).

For the transformation of the canonical Poisson bracket in the Clebsch variables (ρ, ϕ), $(\boldsymbol{l}, \rho\boldsymbol{\pi})$, (ϕ_s, h) to the Eulerian variables $\rho, \mathbf{u}$, and h, we need to relate functional derivatives in Clebsch variables to functional derivatives in Eulerian variables. For this we apply the functional derivative chain rule.

The variation of an arbitrary functional $\mathcal{F}$ expressed in terms of Clebsch variables is equal to

$$\delta\mathcal{F} = \int_D \frac{\delta\mathcal{F}}{\delta\rho}\delta\rho + \frac{\delta\mathcal{F}}{\delta\phi}\delta\phi + \frac{\delta\mathcal{F}}{\delta\boldsymbol{l}}\cdot\delta\boldsymbol{l} + \frac{\delta\mathcal{F}}{\delta(\rho\boldsymbol{\pi})}\cdot\delta(\rho\boldsymbol{\pi})\ d\Omega + \int_{\Gamma_h}\left(\frac{\delta\mathcal{F}}{\delta\phi}\right)_s\delta\phi_s + \frac{\delta\mathcal{F}}{\delta h}\delta h\ dxdy. \tag{4.27}$$

Taking into account that $\delta(\rho\boldsymbol{\pi}) = \rho\delta\boldsymbol{\pi} + \boldsymbol{\pi}\delta\rho$ we can transform (4.27) into

$$\delta\mathcal{F} = \int_D \left(\frac{\delta\mathcal{F}}{\delta\rho} + \frac{\delta\mathcal{F}}{\delta(\rho\boldsymbol{\pi})}\cdot\boldsymbol{\pi}\right)\delta\rho + \frac{\delta\mathcal{F}}{\delta\phi}\delta\phi + \frac{\delta\mathcal{F}}{\delta\boldsymbol{l}}\cdot\delta\boldsymbol{l} + \frac{\delta\mathcal{F}}{\delta(\rho\boldsymbol{\pi})}\cdot\delta(\boldsymbol{\pi})\rho \quad d\Omega + \int_{\Gamma_h}\left(\frac{\delta\mathcal{F}}{\delta\phi}\right)_s\delta\phi_s + \frac{\delta\mathcal{F}}{\delta h}\delta h\ dxdy. \tag{4.28}$$

Similarly, variations of the functional $\mathcal{F}$ in terms of the non-canonical Eulerian variables are equal to

$$\delta\mathcal{F} = \int_D \frac{\delta\mathcal{F}}{\delta\rho^e}\delta\rho^e + \frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\delta\mathbf{u}\ d\Omega + \int_{\Gamma_h}\frac{\delta\mathcal{F}}{\delta h^e}\delta h^e\ dxdy. \tag{4.29}$$

To avoid confusion where any Clebsch variable coincides with an Eulerian variable, an ‘e’ superscript is introduced to indicate that it belongs to the Eulerian set.

Substitution of the identity $\mathbf{u} = \boldsymbol{\nabla}\phi + \pi_j\boldsymbol{\nabla}l_j$ into (4.29) yields

$$\begin{aligned}\delta\mathcal{F} &= \int_D \frac{\delta\mathcal{F}}{\delta\rho^e}\delta\rho^e + \frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot(\boldsymbol{\nabla}\delta\phi + \delta\pi_j\boldsymbol{\nabla}l_j + \pi_j\boldsymbol{\nabla}\delta l_j)\ d\Omega + \int_{\Gamma_h}\frac{\delta\mathcal{F}}{\delta h^e}\delta h^e\ dxdy \\ &= \int_D \frac{\delta\mathcal{F}}{\delta\rho^e}\delta\rho^e + \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\delta\phi\right) - \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)\delta\phi \\ &\qquad + \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)\cdot\delta\pi_j\boldsymbol{\nabla}l_j + \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\pi_j\delta l_j\right) - \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\pi_j\right)\delta l_j\ d\Omega \\ &\quad + \int_{\Gamma_h}\frac{\delta\mathcal{F}}{\delta h^e}\delta h^e\ dxdy.\end{aligned} \tag{4.30}$$

Using the Gauss divergence theorem on the derivatives of the variational derivatives in combination with the solid wall boundary conditions we can transform this relation into

$$\delta\mathcal{F} = \int_D \frac{\delta\mathcal{F}}{\delta\rho^e}\delta\rho^e - \boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\delta\phi + \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)\cdot\delta\pi_j\boldsymbol{\nabla}l_j - \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\pi_j\right)\delta l_j\ d\Omega + \int_{\Gamma_h}\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_s\cdot\mathbf{n}_s(\delta\phi)_s + \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\pi_j\delta l_j\right)_s\cdot\mathbf{n}_s + \frac{\delta\mathcal{F}}{\delta h^e}\delta h^e\ dxdy. \tag{4.31}$$

If we compare now equivalent variations in (4.28) and (4.31) we obtain the following relations between functional derivatives in Clebsch and Eulerian variables

$$\frac{\delta\mathcal{F}}{\delta\rho} = \frac{\delta\mathcal{F}}{\delta\rho^e} - \frac{\pi_j}{\rho^e}\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)\cdot\boldsymbol{\nabla}l_j, \tag{4.32a}$$

$$\frac{\delta\mathcal{F}}{\delta\phi} = -\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}}, \tag{4.32b}$$

$$\left(\frac{\delta\mathcal{F}}{\delta l}\right)_j = -\boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\pi_j\right), \tag{4.32c}$$

$$\left(\frac{\delta\mathcal{F}}{\delta(\rho\boldsymbol{\pi})}\right)_j = \frac{1}{\rho^e}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j, \tag{4.32d}$$

$$\left(\frac{\delta\mathcal{F}}{\delta\phi}\right)_s = \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_s\cdot\mathbf{n}_s, \tag{4.32e}$$

$$\frac{\delta\mathcal{F}}{\delta h} = \frac{\delta\mathcal{F}}{\delta h^e}, \tag{4.32f}$$

where we used the fact that variation of Lagrangian labels at the free surface is zero, $(\delta l_j)_s = 0$.

Substitution of (4.32) in (4.26), while dropping the superscripts, leads to a new Poisson bracket based on functional derivatives in Eulerian variables

$$\begin{aligned}
\{\mathcal{F},\mathcal{H}\} = &\int_D \left(\frac{\delta\mathcal{H}}{\delta\rho} - \frac{\pi_j}{\rho}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}} - \left(\frac{\delta\mathcal{F}}{\delta\rho} - \frac{\pi_j}{\rho}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\\
&- \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\pi_j\right)\frac{1}{\rho}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j + \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\pi_j\right)\frac{1}{\rho}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\boldsymbol{\nabla}l_j\; d\Omega\\
&+ \int_{\Gamma_h}\frac{1}{\rho_s}\left[\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_s\cdot\mathbf{n}_s\frac{\delta\mathcal{F}}{\delta h} - \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_s\cdot\mathbf{n}_s\frac{\delta\mathcal{H}}{\delta h}\right] dxdy.
\end{aligned} \tag{4.33}$$

Combining similar terms, we can write the Poisson bracket (4.33) as

$$\begin{aligned}
&\int_D \frac{\delta\mathcal{H}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}} - \frac{\delta\mathcal{F}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\; d\Omega +\\
&\int_D -\left(\frac{\pi_j}{\rho}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\left(\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right) + \left(\frac{\pi_j}{\rho}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\left(\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)\\
&\qquad - \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\pi_j\right)\left(\frac{1}{\rho}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right) + \boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\pi_j\right)\left(\frac{1}{\rho}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\; d\Omega +\\
&\int_{\Gamma_h}\frac{1}{\rho_s}\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\frac{\delta\mathcal{F}}{\delta h} - \frac{\delta\mathcal{F}}{\delta\mathbf{u}}\frac{\delta\mathcal{H}}{\delta h}\right)_s\cdot\mathbf{n}_s\; dxdy.
\end{aligned} \tag{4.34}$$

In (4.34) we can already recognise the first two terms in the Poisson bracket (4.7). After some algebraic manipulations we obtain for the second integral in (4.34) the following expression

$$\int_D -\left(\frac{\pi_j}{\rho}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\boldsymbol{\nabla}l_j\right)\left(\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)+\left(\frac{\pi_j}{\rho}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\left(\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)$$
$$-\boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\pi_j\right)\left(\frac{1}{\rho}\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)+\boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\pi_j\right)\left(\frac{1}{\rho}\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\cdot\boldsymbol{\nabla}l_j\right)\,d\Omega \tag{4.35a}$$

$$=\int_D\frac{1}{\rho}\left[\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_k\nabla_k\left(\pi_j\nabla_i l_j\right)\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_i-\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_k\nabla_k\left(\pi_j\nabla_i l_j\right)\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_i\right]\,d\Omega. \tag{4.35b}$$

The details of this derivation are provided in the Appendix 4.6.2. This relation is exactly the second, vortical contribution, in the Poisson bracket (4.7)

$$\int_D\frac{\boldsymbol{\nabla}\times\mathbf{u}}{\rho}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\times\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\,d\Omega. \tag{4.36}$$

Combining all terms in (4.34) gives then the Poisson bracket in non-canonical Eulerian variables that we conjectured in Section 4.2

$$\{\mathcal{F},\mathcal{H}\}=\int_D\frac{\delta\mathcal{H}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{F}}{\delta\mathbf{u}}-\frac{\delta\mathcal{F}}{\delta\rho}\boldsymbol{\nabla}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}+\frac{\boldsymbol{\nabla}\times\mathbf{u}}{\rho}\cdot\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\times\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\,d\Omega$$
$$+\int_{\Gamma_s}\frac{1}{\rho_s}\left[\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\frac{\delta\mathcal{F}}{\delta h}-\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\frac{\delta\mathcal{H}}{\delta h}\right]\cdot\hat{n}_s\,dS. \tag{4.37}$$

Since in every step of the derivation all identities from (1.2) are satisfied for the intermediate Poisson brackets, this also applies to the final result, the Poisson bracket (4.37), which is the same as the Poisson bracket (4.7), which was conjectured in Section 4.2.

4.4 Linearised Hamiltonian framework

4.4.1 Governing equations

Next, we introduce a linearisation of the compressible Euler equations (4.1) around a rest state ($\mathbf{u}_0=\mathbf{0},\rho_0(z),h_0=0$) with $\mathbf{u}=\mathbf{u}_0+\bar{\mathbf{u}}$, $\rho=$

$\rho_0(z) + \bar{\rho}$, $P = P_0(z) + \bar{P}$ and $h = 0 + \bar{h}$, where $\bar{\mathbf{u}}(x, y, z, t)$, $\bar{\rho}(x, y, z, t)$, $\bar{P}(x, y, z, t)$ and $\bar{h}(x, y, t)$ are, respectively, a small perturbation velocity, density, pressure and free surface height. Note, that at the rest state the background velocity $\mathbf{u}_0$ and free surface height h_0 are zero, and the background density $\rho_0(z)$ is z-dependent only. Additionally, the time dependent domain D transforms into the time independent domain $\bar{D}$ with a mean free surface $\bar{\Gamma}_s$ at the top, where $\bar{\Gamma}_s$ ($z = 0$ plane) is the linearised version of the free surface Γ_s.

The nonlinear pressure is proportional to the density, due to the barotropic pressure Ansatz, and thus, the following nonlinear equation of state is adapted

$$P = P_c + c_0^2(\rho - \rho_s). \tag{4.38}$$

where c_0 is the acoustic (constant) wave speed and ρ_s is the constant density at the free surface ($\rho|_{z=0} = \rho_s$). Due to the linearisation (4.38) becomes

$$P \cong P_c + c_0^2(\rho_0(z) - \rho_s) + c_0^2\bar{\rho}. \tag{4.39}$$

From the latter it follows that the rest state pressure is equal to $P_0(z) = P_c + c_0^2(\rho_0(z) - \rho_s)$ and the perturbation pressure $\bar{P} = c_0^2\bar{\rho}$.

Further, the hydrostatic balance equation at the rest state, $(\mathbf{u}_0 = \mathbf{0}, \rho_0(z), h_0 = 0, P_0)$, is as follows

$$-\nabla\big(P_c + (c_0^2\rho_0(z) - \rho_s)\big) + \rho_0\mathbf{g} = 0, \tag{4.40}$$

with $\mathbf{g} = (0, 0, -g)^T$. The background density is stratified only in the z-direction, thus we have

$$-c_0^2\frac{d\rho_0(z)}{dz} = \rho_0 g. \tag{4.41}$$

Taking into account (4.41) and the boundary condition at the free surface ($\rho_0|_{z=0} = \rho_s$) the following profile for the background density can be derived

$$\rho_0 = \rho_s \exp(-gz/c_0^2). \tag{4.42}$$

The linearisation applied to the nonlinear Euler equations (4.1) results in the linear compressible Euler equations with a free surface

$$\rho_0\frac{\partial\bar{\mathbf{u}}}{\partial t} = -\nabla(c_0^2\bar{\rho}) - 2\mathbf{\Omega}\times\bar{\mathbf{u}} + \bar{\rho}\mathbf{g}, \tag{4.43a}$$

$$\frac{\partial\bar{\rho}}{\partial t} = -\nabla\cdot(\rho_0\bar{\mathbf{u}}), \tag{4.43b}$$

$$\frac{\partial\bar{h}}{\partial t} = \bar{w}_s \quad \text{at} \quad z = 0, \tag{4.43c}$$

where $\bar{w}_s = \mathbf{u} \cdot \bar{\boldsymbol{n}}_s$ is the third component of the velocity at the free surface and $\bar{\boldsymbol{n}}_s = (0, 0, 1)$ is a linearised version of the non-linear free surface normal $\hat{\boldsymbol{n}}_s$.

A Taylor expansion of the nonlinear pressure $P(x, y, z = h, t)$ around $z = 0$ results in

$$P(x, y, z = h, t) \cong P(x, y, z = 0, t) + \frac{dP_0(z)}{dz}|_{z=0}\bar{h} = P_c, \tag{4.44}$$

and due to the linearisation we have

$$P(x, y, z = 0, t) = P_0(z)|_{z=0} + \bar{P}(x, y, z = 0, t), \tag{4.45}$$

where $P_0(z)$ is the hydrostatic pressure at the rest state, since $P_0|_{z=0} = P_c$. The latter results in

$$-\frac{dP_0(z)}{dz}|_{z=0}\bar{h} = \bar{P}(x, y, z = 0, t), \tag{4.46}$$

which, in combination with (4.41) and the expression for the rest state pressure $\bar{P}_0(z) = P_c + c_0^2(\rho_0(z) - \rho_s)$, is further transformed into

$$c_0^2\bar{\rho} \equiv \bar{P} = \rho_s g\bar{h} \quad \text{at} \quad z = 0. \tag{4.47}$$

The resulting expression (4.47) is the boundary condition for the perturbation pressure at the free surface.
A similar boundary condition for the pressure at the bottom can be derived if the linearised momentum equation from (4.43a) is multiplied by a normal to a bottom topography $\bar{\mathbf{n}}_b$

$$\nabla(c_0^2\bar{\rho}) \cdot \bar{\mathbf{n}}_b = \bar{\rho}\mathbf{g} \cdot \bar{\mathbf{n}}_b. \tag{4.48}$$

An alternative representation of (4.43), (4.43a), can be derived if the product rule of differentiation is applied in combination with the hydrostatic balance equation (4.40)

$$\frac{\partial \bar{\mathbf{u}}}{\partial t} = -\nabla\left(\frac{c_0^2}{\rho_0}\bar{\rho}\right) - \frac{2\boldsymbol{\Omega}}{\rho_0} \times \bar{\mathbf{u}}, \tag{4.49a}$$

$$\frac{\partial \bar{\rho}}{\partial t} = -\nabla \cdot (\rho_0\bar{\mathbf{u}}), \tag{4.49b}$$

$$\frac{\partial \bar{h}}{\partial t} = \bar{w}_s \quad \text{at} \quad z = 0. \tag{4.49c}$$

4.4.2 Hamiltonian formalism for linearised compressible flow

In this section, the Hamiltonian formulation for linear compressible fluid flow in a domain $\bar{D}$ is introduced. The connection with corresponding linear PDE based model (4.43) is also discussed.

The Hamiltonian formulation for linear compressible fluid flow governed by (4.43) in the domain $\bar{D}$ is described as follows

$$\begin{aligned}\frac{d\mathcal{F}}{dt} = \{\mathcal{F},\mathcal{H}\} = &\int_{\bar{D}} \frac{\delta\mathcal{H}}{\delta\bar{\rho}}\nabla\cdot\frac{\delta\mathcal{F}}{\delta\bar{\mathbf{u}}} - \frac{\delta\mathcal{F}}{\delta\bar{\rho}}\nabla\cdot\frac{\delta\mathcal{H}}{\delta\bar{\mathbf{u}}} + \frac{2\mathbf{\Omega}}{\rho_0}\cdot\frac{\delta\mathcal{F}}{\delta\bar{\mathbf{u}}}\times\frac{\delta\mathcal{H}}{\delta\bar{\mathbf{u}}}\,d\Omega \\ &+\int_{\bar{\Gamma}_s}\frac{1}{\rho_s}\left[\frac{\delta\mathcal{F}}{\delta\bar{h}}\bar{\mathbf{n}}_s\cdot\frac{\delta\mathcal{H}}{\delta\bar{\mathbf{u}}} - \frac{\delta\mathcal{F}}{\delta\bar{\mathbf{u}}}\cdot\bar{\mathbf{n}}_s\frac{\delta\mathcal{H}}{\delta\bar{h}}\right]_{z=0} dS, \qquad (4.50a)\end{aligned}$$

with Hamiltonian energy functional

$$\mathcal{H} = \mathcal{H}[\bar{\mathbf{u}},\bar{\rho},\bar{h}] \equiv \int_{\bar{D}} \frac{1}{2}\rho_0|\bar{\mathbf{u}}|^2 + \frac{c_0^2}{2\rho_0}\bar{\rho}^2 d\Omega + \int_{\bar{\Gamma}_s}\left[\frac{1}{2}\rho_s g\bar{h}^2\right]_{z=0} dS. \qquad (4.50b)$$

The latter Hamiltonian formulation is a free surface extension of the linear Hamiltonian formulation discussed in Chapter 2 and a linearised version of the Hamiltonian formulation stated in (4.7).

The definition of the functional derivative of the Hamiltonian $\mathcal{H}$ is as follows

$$\begin{aligned}\delta\mathcal{H} \equiv \lim_{\epsilon\to 0}&\frac{\mathcal{H}[\bar{\mathbf{u}}+\epsilon\delta\bar{\mathbf{u}},\bar{\rho}+\epsilon\delta\bar{\rho},\bar{h}+\epsilon\delta\bar{h}] - \mathcal{H}[\bar{\mathbf{u}},\bar{\rho},\bar{h}]}{\epsilon} = \\ &\int_{\bar{D}}\frac{\delta\mathcal{H}}{\delta\bar{\mathbf{u}}}\cdot\delta\bar{\mathbf{u}} + \frac{\delta\mathcal{H}}{\delta\bar{\rho}}\delta\bar{\rho}\,d\Omega + \int_{\bar{\Gamma}_s}\left[\frac{\delta\mathcal{H}}{\delta\bar{h}}\delta\bar{h}\right]_{z=0} dS. \qquad (4.51)\end{aligned}$$

Hence, the functional derivatives of $\mathcal{H}$ are equal to

$$\frac{\delta\mathcal{H}}{\delta\bar{\mathbf{u}}} = \rho_0\bar{\mathbf{u}}, \quad \frac{\delta\mathcal{H}}{\delta\bar{\rho}} = \frac{c_0^2}{\rho_0}\bar{\rho},$$

$$\frac{\delta\mathcal{H}}{\delta\bar{h}} = \rho_s g\bar{h} \quad \text{at} \quad \bar{\Gamma}_s. \qquad (4.52)$$

The linear Poisson bracket stated above satisfies all required properties defined in (1.2): skew-symmetry can be recognised from the structure of the bracket; the bracket is bilinear, thus the linearity and Leibniz identity are automatically satisfied; and, the Jacobi identity can be checked directly.

Next, the equivalence between the Hamiltonian formulation (4.50) and the PDE representation (4.43) of compressible fluid flow in the domain $\bar{D}$ is shown.

The momentum, continuity and free surface dynamics equations are obtained if the following functionals are chosen

$$\mathcal{F}_{\bar{\mathbf{u}}} \equiv \int_{\bar{D}} \bar{\mathbf{u}}(\mathbf{x},t) \cdot \mathbf{\Phi}(\mathbf{x}) d\Omega, \tag{4.53a}$$

$$\mathcal{F}_{\bar{\rho}} \equiv \int_{\bar{D}} \bar{\rho}(\mathbf{x},t) \phi(\mathbf{x}) d\Omega, \tag{4.53b}$$

$$\mathcal{F}_{\bar{h}} \equiv \int_{\bar{\Gamma}_s} \bar{h}(\mathbf{x},t) \phi^s(\mathbf{x}) dS, \tag{4.53c}$$

with $\mathbf{\Phi} \in \bar{\mathcal{Y}}$, $\phi \in \bar{\mathcal{Q}}$ and $\phi_s = \mathbf{\Phi} \cdot \bar{\mathbf{n}}_s$ arbitrary test functions, where

$$\bar{\mathcal{Q}} = \{\phi \in L^2(\bar{D})\} \ , \bar{\mathcal{Y}} = \{\mathbf{\Phi} \in (L^2(\bar{D}))^3 : \ \mathbf{n} \cdot \mathbf{\Phi} = 0 \text{ at} \cup_{i=1}^{5} \partial \bar{D}_i\}, \tag{4.54}$$

and $L^2(\bar{D})$ is the space of square integrable functions on $\bar{D}$. To incorporate the appropriate boundary conditions at $\cup_{i=1}^{5} \partial \bar{D}_i$ we restrict the space for the test functions $\mathbf{\Phi}$ at the solid-wall boundaries, whereas for the free surface boundary condition the functional $\mathcal{F}$ is restricted as follows

$$\frac{\delta \mathcal{F}}{\delta \bar{\rho}} = \frac{1}{\rho_s} \frac{\delta \mathcal{F}}{\delta \bar{h}}. \tag{4.55}$$

In particular, for the Hamiltonian energy functional $\mathcal{H}$ the latter restriction results in

$$\frac{\delta \mathcal{H}}{\delta \bar{\rho}} - \frac{1}{\rho_s} \frac{\delta \mathcal{H}}{\delta \bar{h}} \equiv \frac{c_0^2}{\rho_s} \bar{\rho} - \frac{1}{\rho_s} \rho_s g \bar{h} = 0, \tag{4.56}$$

which exactly coincides with the free surface boundary condition for the pressure (4.47).

The corresponding functional derivatives of (4.53a), (4.53b) and (4.53c) are given as follows

$$\begin{aligned} &\frac{\delta \mathcal{F}_{\bar{\mathbf{u}}}}{\delta \bar{\mathbf{u}}} = \mathbf{\Phi}(\mathbf{x}), \text{ with } && \frac{\delta \mathcal{F}_{\bar{\mathbf{u}}}}{\delta \bar{\mathbf{u}}} \cdot \bar{\mathbf{n}}_i = 0 \text{ at } \cup_i \partial D_i \\ &\frac{\delta \mathcal{F}_{\bar{\rho}}}{\delta \bar{\rho}} = \phi(\mathbf{x}), && \frac{\delta \mathcal{F}_{\eta}}{\delta \bar{h}} = \phi^s(\mathbf{x}). \end{aligned} \tag{4.57}$$

Using functionals (4.53b) and (4.53c) with corresponding functional derivatives (4.57) and (4.52), in the bracket formulation (4.50a), the equations of the continuity (4.49b) and free surface dynamics (4.49c) are obtained.

In contrast to the latter, derivation of the momentum equation is less straightforward. After the functional (4.53a) is substituted into the bracket (4.50a) and integration by parts on the first term is performed, we arrive at

$$\frac{d}{dt}\int_{\bar{D}} \bar{\mathbf{u}} \cdot \mathbf{\Phi}\, d\Omega = -\int_{\bar{D}} \nabla\left(\frac{c_0^2}{\rho_0}\bar{\rho}\right) \cdot \mathbf{\Phi} + \frac{2\mathbf{\Omega}}{\rho_0} \times \bar{\mathbf{u}} \cdot \mathbf{\Phi}\, d\Omega + \int_{\partial\bar{D}/\bar{\Gamma}_s} \frac{\delta\mathcal{H}}{\delta\bar{\rho}} \mathbf{\Phi} \cdot \bar{\mathbf{n}} d\Gamma$$
$$- \int_{\bar{\Gamma}_s} \left[\left(\frac{\delta\mathcal{H}}{\delta\bar{\rho}} - \frac{1}{\rho_s}\frac{\delta\mathcal{H}}{\delta\bar{h}}\right)\right] \mathbf{\Phi} \cdot \bar{\mathbf{n}}_s dS. \tag{4.58}$$

The arbitrary test function $\mathbf{\Phi}$ is a constituent of $\bar{\mathcal{Y}}$, thus the first boundary integral can be set to zero. The second boundary integral is also zero due to the introduced restriction on the functionals (4.55).

Finally, using the arbitrariness of test function $\mathbf{\Phi}$ we get

$$\frac{d\bar{\mathbf{u}}}{dt} = -\nabla\left(\frac{c_0^2}{\rho_0}\bar{\rho}\right) - \frac{2\mathbf{\Omega}}{\rho_0} \times \bar{\mathbf{u}}, \tag{4.59}$$

which coincides exactly with the momentum equation in (4.49), the alternative representation of (4.43). The latter ensures the correspondence of the Hamiltonian formulation (4.50) to the linear Euler equations (4.43).

4.5 Discussion and future plans

A nonlinear Hamiltonian formulation for the Euler equations describing an inviscid compressible fluid flow in a rotating domain with a free-surface and solid walls was presented. First, a corresponding (generalised) Poisson bracket and Hamiltonian functional were conjectured, and, as an initial verification step, the nonlinear Euler equations with a free surface were derived from it. Next, the conjectured Poisson bracket was derived starting from a variational principle using the following steps: (i) a Lagrangian functional was introduced expressed in terms of Eulerian variables; (ii) the Lagrangian functional in Eulerian variables was subsequently transformed into a Lagrangian functional in Clebsch canonical variables; (iii) an action principle was applied to the transformed Lagrangian, which resulted in a canonical Poisson bracket in Clebsch variables and (iv) the canonical Poisson bracket in the canonical Clebsch variables was reduced to a Poisson bracket in Eulerian variables. This derivation ensured satisfaction of the Jacobi identity in every step in the derivation, and thus, the (generalised) Poisson bracket, which was originally conjectured, also satisfies the required Jacobi identity.

Additionally, a linear Hamiltonian formulation for a compressible fluid flow in a domain with a free surface was introduced. The presented formulation was derived by applying perturbation theory on the nonlinear equations introduced earlier. Furthermore, the linear Hamiltonian formalism was demonstrated to be a free surface extension of the Hamiltonian formulation discussed in Chapter 2

The next step will be to develop a numerical discretisation based on the newly derived (non)linear Hamiltonian formulation followed by the Dirac technique discussed in Chapter 2 to obtain a Hamiltonian discretisation of the incompressible Euler equations with a free surface. In both linear and nonlinear case, the main difficulty of the discretisation is in the satisfaction of boundary conditions at the free surface. Also, the numerical flux functions should be such that the resulting DGFEM discretisation is nonetheless a Hamiltonian system.

4.6 Appendix

4.6.1 Calculation of the variations of the Lagrangian

The detailed calculation of the variation of the Lagrangian (4.22) is given below.

$$
\begin{aligned}
&0 = \delta \int_0^T L(\boldsymbol{l}, \boldsymbol{\pi}, \rho, \phi, h) dt = \\
&\int_0^T \int_{\Gamma_h} \int_b^{h+b} \rho(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \delta(\boldsymbol{\nabla}\phi + \mathbf{v}) + \frac{1}{2}|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 \delta\rho + gz\delta\rho + (\rho U)_\rho \delta\rho \\
&\qquad\qquad + \delta(\rho\partial_t\phi) + \delta(\rho\boldsymbol{\pi}) \cdot \partial_t \boldsymbol{l} + \rho\boldsymbol{\pi} \cdot \partial_t \delta\boldsymbol{l} \; dxdydz \\
&\quad + \int_{\Gamma_h} \left[\frac{1}{2}\rho|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + \rho g z + \rho\partial_t\phi + \rho\boldsymbol{\pi} \cdot (\partial_t \boldsymbol{l}) \right]_s \delta h \; dxdy\, dt,
\end{aligned}
\tag{4.60}
$$

where, after combining terms with the same variation the right hand side of (4.60) can be written as

$$\begin{aligned}
&\int_0^T \int_{\Gamma_h} \int_b^{h+b} \rho(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot (\boldsymbol{\nabla}\delta\phi + \delta\pi_j \boldsymbol{\nabla} l_j + \pi_j \boldsymbol{\nabla}(\delta l_j)) + \rho\partial_t(\delta\phi) \\
&\quad + \delta(\rho\boldsymbol{\pi}) \cdot \partial_t \boldsymbol{l} + \rho\boldsymbol{\pi} \cdot \partial_t \delta\boldsymbol{l} + \left(\frac{1}{2}|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + gz + \partial_t\phi\right) \delta\rho \; dxdydz \\
&+ \int_{\Gamma_h} \left[\frac{1}{2}\rho|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + \rho g z + \rho\partial_t\phi + \rho\boldsymbol{\pi} \cdot (\partial_t \boldsymbol{l})\right]_s \delta h \, dxdy \; dt.
\end{aligned} \tag{4.61}$$

The next step is to remove derivatives of the variations by integrating by parts, resulting in

$$\begin{aligned}
&\int_0^T \int_{\Gamma_h} \int_b^{h+b} -\boldsymbol{\nabla} \cdot (\rho(\boldsymbol{\nabla}\phi + \mathbf{v}))\delta\phi \; dxdydz \\
&\quad + \int_{\Gamma_h} (\delta\phi)_s \rho_s ((\boldsymbol{\nabla}\phi + \mathbf{v})_s) \cdot \boldsymbol{n}_s \; dxdy \\
&\quad + \int_{\Gamma_h} \int_b^{h+b} \delta(\rho\pi_j)(\boldsymbol{\nabla} l_j \cdot (\boldsymbol{\nabla}\phi + \mathbf{v})) - \delta\rho(\pi_j(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \boldsymbol{\nabla} l_j) \\
&\qquad\quad - \delta l_j \boldsymbol{\nabla} \cdot (\rho(\boldsymbol{\nabla}\phi + \mathbf{v})\pi_j) \; dxdydz \\
&\quad + \int_{\Gamma_h} \rho_s (\boldsymbol{\nabla}\phi + \mathbf{v})_s (\pi_j)_s \cdot \boldsymbol{n}_s (\delta l_j)_s \, dxdy \\
&\quad + \int_{\Gamma_h} \int_b^{h+b} \left(\frac{1}{2}|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + gz + \partial_t\phi\right) \delta\rho \; dxdydz \\
&\quad + \frac{d}{dt} \int_{\Gamma_h} \int_b^{h+b} \rho\delta\phi \; dxdydz - \int_{\Gamma_h} \int_b^{h+b} \delta\phi\partial_t\rho \; dxdydz \\
&\quad - \int_{\Gamma_h} \rho_s (\delta\phi)_s \partial_t h \; dxdy + \int_{\Gamma_h} \int_b^{h+b} \delta(\rho\boldsymbol{\pi}) \cdot (\partial_t \boldsymbol{l}) \; dxdydz \\
&\quad + \frac{d}{dt} \int_{\Gamma_h} \int_b^{h+b} \rho\boldsymbol{\pi} \cdot \delta\boldsymbol{l} \; dxdydz - \int_{\Gamma_h} \int_b^{h+b} \delta\boldsymbol{l} \cdot \partial_t(\rho\boldsymbol{\pi}) \; dxdydz \\
&\quad - \int_{\Gamma_h} \rho_s \boldsymbol{\pi}_s \cdot (\delta\boldsymbol{l})_s \partial_t h \\
&\qquad + \left[\frac{1}{2}\rho|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + \rho g z + \rho\partial_t\phi + \rho\boldsymbol{\pi} \cdot (\partial_t \boldsymbol{l})\right]_s \delta h dxdydt.
\end{aligned} \tag{4.62}$$

Here, the Gauss divergence theorem was used on the following contributions

$$\int_D \rho(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \boldsymbol{\nabla}(\delta\phi)\; d\Omega = \int_{\Gamma_s} (\delta\phi)_s \rho_s (\boldsymbol{\nabla}\phi + \mathbf{v})_s \cdot \hat{\boldsymbol{n}}_s\; dS - \int_D \boldsymbol{\nabla} \cdot (\rho(\boldsymbol{\nabla}\phi + \mathbf{v}))\delta\phi\; d\Omega, \tag{4.63a}$$

$$\int_D \rho(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \pi_j \boldsymbol{\nabla}(\delta l_j) d\Omega = \int_{\Gamma_s} (\delta l_j)_s \rho_s (\boldsymbol{\nabla}\phi + \mathbf{v})_s (\pi_j)_s \cdot \hat{\boldsymbol{n}}_s\; dS - \int_D \boldsymbol{\nabla} \cdot (\rho(\boldsymbol{\nabla}\phi + \mathbf{v})\pi_j)\delta l_j\; d\Omega, \tag{4.63b}$$

where the solid wall boundary condition is applied at $\cup_{i=1}^{5} \partial D_i$ and the Einstein summation convention is used on repeated indices. Since the free surface h depends on time, viz. $h = h(x, y, t)$, we have applied the Leibnitz theorem to the following integrals

$$\int_b^{h+b} \rho \frac{\partial}{\partial t}(\delta\phi)\, dz = \frac{d}{dt}\int_b^{h+b} \rho\delta\phi\, dz - \int_b^{h+b} \delta\phi\partial_t\rho dz - \rho_s(\delta\phi)_s\partial_t h, \tag{4.63c}$$

$$\int_b^{h+b} \rho\boldsymbol{\pi} \cdot \frac{\partial}{\partial t}(\delta\boldsymbol{l})\, dz = \frac{d}{dt}\int_b^{h+b} \rho\boldsymbol{\pi} \cdot \delta\boldsymbol{l}\, dz - \int_b^{h+b} \delta\boldsymbol{l} \cdot \partial_t(\rho\boldsymbol{\pi})\; dz - \rho_s\boldsymbol{\pi}_s \cdot (\delta\boldsymbol{l})_s\partial_t h \tag{4.63d}$$

A further simplification of (4.62) is obtained via integration by parts and using the Gauss divergence theorem. Collecting contributions with the same variation and integrating by parts again, finally results in

$$\begin{aligned}
\int_0^T \int_{\Gamma_h} \int_b^{h+b} & \Big(\frac{1}{2}|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + gz + \partial_t\phi - \pi_j(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \boldsymbol{\nabla} l_j\Big)\,\delta\rho \\
& - \Big(\boldsymbol{\nabla} \cdot (\rho(\boldsymbol{\nabla}\phi + \mathbf{v})) + \partial_t\rho\Big)\delta\phi \\
& + \Big((\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \boldsymbol{\nabla} l_j + \partial_t l_j\Big)\delta(\pi_j\rho) \\
& - \Big(\boldsymbol{\nabla} \cdot (\rho\pi_j(\boldsymbol{\nabla}\phi + \mathbf{v})) + \partial_t(\rho\pi_j)\Big)\delta l_j \; dxdydz \; dt \\
+ \int_{\Gamma_h} \int_b^{h+b} & \rho\boldsymbol{\pi} \cdot \delta\boldsymbol{l}|_0^T + \delta(\rho\phi)|_0^T \; dxdydz \\
+ \int_0^T \int_{\Gamma_h} & \rho_s\left((\boldsymbol{\nabla}\phi + \mathbf{v})_s \cdot \mathbf{n}_s - \partial_t h\right)(\delta\phi)_s \\
& + \left((\rho\boldsymbol{\pi})_s(\boldsymbol{\nabla}\phi + \mathbf{v})_s \cdot \mathbf{n}_s - (\rho\boldsymbol{\pi})_s\partial_t h\right) \cdot (\delta\boldsymbol{l})_s \\
& + \left[\frac{1}{2}\rho|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + \rho g z + \rho\partial_t\phi + \rho\boldsymbol{\pi} \cdot (\partial_t\boldsymbol{l})\right]_s \delta h \; dxdy \; dt.
\end{aligned} \tag{4.64}$$

Equation (4.63d) can be used to connect the variations $(\delta\boldsymbol{l})_s$ and $(\partial_t\boldsymbol{l})_s$ at the free-surface $h = h(x, y, t)$ and after assuming that $\delta\boldsymbol{l}|_0^T = 0$ and $\delta(\rho\phi)|_0^T = 0$ at the end points gives the expression

$$\begin{aligned}
\int_0^T \int_{\Gamma_h} \int_b^{h+b} & \Big(\frac{1}{2}|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + gz + \partial_t\phi - \pi_j(\boldsymbol{\nabla}\phi + \mathbf{v}) \cdot \boldsymbol{\nabla} l_j\Big)\,\delta\rho \\
& - \Big(\boldsymbol{\nabla} \cdot \rho(\boldsymbol{\nabla}\phi + \mathbf{v}) + \partial_t\rho\Big)\delta\phi + \Big((\boldsymbol{\nabla}\phi + \mathbf{v})\boldsymbol{\nabla} l_j + \partial_t l_j\Big)\delta(\pi_j\rho) \\
& - \Big(\boldsymbol{\nabla} \cdot (\rho\pi_j(\boldsymbol{\nabla}\phi + \mathbf{v})) + \partial_t(\rho\pi_j)\Big)\delta l_j \; dxdydz \\
+ \int_{\Gamma_h} & \rho_s\Big((\boldsymbol{\nabla}\phi + \mathbf{v})_s \cdot \mathbf{n}_s - \partial_t h\Big)(\delta\phi)_s \\
& - \Big(\rho_s\boldsymbol{\pi}_s(\boldsymbol{\nabla}\phi + \mathbf{v})_s \cdot \mathbf{n}_s - \rho_s\boldsymbol{\pi}_s\partial_t h\Big) \cdot \left[\delta h\partial_z\boldsymbol{l}\right]_s + \\
& \left[\frac{1}{2}\rho|\boldsymbol{\nabla}\phi + \mathbf{v}|^2 + (\rho U)_\rho + \rho g z + \rho\partial_t\phi\right]_s \delta h + \rho_s\boldsymbol{\pi}_s \cdot \left[-\partial_t h\partial_z\boldsymbol{l}\right]_s \delta h \; dxdy \; dt.
\end{aligned} \tag{4.65}$$

Cancelling terms with opposite sign then yields the action principle in Clebsch variables

$$\begin{aligned}
&\int_0^T \int_{\Gamma_h} \int_b^{h+b} \Big(\frac{1}{2}|\boldsymbol{\nabla}\phi+\mathbf{v}|^2 + (\rho U)_\rho + gz + \partial_t\phi - \pi_j(\boldsymbol{\nabla}\phi+\mathbf{v})\boldsymbol{\nabla} l_j\Big)\,\delta\rho \\
&\qquad\qquad - \Big(\boldsymbol{\nabla}\cdot\rho(\boldsymbol{\nabla}\phi+\mathbf{v}) + \partial_t\rho\Big)\delta\phi + \Big((\boldsymbol{\nabla}\phi+\mathbf{v})\boldsymbol{\nabla} l_j + \partial_t l_j\Big)\delta(\pi_j\rho) \\
&\qquad\qquad - \Big(\boldsymbol{\nabla}\cdot(\rho\pi_j(\boldsymbol{\nabla}\phi+\mathbf{v})) + \partial_t(\rho\pi_j)\Big)\delta l_j \; dxdydz \\
&+ \int_{\Gamma_h} \rho_s\Big((\boldsymbol{\nabla}\phi+\mathbf{v})_s\cdot\mathbf{n}_s - \partial_t h\Big)(\delta\phi)_s + \\
&\Big(\frac{1}{2}\rho|\boldsymbol{\nabla}\phi+\mathbf{v}|^2 + (\rho U)_\rho + \rho g z + \rho\partial_t\phi - \rho(\boldsymbol{\nabla}\phi+\mathbf{v})\cdot\mathbf{n}_s\boldsymbol{\pi}\cdot\partial_z\boldsymbol{l}\Big)_s \delta h\, dxdy\; dt.
\end{aligned} \tag{4.66}$$

From the arbitrary variations for $\delta\rho$, $\delta\phi$, $\delta(\pi_j\rho)$, δl_j, $(\delta\phi)_s$ and δh the corresponding PDEs given in (4.24) are derived.

4.6.2 Rotational velocity

In this Appendix we give a detailed description of the algebraic manipulations on (4.35a) that will result in (4.35b).

After integration by parts of every term on the left hand side in (4.35a) we obtain the following relation

$$\begin{aligned}
&\int_D -\boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\frac{\pi_j}{\rho}\nabla_i l_j\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_i\right) + \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_k \nabla_k\left(\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_i \nabla_i l_j \frac{\pi_j}{\rho}\right) \\
&\qquad +\boldsymbol{\nabla}\cdot\left(\frac{\pi_j}{\rho}\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_i \nabla_i l_j \frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right) - \left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_k \nabla_k\left(\frac{\pi_j}{\rho}\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_i \nabla_i l_j\right) \\
&\qquad -\boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\frac{\pi_j}{\rho}\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_i \nabla_i l_j\right) + \left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_k \pi_j\nabla_k\left(\frac{1}{\rho}\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_i \nabla_i l_j\right) \\
&\qquad +\boldsymbol{\nabla}\cdot\left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\frac{\pi_j}{\rho}\nabla_i l_j\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_i\right) - \left(\frac{\delta\mathcal{F}}{\delta\mathbf{u}}\right)_k \pi_j\nabla_k\left(\frac{1}{\rho}\left(\frac{\delta\mathcal{H}}{\delta\mathbf{u}}\right)_i \nabla_i l_j\right)\; d\Omega,
\end{aligned} \tag{4.67}$$

with $\nabla_k = \partial/\partial x_k$. We can further simplify (4.67) by cancelling contributions with opposite sign, resulting in

$$\int_D \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_k \nabla_k \left(\left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_i \nabla_i l_j \frac{\pi_j}{\rho}\right) - \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_k \nabla_k \left(\frac{\pi_j}{\rho}\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_i \nabla_i l_j\right) +$$

$$\left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_k \pi_j \nabla_k \left(\frac{1}{\rho}\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_i \nabla_i l_j\right) - \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_k \pi_j \nabla_k \left(\frac{1}{\rho}\left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_i \nabla_i l_j\right)\, d\Omega. \tag{4.68}$$

After expanding the terms with ∇_k using the product rule, the final result is

$$\int_D \frac{1}{\rho}\left[\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_k \nabla_k \left(\pi_j \nabla_i l_j\right)\left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_i - \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_k \nabla_k \left(\pi_j \nabla_i l_j\right)\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_i\right] d\Omega. \tag{4.69}$$

Next, we will show that the expression (4.69) is the same as (4.35a).

Introducing the representation of the velocity in Clebsch variables $\mathbf{u} = \boldsymbol{\nabla}\phi + \pi_j \nabla l_j$ and, subsequently, using the relations $\boldsymbol{\nabla} \times \mathbf{u} = \boldsymbol{\nabla} \times \boldsymbol{\nabla}\phi + \boldsymbol{\nabla} \times (\pi_j \nabla l_j) = \boldsymbol{\nabla} \times (\pi_j \nabla l_j)$, (4.36) can be transformed into

$$\begin{aligned}
&\int_D \frac{\boldsymbol{\nabla} \times \mathbf{u}}{\rho} \cdot \frac{\delta \mathcal{H}}{\delta \mathbf{u}} \times \frac{\delta \mathcal{F}}{\delta \mathbf{u}}\, d\Omega = \\
&\int_D \frac{1}{\rho}\Bigg(\left[\partial_y(\pi_j \boldsymbol{\nabla} l_j)_3 - \partial_z(\pi_j \boldsymbol{\nabla} l_j)_2\right]\left[-\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_2 \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_3 + \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_3 \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_2\right] \\
&+ \left[\partial_z(\pi_j \boldsymbol{\nabla} l_j)_1 - \partial_x(\pi_j \boldsymbol{\nabla} l_j)_1\right]\left[\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_1 \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_3 - \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_3 \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_1\right] + \\
&\left[\partial_x(\pi_j \boldsymbol{\nabla} l_j)_2 - \partial_y(\pi_j \boldsymbol{\nabla} l_j)_1\right]\left[-\left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_1 \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_2 + \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_2 \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_1\right]\Bigg)\, d\Omega.
\end{aligned} \tag{4.70}$$

Recombining, the various terms yields

$$\begin{aligned}\int_D \frac{1}{\rho}\Bigg(&- \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_i \nabla_k (\pi_j \boldsymbol{\nabla} l_j)_i \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_k + \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_1 \partial_x (\pi_j \boldsymbol{\nabla} l_j)_1 \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_1 \\ &+ \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_2 \partial_y (\pi_j \boldsymbol{\nabla} l_j)_2 \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_2 + \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_3 \partial_z (\pi_j \boldsymbol{\nabla} l_j)_3 \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_3 \\ &+ \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_k \nabla_k (\pi_j \nabla l_j)_i \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_i - \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_1 \partial_x (\pi_j \boldsymbol{\nabla} l_j)_1 \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_1 \\ &- \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_2 \partial_y (\pi_j \boldsymbol{\nabla} l_j)_2 \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_2 - \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_3 \partial_z (\pi_j \boldsymbol{\nabla} l_j)_3 \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_3 \Bigg)\, d\Omega,\end{aligned} \tag{4.71}$$

which after cancellation of terms with opposite sign gives the same result as (4.69)

$$\int_D \frac{1}{\rho}\left[- \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_i \nabla_k (\pi_j \boldsymbol{\nabla} l_j)_i \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_k + \left(\frac{\delta \mathcal{H}}{\delta \mathbf{u}}\right)_k \nabla_k (\pi_j \boldsymbol{\nabla} l_j)_i \left(\frac{\delta \mathcal{F}}{\delta \mathbf{u}}\right)_i\right] d\Omega. \tag{4.72}$$

Chapter 5

Conclusions and recommendations

Numerical models for wave propagation based on a variational or Hamiltonian formalism are in general preferable to models based on a purely PDE representation. These numerical models are coordinate independent and have a concise and exquisite structure. More importantly, they preserve not only the mathematical structure of the problem, but also the conservation laws associated with it and the phase space, in particular. Careful discretisation of a model constructed on the basis of a variational or Hamiltonian formalism may also preserve these nice properties at the discrete level. Conservative numerical schemes are in general also more robust and accurate, while in the case of variational or Hamiltonian based discretisations the scheme inherits all these properties automatically. Nevertheless, the downside of this approach is that preserving the variational or Hamiltonian structure (which should exist and be derived, in the first place) after numerical discretisation may not be an easy task.

In Chapter 2, a Hamiltonian based numerical model for linear (in)compressible, inviscid fluid flow in a rotating domain was derived. First, the corresponding Hamiltonian formalism for a compressible fluid was introduced and carefully discretised to preserve the underlying conservation properties. It is important to emphasise that the resulting DGFEM discretisation has been proven to have a Hamiltonian representation. Next, the incompressible limit was achieved with an application of Dirac's constrained theory directly on the discrete Hamiltonian formalism for compressible fluid flow. This is a novel way to account for the incompressibility constraint in a DGFEM discretisation. Finally, a Hamiltonian structure preserving symplectic space and time discretisation, with the modified midpoint time integrator derived in [7], ensures the accurate and robust discretisation of the various wave phenomena in compressible and incompressible fluids considered in this chapter. The simulation of inertial waves in a rotating rectangular parallelepiped was one of the important applications that was discussed. The latter enabled an alternative route for studying the most interesting property of inertial waves: the possibility of a chaotic wave attractor formation in geometries with a broken symmetry.

Based on the continuous Hamiltonian formalism for incompressible fluid flow in a rotating domain that was introduced in Chapter 2 several numerical schemes could be derived. In case of DGFEM discretisations a discrete Hamiltonian system can be obtained provided that the flux terms and boundary conditions are chosen properly. Depending on these choices, the resulting scheme may or may not coincide with the one presented in Chapter 2. It should be emphasised that special attention should be paid to the boundary conditions for a rotating domain, due to the mandatory requirement of satisfaction of the geotropic balance in rotating systems. The latter is of direct relevance to preserving the symmetry of the problem.

In Chapter 3, a new semi-analytical solution for three-dimensional inertial waves in a rotating rectangular parallelepiped was introduced, which was compared with a semi-analytical solution available in the literature. The advantages and disadvantages of both semi-analytical solutions were discussed in full detail. It is worth to emphasise that the numerical solution based on the algorithm discussed in Chapter 2 was more accurate than the two alternative semi-analytical solutions discussed in Chapter 3. The numerical solution in Chapter 2 was especially more accurate than the semi-analytical solutions in two aspects: accuracy and robustness. Next to the semi-analytical solutions, a discrete energy preserving FEM solution for inertial waves in a rotating rectangular parallelepiped was introduced to deal with the disadvantages of both semi-analytical solutions.

The semi-analytical techniques used in the Chapter 3, can be utilised for a derivation of semi-analytical solutions of inertial waves in other simple geometries. The suggested FEM solution, by and large, is applicable to more general domains. Thus, it will be useful for the computation of eigenvalues and their corresponding eigenvectors for inertial waves in more complex geometries.

In Chapter 4, a novel (non)linear Hamiltonian formulation was presented for inviscid, compressible fluid flow in a rotating domain with a free surface. First, the nonlinear Hamiltonian dynamics was conjectured and compared with the corresponding nonlinear compressible Euler equations with a free surface. Second, the conjectured nonlinear Hamiltonian formalism was derived via variation of a Lagrangian expressed in (extended) Clebsch variables and a subsequent reduction to Eulerian variables. In addition, the linearisation of the corresponding nonlinear Hamiltonian structure was derived, which appeared to be the free surface extension of the Hamiltonian formalism presented in Chapter 2.

These linear and nonlinear continuous Hamiltonian formulations can be used for the derivation of a numerical model for (in)compressible fluid flows with free surface dynamics. Here also, the main sensitive part of the scheme is the satisfaction of appropriate boundary conditions at free surface and the incompressibility constraint. The boundary conditions and appropriate DGFEM flux functions have to be chosen such that the resulting discrete system can still be represented in the Hamiltonian framework. The incompressible limit of the presented Hamiltonian structure can be derived via the Dirac theory of constraints, as it has proven to result in a robust, stable and accurate numerical technique for obtaining the incompressible limit on the Hamiltonian systems discussed in Chapter 2.

Appendix A

Design and implementation of a DGFEM application

In Chapter 2 we have introduced a new Hamiltonian DGFEM discretisation for the linearised Euler equations to compute inertial waves in a rotating domain. The implementation of this algorithm is a non-trivial task due to its complexity. In this Appendix we will give an overview of the design and implementation of this Hamiltonian DGFEM using the hpGEM toolkit, which was used and extended during this project. To a greater or lesser extent any implementation of a DGFEM discretisation requires three cornerstones in the heart of its design: a mesh, numerical integration and data structures, and output routines. These three basic modules are a must, and do not depend on the choice of the basis functions, flux functions or slope limiters that are used in the implementation. Hence, a library (framework) of predefined structures and methods that are common to many (discontinuous) finite element methods, is a desirable tool that can be used to ease and accelerate the implementation process of robust and efficient applications.

In this Appendix, the second version of hpGEM, a C++ software framework for DGFEM discretisations [80], will be introduced. As a particular example, an implementation of the application based on the Hamiltonian DGFEM discretisation discussed in Chapter 2 will be presented.

A.1 Introduction

The process of the transformation of a continuous model and respective governing equations to its discrete counterpart is called discretisation. This process is usually carried out prior to the implementation of a numerical model into a digital set of commands (application). This software application enables then computer simulations of the physical phenomena to be studied.

The discussion of the mathematical techniques and challenges to be met during the discretisation of the Hamiltonian formulation of the linear Euler equations for inertial waves is dealt with in the main part of this thesis, here we will focus on the implementation aspects. Simple and small problems, with their corresponding discretisations, usually are implemented in Matlab, Mathematica or any other general numerical mathematics framework. These packages provide not only a wide variety of tools for the implementation of specific numerical algorithms, but also frameworks for building your own problem specific applications. Unfortunately, those numerical environments are unsuitable for the implementation of complex problems that require more computational resources: in particular processing speed and memory. Additionally, complex problems result in an enormous number of commands (computer program), which are virtually impossible to structure, maintain and extend, due to the lack of an Object Oriented Programming (OOP) environment.

In contrast to conventional linear programming, where the program is presented as a list of commands and/or collection of commands (subroutines) to perform, an OOP is a set of interacting objects. An object is viewed as an encapsulator of some particular data with a set of functions designed to assist the usage of the data in a safe and responsible manner. This way small changes to the internal structure or methods of an object are extremely localised and do not require any additional changes in the rest of the program. Existing abstraction levels enable inheritance for objects (new object descriptors, classes can be created based on existing ones) and polymorphism (possibility of one common interface for many implementations, with mandatory common kernel (base) parts for different objects). Above all, OOP makes program code more readable, enables code reuse, facilitates future support and extendibility. Hence, flexible, robust and OOP-enabled environments are very desirable tools in the set of techniques useful for any numerical mathematician, especially in the development of software for problems with a high degree of complexity.

Nevertheless, those environments are not able to provide frameworks suitable for arbitrary numerical discretisations, and thus they are developed for specific discretisation methods, such as the finite difference method, i.e MathPDE [97], the finite volume method (FV), i.e. OpenFVM [48] and the finite element method (FEM), i.e., libMesh [52].

In this Appendix we will discuss the second version of the C++ software framework hpGEM for the implementation of discontinuous Galerkin finite element discretisations. The hpGEM framework provides the building blocks for developing (DG)FEM algorithms, enabling an easy, fast and robust application development. In Section A.2, we will give a brief introduction to the second version of the C++ software framework hpGEM. Next, in Section A.3, we consider a particular case study, viz. the computation of (in)compressible inertial waves discussed in Chapter 2. We will demonstrate the implementation procedure of the numerical application based on the algorithms and methods provided by hpGEM2.0. In Section A.4, the current state of hpGEM and future plans are discussed.

A.2 hpGEM: design and implementation

A.2.1 Philosophy of hpGEM

The hpGEM framework is a software package for the implementation of applications based on the discontinuous Galerkin finite element method. It was first released six years ago (hpGEM1.0) and the second release is expected this year (hpGEM2.0). Since the package is mainly concerned with the implementation of DGFEM discretisations, first a brief introduction on DGFEM is given below.

The starting point for a discontinuous Galerkin discretisation is a weak formulation of the PDEs. Next, the computational domain is tessellated into a large number of elements. In every element, the discrete equations are obtained via an approximation of the unknown variables using element-wise continuous basis functions. The solution is, however, allowed to be discontinuous across neighbouring element faces; there is only a weak coupling in between elements using numerical fluxes. The numerical fluxes are defined at the element faces in exactly the same way as in the finite volume method. So even though DGFEM is an extension of the finite element method, from an implementational point of view it has more similarities to a finite volume method. Actually, a zeroth-order finite volume method and zeroth-order DGFEM method are the same numerical method. For the element integration and other single element and face manipulation routines every element is mapped to a corresponding unit reference element. In this way a unified action can be performed on the same reference element and later be transformed to every physical element, via corresponding transformation matrices.

All DGFEM discretisations have one common entity: they all define a weak formulation of the corresponding PDEs and introduce numerical fluxes. Hence, in these terms the mathematical formulation of a DGFEM discretisation is independent of the specific problem, and more importantly, the basics of the implementation of such a problem is the same for all DGFEM discretisations. The latter motivated the introduction of the hpGEM library based on the following general ideas:

- *Framework provides the building blocks for the implementation of DGFEM methods upon which the users' application can be built.*

  The kernel of the framework is viewed as a library of data structures and corresponding methods, contrary to the view of a solver for a specific type of equations or physical processes. It is intended to provide key ingredients for numerical applications to help numerical scientists to concentrate more on application specific problems, rather than routine implementation of DGFEM basics.

- *Framework is dimension and element type independent.*

  The functionality and data structures provided by the framework is the same for all dimensions $d = 1, \dots, 4$. The latter is achieved by using special templated classes available in C++. Templates are a metaprogramming technique where special tags are used as an indication for the compiler to unwrap temporary source code for all types instantiated in the user source code, which is merged by the compiler with the rest of the user source code and then compiled.

  The framework provides a large variety of element types for all supported dimensions. To support this, the architecture of the software framework largely exploits inheritance and polymorphism concepts of C++.

- *Framework provides different levels of data and method access, depending on the experience of the user.*

  An experience driven application development is at the heart of this library. Depending on the proficiency in C++ and numerical methods different interfaces are suggested. The latter means that the framework offers fast and easy implementation for simple, straightforward applications in order to assist inexperienced users, but at the same time it enables a full control of the data and available methods for an experienced numerical scientist.

- *Framework includes a full self-test suite and extensive documentation.*

  The hpGEM framework is positioned as open-source software, is distributed freely, and encourages the external users to be part of the development team. This means that ideally the kernel of the library is not fixed and will be evolving in time. Therefore, an extensive unit test framework is a must. It is an essential validation tool for any change in the hpGEM kernel. Unit tests validate the correctness of individual classes, procedures and modules. In addition to the unit tests, the hpGEM testing framework contains test-applications, which validate collaborative tasks and features.

  Documentation is provided via precompiled Doxygen files, which are also available online, and some application-tutorials for the demonstration purposes.

- *Utilities wrapping the kernel are included in the framework.*

  Various utilities, such as multigrid, generation of various sets of basis functions, algorithms for *hp*-refinement, projection of continuous functions on FEM spaces, etc. are provided via extra layers of utilities around the kernel of hpGEM.

- *Framework does not require any external library.*

  The hpGEM framework is a self-sufficient library. The library installs as a one entity and does not require any external libraries. However, it is worth to mention that the framework embed some linear algebra functionality borrowed from open-source BLAS libraries. User applications may include their own external libraries, such as boost, PETSc and/or SLEPc, etc.

- *Framework supports various input and output environments.*

  The mesh of the tessellated domain can be either imported or, in the case of a simple geometry, be generated via internal utilities. The main external mesh format is the Centaur mesh generator. However, additional gMesh format support is also planned to be implemented.

  Tecplot data format is our main data-output format. Despite the fact that the open-source Paraview data visualisation tool has support for commercial Tecplot data files, we plan to add conversion utilities to various file formats (Matlab, Mathematica, native Paraview and etc.)

A.2.2 Design Considerations

The previous sections of this Appendix have already given a motivation for the specific choice of the C++ general-purpose programming language as the main programming environment for hpGEM. The latter, of course, does not exclude a possibility of linking to libraries written in other languages. The kernel of the hpGEM framework uses, however, mainly C++, in combination with the STL library, and includes in a cross-compilation a Fortran library from BLAS for basic linear algebra operations.

The general advantages of OOP are reusability, reliability, robustness, extensibility and maintainability. All of these properties are essential in the development of software for highly complex applications. Especially for numerical methods OOP has the additional advantage that mathematical entities can be directly related to individual classes. The latter allows numerical scientists, who are used to manipulate mathematical formulations rather than objects, subroutines or other sequences of program instructions, to be more comfortable with the development of applications.

The general structure of the hpGEM framework is described using the Unified Modelling Language (UML). The Unified Modelling Language is a general-purpose modelling language often used in the design stage of a complex software development process. It includes a set of graphic notation techniques to create visual models of object-oriented software architecture. Later in this chapter UML class and collaboration diagrams will be used to demonstrate classes, attributes and operations.

The hpGEM framework consists of five kernel forming namespaces[1] **Base**, **Integration**, **Geometry**, **Linear Algebra**, **Output** and a subjunctive **Utilities** namespace. These namespaces are discussed in turn next.

Namespace Geometry

All geometrical concepts (reference and physical geometries, corresponding transformations, etc.) are assembled in the **Geometry** namespace. The hpGEM framework supports various mesh-types, including mixed meshes. In Table A.1 possible geometric shapes are listed. Without exception, every specific reference geometry is derived from an abstract *ReferenceGeometry* class, which is templated with a dimension (*DIM*). Templates enable a single base class description for every dimension ($d = 1, \dots, 4$ is implemented).

---

[1] Namespaces allow to localise the visibility of various entities such as classes, objects and functions under a name. This way the global scope of the programm can be divided in 'sub-scopes', each one with its own name.

A detailed inheritance map diagram for reference geometries is given in Figure A.1. One can notice that all reference geometries are regular, unit geometrical shapes, in contrast with physical geometries (see Table A.1), which can be of a general kind.

Polymorphism, which is supported by C++, enables the possibility to operate with any of the derived objects (reference geometries) through one generic pointer to the base class *ReferenceGeometry*, which defines a generic interface. Therefore, mesh construction does not require a separate implementation for every shape type, but rather needs one generic implementation, assuming manipulations with a pointer to the base class.

As already mentioned polymorphism is used to define the variety of physical shapes with *PhysicalGeometry* as a base class, which is templated with a dimension. Moreover, every derived instance of *PhysicalGeometry* incapsulates a pointer to *ReferenceGeometry*. Meanwhile, *ReferenceGeometry* inherits additional functionality: refinement and mapping of codimensions of a given shape through *RefinementGeometry* and *MappingCodimensions*, respectively. It is very important to note, that only one particular reference geometry object is created per shape-type, which is accomplished via the singleton design pattern [36]. A summary of the geometry oriented part of the kernel is given in the UML class map in Figure A.2.

Dimension	Shape	Number of Nodes	Number of Faces
1	Line	2	2
2	Triangle	3	3
	Quadrilateral	4	4
3	Pyramid	5	5
	Tetrahedron	4	4
	Hexahedron	8	6
	Triangular Prism	6	5
4	Octachoron	16	24

Table A.1: Geometrical shapes supported in hpGEM for each dimension.

Namespace Base

The namespace **Base** provides safe interfaces to the data structures and corresponding methods of hpGEM.

In particular, the user is provided with two main classes as entry points to the kernel, *HpgemUI* and *HpgemUISimplified*. The difference, made ob-

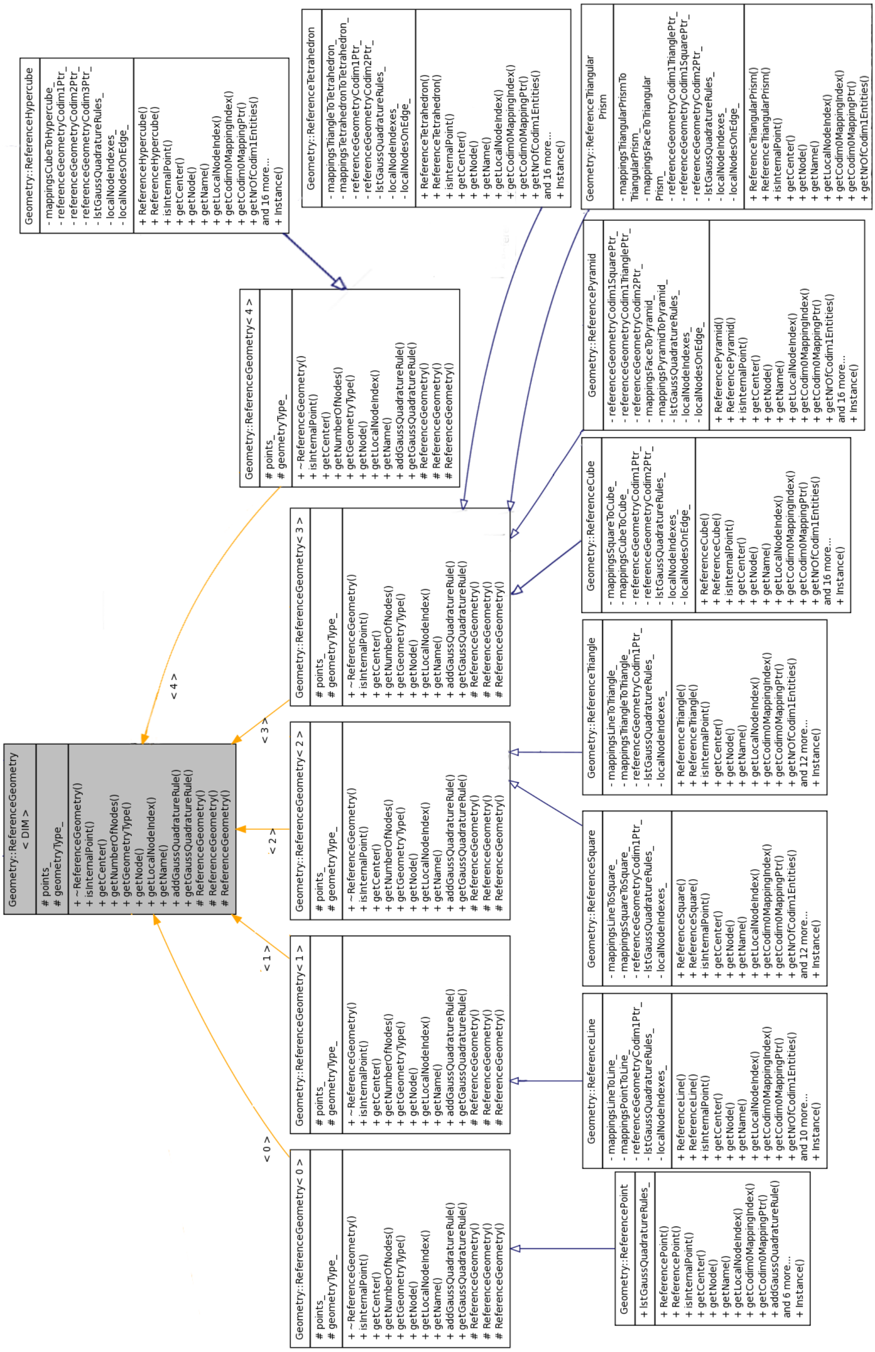


Figure A.1: Doxygen compiled UML diagram of the *ReferenceGeometry* class and its 'children'.

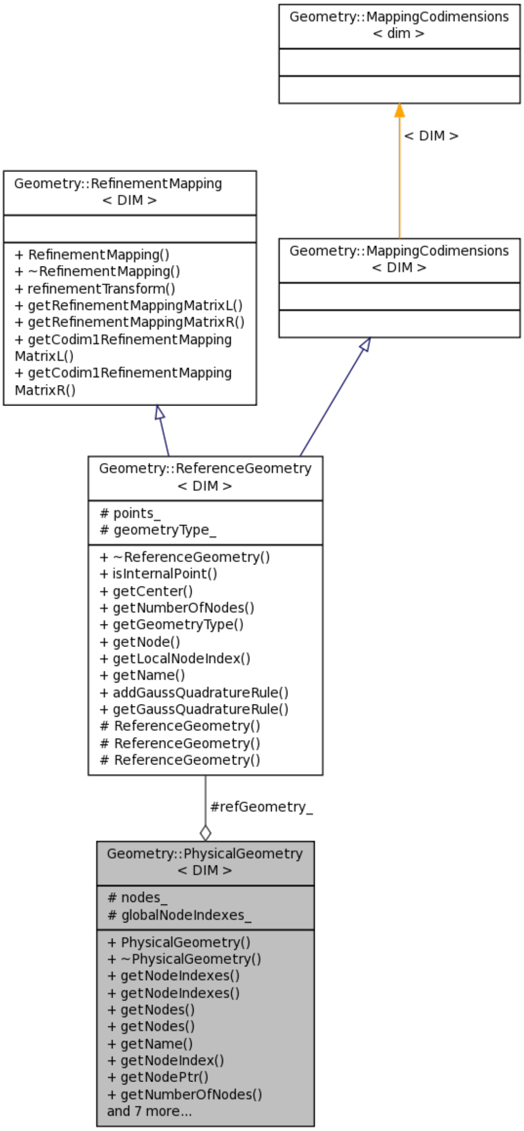


Figure A.2: UML collaboration diagram of the *PhysicalGeometry* class.

vious in the naming convention, is revealed in the level of access to the kernel and required default behaviour. In case of *HpgemUISimplified*, the user is choosing a fully automatic regime, where most of the required manipulations are defined through the default behaviour of the package, e.g definition of basis functions, choice of output routines, definitions of (Gauss) quadrature, pre-existing face and element integrands to be filled with corresponding expressions, etc.

Simple DGFEM discretisations, which do not require any special treatment, are advised to be implemented through the *HpgemUISimplified* interface. The user inherits from *HpgemUISimplified* the following information

- face and element integrand virtual functions, to be implemented via appropriate numerical flux and element integrand expressions;
- a default set of basis functions, the order of which is chosen through the configuration variables and, by default, is the same for every element;
- a default set of Gauss integration rules, the order of which is chosen automatically depending on the order of the basis functions for the current element;
- a virtual function to be implemented for setting initial conditions and reading configuration variables;
- a virtual function for importing or generating a mesh and initialising corresponding mesh-manipulation tools (space-time converter, mesh mover, mesh refiner, etc.);
- a predefined *solve* method, which initialises the data structures, checks if the initialisation step is complete and valid, performs a time step, which includes an iteration over all elements and faces, with subsequent element and face integration and output in the corresponding data structure, associated with every element and face.

This simplified interface assists non-experienced users in implementing their DGFEM methods. Hiding most of the implementation details in the automated code under the interfaces, the framework helps the numerical scientist to concentrate only on the specific part of his implementation, leaving cumbersome routine implementation details to hpGEM.

In case of a DGFEM implementation for more complex problems, which require direct access and nontrivial manipulations with the kernel of the package, the suggested entry class is *HpgemUI*. In this case, the user has a safe direct access to all data and methods. To enable support of multigrid and/or a cascade of meshes, *HpgemUI* encapsulates a collection of meshes in a *LevelTree* hierarchical tree-type data structure, where a single mesh is presented via the *MeshManipulator* class (see Figure A.3).

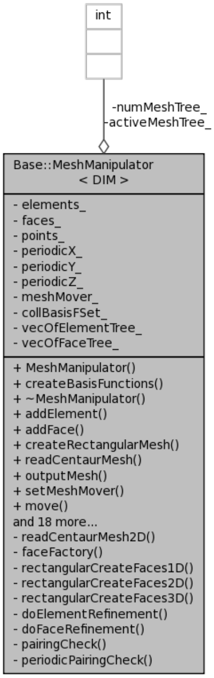


Figure A.3: *MeshManipulator* UML class diagram.

The *MeshManipulator* class presents an object for a mesh representation in hpGEM. It incapsulates a collection of elements (*Element*), faces (*Face*) and nodes (*Node*), and provides methods for configuring, initialising, generating and destroying them.

Every *Element* class, templated with a dimension, inherits from a templated *ElementGeometry* class, which provides the geometrical details of the element, and a templated *ElementData* class, which enables element relevant data storage, including the possibility of archiving data from several time steps. See Figure A.4 for details. It is worth mentioning that the geometry related part of the element belongs to the namespace **Geometry**.

From the class map given in Figure A.4 it is apparent that every element has: (i) its unique identifier; (ii) its unique geometry, with corresponding reference geometry, and its unique mappings and refinement techniques; and (iii) a set of basis functions and quadrature rules.

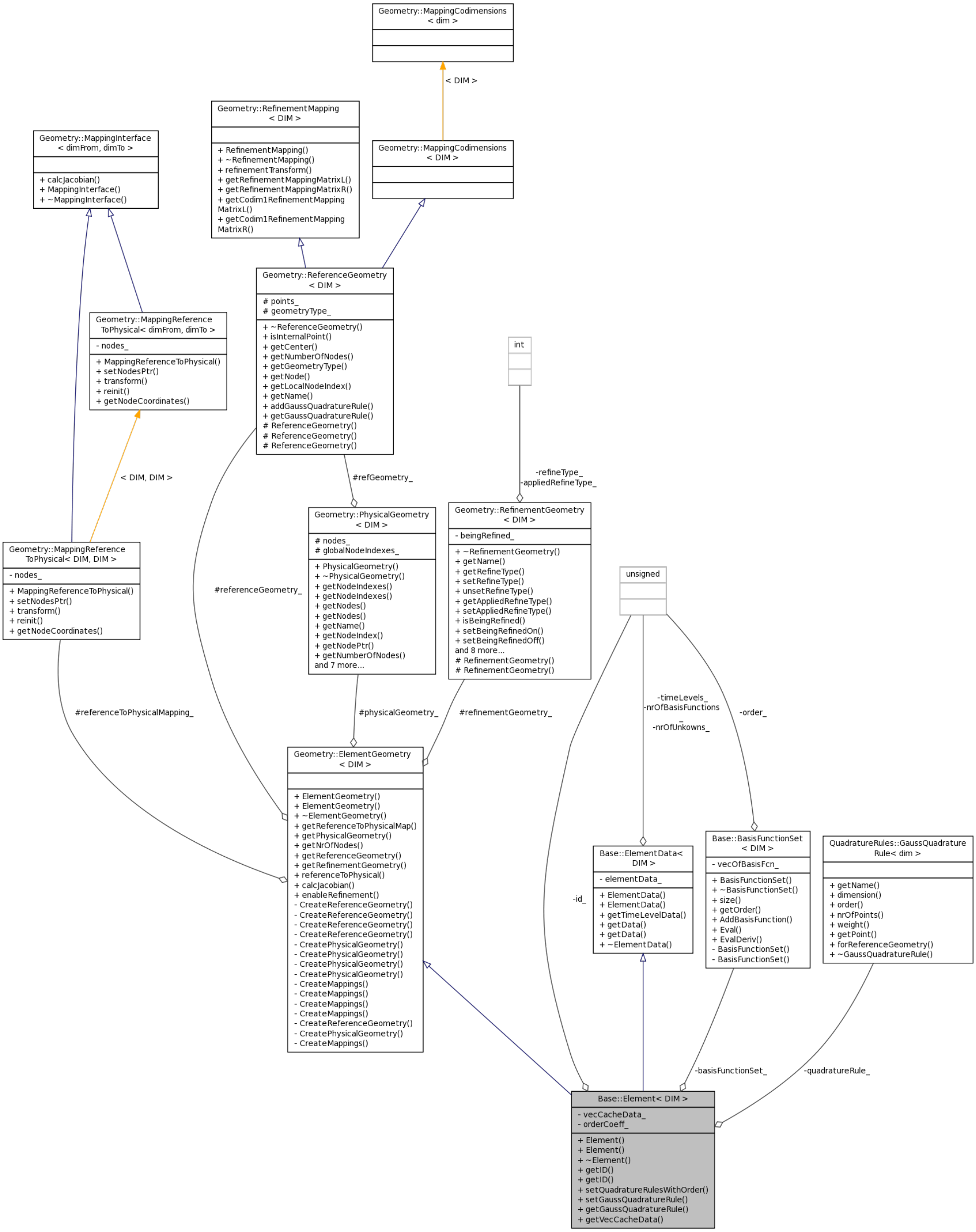


Figure A.4: *Element* UML collaboration diagram.

The same division into geometrical and data related entities is performed for the *Face* class, for details see Figure A.5. As one can see from the class diagram, *Face* is described through only two pointers to the left and right neighbouring elements. Note, left and right here are not used as an indicator for the direction, but as a simple identifier. Additionally, *Face* incapsulates a data container for storage of a pre-computed cache: such as normals, Jacobians, etc., to speed-up the calculations in case of time-static meshes.

Namespace Integration

The namespace **Integration** aggregates every aspect of element and face integrations. Integration routines are an essential part of any (DG)FEM package, due to their presence in each and every implementation of a weak formulation. The architectural design of integration routines is very challenging. On the one hand, the implementation has to be as robust and efficient as possible, since it is executed in nearly every manipulation with FEM based algorithms. On the other hand, the variety of places, ways and objects to be integrated force the implementation to be as generic as possible. Additionally, the user should have an ability to use default integration rules (regenerated for each element type), as well as to add his/her own integration rule implementations.

All numerical integration is handled through the utilisation of a default set of Gauss quadrature rules, which use a weighted sum of the integrand at the quadrature points for the given reference geometry. The hpGEM framework supports up-to seventh order Gauss quadrature rules for every available shape. Of course, the user is free to add new and higher order integration rules. It is also worth mentioning that the transformation between reference and physical spaces takes place automatically via the Jacobian of the corresponding mapping.

To integrate any object (mathematical function, variable, constant, etc.) one has to create an *ElementIntegral* or *FaceIntegral* object, call the corresponding function *integrate* and provide the required arguments, as follows:

```
MyElementIntegrandType<DIM>                 myElIntegrand;
for (ListOfElementsT::iterator it=elements.begin();
        it != elements.end(); ++it)
{
    Element<dim>& element = *it;
    /// Integrate using a quadrature rule of the element
    elIntegral.integrate(element, myElIntegrand, result);
}
```

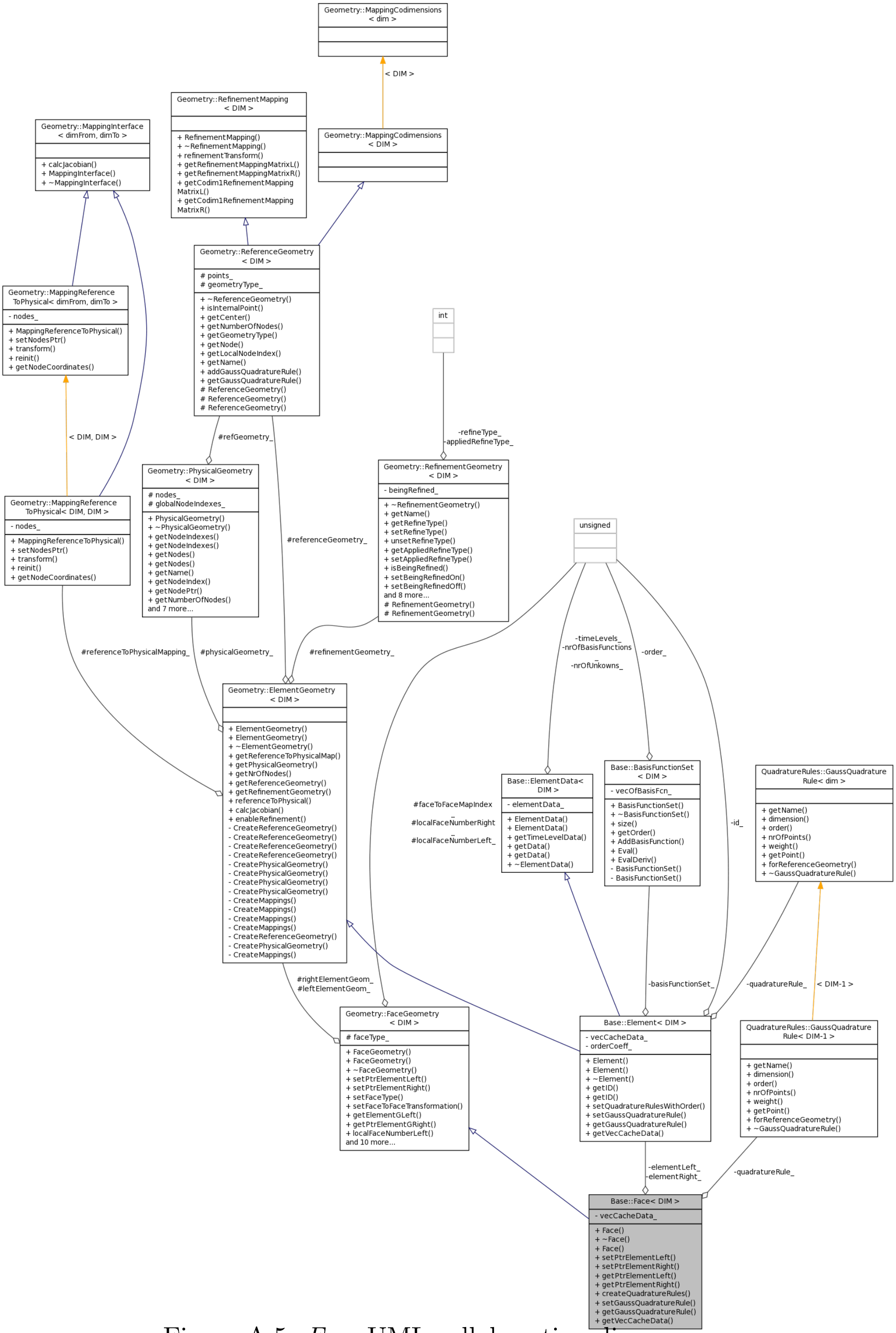


Figure A.5: *Face* UML collaboration diagram.

The code sample provided above, iterates through the collection of dimension dependent *Element*s and integrates the integrand provided via the variable *myElIntegrand* over each element. The second argument of the *integrate* function can be

- a class or struct with an appropriate operator definition;

```
template <unsigned int DIM>
class MyElementIntegrand :
       public Integration::ElementIntegrandBase<DIM>
{
public:
    void operator()(const Base::Element<DIM>& elem,
                    const Geometry::PointReference<DIM>& p,
                    double& ret)
    {
         /// add your integrand here
         ret = p[0]*p[1]*p[2]; // f = x*y*z;
    }
};
```

- a global function with appropriate arguments;

```
template <unsigned int DIM>
void MyElementIntegrand()(const Base::Element<DIM>& elem,
                          const Geometry::PointReference<DIM>& p,
                          LinearAlgebra::NumericalVector& ret)
        {
           /// add your integrand here
           ret[0] = p[0]*p[1]*p[2]; // f = x*y*z;
        }
```

- a class method with appropriate arguments (in this case, the method *integrate* needs an additional pointer to the object owner of the integrand-function);

```
template <unsigned int DIM>
class SimpleDemoProblem : public HpgemUISimplified<DIM>
{

public:
    void elementIntegrand(const Base::Element<DIM>& elem,
                          const Geometry::PointReference<DIM>& p,
                          LinearAlgebra::Matrix& ret)
    {
     /// add your integrand here
           ret[0][0] = p[0]*p[1]*p[2]; // f = x*y*z;
    }
}
```

The last argument of the function, which implements the integrand expression, is in all three cases different. The type of this last argument in the integrand-function automatically sets the type of the returning value in the method *integrate*. The latter means that one can control the data type of the return object through the definition of the integrand-function. For example, in the first case, the type of the return value *result* is a *double*, in the second case a *NumericalVector* and in the last case a *Matrix*.

Namespace Linear Algebra

The namespace **Linear Algebra** provides efficient and robust linear algebra functionality. Numerous tests showed that the current implementation based on *valarray*[2] is the fastest, most efficient and robust implementation of the required functionality. Even though the C++ implementation is cross-compiled with the BLAS library that is written in Fortran, for some additional functionality, it does not require the external library installation. The two main representatives of this namespace are *Numerical Vector* and *Matrix*. A detailed description of the functionality can be found in the Doxygen documentation available online. A summary is given in Figure A.6

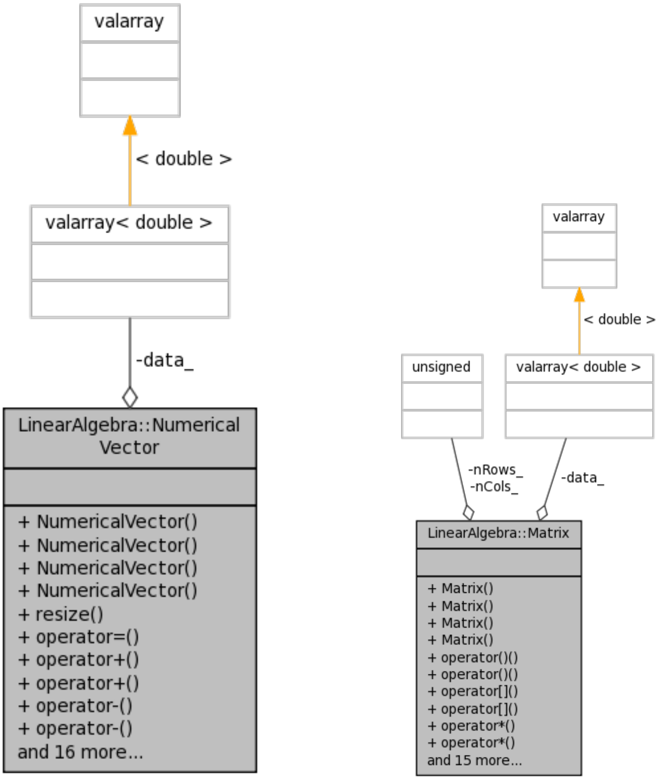
Namespace Output

The hpGEM framework offers not only an easy application development and robust computations, but also a graphical representation of the processed data. The default output format is Tecplot (Tecplot, Inc.). Even though most of the commercially-available Tecplot software files can be opened by some open-source plotting environments, i.e. Paraview, it is planned to provide additional scripts for the translation of the Tecplot output data file into Matlab and Mathematica data format, which are more widely used in the scientific computing community.

In addition, the output can be written on a per-element basis, meaning that the discontinuous origin of the solution is preserved, which is particularly useful for code debugging, or as nodally averaged data with forced continuity. Also, it is worth to mention that all plots in Chapter 2 were prepared with these output facilities.

---

[2] Next generation of constant size arrays available in STL.



(a) Numerical Vector (b) Numerical Matrix

Figure A.6: Collaboration diagrams of the two main classes in **Linear Algebra**

Namespace Utilities

The namespace **Utilities** assembles all the features that are not considered essential, but extraneous to have in the DGFEM software package. Those features either provide additional functionality or the implementation of predefined algorithms and methods, such as the calculation of various norms and normal vectors; multigrid support, with several levels of meshes; *hp*-refinement; a default pre-generated set of basis functions up-to fourth order; a default pre-generated set of Gauss quadrature rules for every supported geometry at least up-to seventh order; the ability to generate a global algebraic problem (global assembly) based on the local representation on every element; etc. For a full list of features and the instructions, we refer to the documentation of hpGEM package, available online.

A.3 Inertial waves application in hpGEM

In this section an implementation of the application for (in)compressible inertial waves presented in Chapter 2 will be discussed. This description can also be viewed by future users as a tutorial for implementing DGFEM discretisations in hpGEM.

A.3.1 Configuration file

To make sure that the user is able to run his/her application with different parameters without the need to recompile the code, hpGEM introduces the possibility of configuration variables. Default configuration variables are defined in the class *ConfigurationData*, and can be extended by the introduction of a new, user-defined class for configuration variables inherited from *ConfigurationData*. The initialisation of these variables is done through the configuration file and passed to the executable.

To control the safety of the data format of the configuration file, a special HTML based configuration file generator was created. This wizard-like builder walks the user through the sheets of configuration variables, grouped by similarity, and asks the user to set the appropriate values, while in parallel performing some sanity checks.

The configuration variables for the inertial waves application are presented below:

- *mode of the application*

  First, the user has to choose between the various modes available, see Figure A.7. The possibilities are (i) compressible fluid flow in a

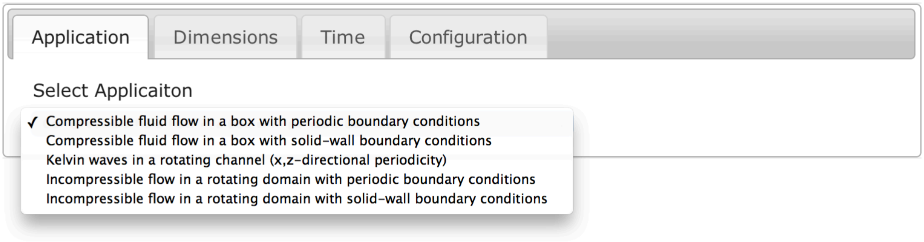


Figure A.7: The application type is chosen from drop down box.

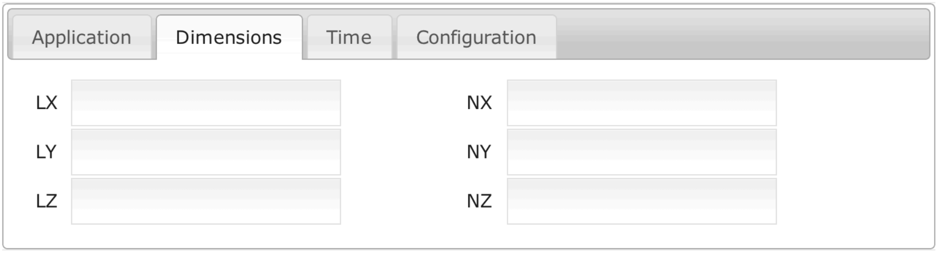


Figure A.8: Dimensions of the domain in all three directions, with corresponding number of elements in the respective directions.

box with periodic boundary conditions, (ii) compressible fluid flow in a box with solid-wall boundary conditions, (iii) Kelvin waves in a rotating channel (x,z-directional periodicity), (iv) incompressible flow in a rotating domain with periodic boundary conditions, and (v) incompressible flow in a rotating domain with solid-wall boundary conditions (‘classic’ inertial waves). After making a choice, the configuration file is automatically renamed to the appropriate filename.

- *dimensions of the domain and tessellation parameters*

  Second, the user has to specify the dimensions of the problem by setting the length of the domain in each direction, and to give the number of elements for each direction in the tessellation (see Figure A.8).

- *time related values*

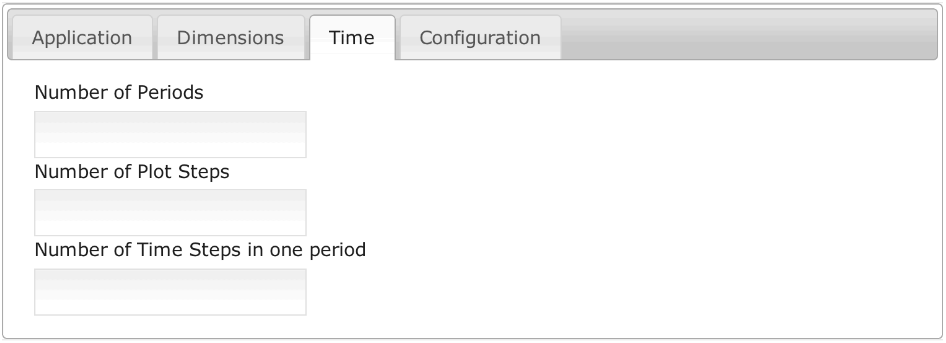


Figure A.9: Configuration variables related to time.

Figure A.10: Time related configuration variables used in the discretisation.

Third, the end-time, the frequency of plots in terms of the time steps, and the number of time steps in each period have to be set (see Figure A.9).

- *discretisation parameters*

  The final input the user has to specify is the number of degrees of freedom in each element and the corresponding value of the θ parameter, which is used in the definition of the flux functions in the corresponding DGFEM discretisation, see Figure A.10. After these values are set, the user has to push the **Generate** button and save the created configuration file to some location on disc.

The generated file is passed to the executable at run-time, which is being parsed by the virtual method *readFile()* implemented by the user. The latter manipulations results in a fully initialised inertial waves application.

A.3.2 Mesh generation

In Chapter 2, the inertial waves model was mainly discussed in cuboid-type domains in three-dimensional space. The domain may, however, also have a constant background rotation. The latter indicates that the implementation concerns a discretisation in three-dimensional space and in a possibly rotating cuboid with solid walls and/or periodic boundaries.

Fortunately, hpGEM provides sufficient functionality: it automatically generates a rectangular (triangular) mesh for a simple cube, provided the dimensions of the cube and the number of elements in each coordinate direction are known. Note, the generation of a mesh for other geometries can either be implemented in hpGEM as a new utility, or the required mesh can be imported through the supported interfaces, assuming the mesh is generated by an external package.

The class *HEuler* is inherited from the three-dimensional specialised instantiation of the *HpgemUI* providing an access point into the hpGEM framework. Due to the non-trivial DGFEM discretisation, the *HpgemUI* interface was chosen as a parent for the inertial waves implementation class, instead of the simplified ‘semi-automatic’ *HpgemUISimplified*.

```
unsigned int DIM = 3; /// setting dimension of the problem to 3D
class HEuler : public Base<DIM>
{
public:
     /// initialising the mesh by giving the range of the domain and
    /// the number of elements in each direction.
    void initialiseMesh()
    {
        RectangularMeshDescriptor<DIM> rectangularMesh;

        rectangularMesh.bottomLeft_[0] = 0; // x coordinate
        rectangularMesh.bottomLeft_[1] = 0;//  y coordinate
        rectangularMesh.bottomLeft_[2] = 0;//  z coordinate
        rectangularMesh.topLeft_[0] = 1;       //  x coordinate
        rectangularMesh.topLeft_[1] = 1;       //  y coordinate
        rectangularMesh.topLeft_[2] = 1;       //  z coordinate
        rectangularMesh.numElementsInDIM_[0] = 8; // in x-direction
        rectangularMesh.numElementsInDIM_[1] = 8; // in y-direction
        rectangularMesh.numElementsInDIM_[2] = 8; // in z-direction

        addMesh(rectangularMesh, RECTANGULAR);
        return;
    }
};
```

The code fragment shows a mesh generation routine in *HEuler*, where a simple rectangular mesh is created based on input parameters provided via the mesh descriptor *RectangularMeshDescriptor*. A cuboid is represented with two diagonal points, e.g. bottom-left and top right. Additionally, the *RectangularMeshDescriptor* includes variables with a number of points in each direction. After initialisation of a rectangular domain with size $[0, 1]^3$, and the assignment of the tessellation variables to 8, viz. the number of cubes in each direction, the descriptor is added to the mesh collection. Note, the inertial waves application uses only one mesh per application, and the specific numbers in the assignments are only for presentation purposes, in a real application they are read from a configuration file.

A.3.3 Initial projection

In all possible modes of the inertial waves application, the user is initialising the program with the solution at initial time. Usually, the initialisation of the unknowns involves a projection of continuous (discrete) functions on the chosen FEM spaces. In general, let $\mathbf{p}(\mathbf{x}, t)$ be a vector of unknowns and $\mathbf{p}(\mathbf{x}, 0) = \mathbf{q}(\mathbf{x})$. In each element of the tessellation, the following algebraic system is formed

$$M\mathbf{p} = \mathbf{b}, \tag{A.1}$$

where $M_{ij} = \int_K \phi_i \phi_j \; dk$ is a local mass matrix in the current element K and the right hand side vector is defined through $\mathbf{b}_j = \int_K \mathbf{q}\phi_j \; dk$, with $i, j = 1, \ldots, dof$. Subsequently, the unknown variables $\mathbf{p}$ can be computed by solving the algebraic equation (A.1). For definitions of the FEM spaces and corresponding basis functions, we refer to Chapter 2.

In case of a stationary mesh, the inverse of the local mass matrix can be also stored in the element, to speed-up future calls.

A.3.4 Integrands

As was already discussed in the previous sections, to integrate a function over a face or element one needs to provide an integrand function to the corresponding integrator. The integrand function defines the integral expression. In the current application various integrands has been defined. First of all, to prepare for an application of Dirac's constraint theory to the compressible framework, one has to create the global algebraic system.

$$\dot{\mathbf{U}}_j = - M^{-1}_{jk} R_l \mathbf{DIV}_{kl} - 2\boldsymbol{\Omega} \times \mathbf{U}_j \tag{A.2a}$$

$$M_{kl}\dot{R}_l = \mathbf{U}_j \cdot \mathbf{DIV}_{jk} \tag{A.2b}$$

Recalling the definitions from Chapter 2, M is the global mass matrix (contains element integrals only), $\mathbf{DIV} = \mathbf{E} - \mathbf{G}$ is the global representation of a discrete divergence operator (contains element integrals in $\mathbf{E}$ and face integrals in $\mathbf{G}$), and constant rotational terms (containing element integrals only). The latter suggests that the following integrands have to be implemented

- *Mass matrix*

```
void calculateMassMatrix(const ElementT& element,
                         const PointReferenceT& p,
                         LinearAlgebra::Matrix& massMatrix)
    {
      .
      .
      .
    }
```

- *Gradient of a basis function*

```
 void elementIntegrand(const ElementT& element,
                       const PointReferenceT& p,
                       VectorOfMatrices& returnObject)
    {
        .
        .
        .
     }
```

- *Flux integration over a face*

```
void faceIntegrand(const Face& face,
                   const PointPhysicalT& normal,
                   const PointReferenceT& p,
                   VectorOfMatrices& ret)
    {
        .
        .
        .
    }
```

Due to the locality of the DGFEM discretisation, all global matrices are notably sparse. Therefore, the PETSc sparse data format [10] has been used to take advantage of the reduced memory storage.

A.3.5 Dirac's algorithm

Next, the compressible system has to be taken to its incompressible limit. Application of the Dirac algorithm from Chapter 2 results in the following algebraic system

$$\dot{\mathbf{U}}_j = -2\boldsymbol{\Omega} \times \mathbf{U}_j - \lambda_l M_{jk}^{-1} \mathbf{DIV}_{kl} \tag{A.3a}$$

$$\lambda_l \mathbf{DIV}_{jk} M_{jm}^{-1} \cdot \mathbf{DIV}_{ml} = -\mathbf{DIV}_{jk} \cdot 2\boldsymbol{\Omega} \times \mathbf{U}_j. \tag{A.3b}$$

The resulting two algebraic systems of equations can be merged into one global system, with corresponding matrices

$$\mathbf{P}\dot{\mathbf{X}} = \mathbf{Q}\mathbf{X}, \tag{A.4}$$

where $\mathbf{X} = (\mathbf{U}, \Lambda)^T$. Obviously, $\mathbf{P}$ and $\mathbf{Q}$ are not time dependent, in contrast to $\mathbf{X}$. The formation of those matrices is done with the *createIncompressibleSystem()* function. The required sparse matrix operators are imported from the corresponding PETSc library.

A.3.6 Time integration

The appropriate time integration applied to (A.4) is implemented in the function *solve()*. The time integrator for discrete Hamiltonian systems needs to be symplectic. The current version implements a modified midpoint time integrator.

A.3.7 Output

For output the implementation uses the default Tecplot routines, available in hpGEM. All plots presented in the Chapter 2 were produced via the *output()* function of the *HEuler* class. Note also that the number of time-steps after which the solution data is uploaded to the external file is defined through the corresponding configuration variable.

A.4 Summary and future of hpGEM

In this Appendix, a brief overview of the hpGEM C++ software package was presented. First, the philosophy of the library was introduced combined with the basic requirements for a general open-source DGFEM package. Next, the architectural solutions were discussed in detail.

An example of the implementation of an application to compute inertial waves in an (in)compressible fluid was presented. Several important stages of the application have been pinpointed and difficulties have been discussed.

Future work will include the extension to parallel simulations and runtime live-output regimes, but more importantly the hpGEM framework is seen to transform into a wizard-driven DGFEM application development environment, where the code of an application is generated from the inserted or chosen properties provided by the wizard sheets.

Summary

The main focus of this thesis is to develop numerical discretisations for both compressible and incompressible inviscid flows that also preserve conservation laws at the discrete level. Two alternative approaches are discussed in detail: a semi-analytical solution; and, a fully numerical discretisation.

The semi-analytical solution is derived for the case of incompressible inertial gyroscopic waves in a three-dimensional rotating rectangular parallelepiped. By performing a detailed numerical comparison it is shown that the semi-analytical solution vastly improves on the state-of-the-art solution previously available in the literature. Despite the improved accuracy of this new semi-analytical solution, further comprehensive investigations revealed a small weakness near the boundaries of the domain. A novel finite element method is derived that compensates for the known weaknesses in both the original and new semi-analytical approaches. This numerical approach allowed further investigation of inertial waves and associated physical phenomena.

Subsequently, the thesis investigates the development of conservation law preserving Hamiltonian discretisations modelling (in)compressible fluid flow in three-dimensional domains with various boundary conditions. The continuous Hamiltonian description of the physical phenomenon is discretised via a discontinuous Galerkin method, which allows the construction of highly stable, conservative, energy preserving numerical discretisations for both the compressible and incompressible cases. This numerical scheme preserves the Hamiltonian mathematical structure even at the discrete level, which facilitates highly accurate and robust simulations of (in)compressible fluid flows. For the particular case of inertial gyroscopic waves this numerical scheme is more robust and accurate than the corresponding aforementioned semi-analytical solutions.

Finally, a new version (version 2) of the in-house open-source C++ software that enables fast and easy implementation of discontinuous Galerkin discretisations (hpGEM) is introduced. The discussion evolves around the philosophy, design principles and aim of the package. Additionally, a set of new features and guidelines how to use the package are highlighted via a series of illustrative, small, step-by-step examples.

Samenvatting

Het centrale thema van dit proefschrift is de ontwikkeling van numerieke methoden voor zowel samendrukbare als niet-samendrukbare wrijvingsloze stromingen die ook aan de behoudswetten voldoen na numerieke discretizatie. Twee alternatieve methoden worden in detail besproken: semi-analytische methoden en een volledige numerieke aanpak.

De semi-analytische oplossing is berekend voor inertiale gyroscopische golven in een driedimensionaal roterend rechthoekig parallellepipedum. Via een gedetailleerde vergelijking van de resultaten wordt aangetoond dat de oplossing aanzienlijk nauwkeuriger is dan de tot nu toe in de literatuur beschikbare oplossing. Na een uitgebreide vergelijking van de resultaten laat de nieuwe semi-analytische methode echter ook zien dat de oplossing aan de rand van het domein niet optimaal is. Om dit probleem op te lossen is er een nieuwe eindige elementenmethode ontwikkeld die de onnauwkeurigheden in de oorspronkelijke en nieuwe semi-analytische methoden reduceert. Deze nieuwe numerieke benadering maakt het mogelijk om traagheidsgolven en gerelateerde fysische verschijnselen te bestuderen.

Het tweede onderwerp in dit proefschrift is de ontwikkeling van Hamiltoniaanse numerieke technieken voor niet-visceuze (on)samendrukbare stromingen in driedimensionale domeinen met verschillende typen randcondities behoudt. De oorspronkelijke Hamiltoniaanse formulering van de fysische verschijnselen wordt gediscretiseerd met een discontinue Galerkin methode, hetgeen resulteert in een stabiele, conservatieve en energie behoudende numerieke discretizatie voor zowel samendrukbare als niet-samendrukbare stromingen. Deze numerieke discretizatie behoudt de Hamiltoniaanse structuur, hetgeen resulteert in zeer nauwkeurige en robuuste numerieke simulaties van (on)samendrukbare stromingen. Voor het specifieke geval van inertiale gyroscopische golven is de numerieke methode zelfs nauwkeuriger en robuuster dan de eerder genoemde semi-analytische oplossingen.

Als laatste onderwerp wordt een nieuwe versie (versie 2) van de hpGEM C++ software voor het snel en makkelijk implementeren discontinue Galerkin discretizaties besproken. Deze software is beschikbaar voor gebruik door derden. Speciale aandacht wordt gegeven aan de ontwerp filosofie, ontwerp principes en het doel van het software pakket. Daarnaast worden een aantal nieuwe mogelijkheden besproken en richtlijnen gegeven hoe het software pakket gebruikt kan worden. Deze voorbeelden worden stap voor stap uitgewerkt.

Bibliography

[1] H. D. I. Abarbanel, R. Brown, and Y. M. Yang. Hamiltonian formulation of inviscid flows with free boundaries. *Physics of Fluids*, 31:2802–2809, October 1988.

[2] K. D. Aldridge and L. I. Lumb. Inertial waves identified in the Earth's fluid outer core. *Nature*, 325(1-2):421–423, 1987.

[3] K. D. Aldridge, L. I. Lumb, and G. A. Henderson. A Poincaré model for the Earth's fluid core. *Geophysical and Astrophysical Fluid Dynamics*, 48:5–23, October 1989.

[4] K. D. Aldridge and A. Toomre. Axisymmetric oscillations of a fluid in a rotating spherical container. *J. Fluid Mech.*, 37:307–323, 1969.

[5] V. R. Ambati. Forecasting Water Waves and Currents: A Space-time Approach. *PhD thesis, University of Twente*, 2008.

[6] V. R. Ambati and O. Bokhove. Space-time discontinuous Galerkin discretization of rotating shallow water equations. *Journal of Computational Physics*, 225(2):1233–1261, 2007.

[7] V. R. Ambati, S. Nurijanyan, E. Gagarina, and O. Bokhove. Sympletic time discontinuous Galerkin discretizations for nonlinear harmonic oscillator. *In preparation, 2013.*

[8] I. Anderson, A. Huyer, and R. L. Smith. Near-inertial motions off the Oregon Coast. *Journal of Geophysical Research*, pages 5960–5972, 1983.

[9] V. I. Arnol'd. The Hamiltonian nature of the Euler equations in the dynamics of a rigid body and of an ideal fluid. *Uspekhi Mat. Nauk*, 24(3(147)):225–226, 1969.

[10] S. Balay, J. Brown, K. Buschelman, V. Eijkhout, W. Gropp, D. Kaushik, M. Knepley, L. Curfman McInnes, B. Smith, and H. Zhang. PETSc Users Manual.

[11] H. Bateman. Notes on a Differential Equation Which Occurs in the Two-Dimensional Motion of a Compressible Fluid and the Associated Variational Problems. *Royal Society of London Proceedings Series A*, 125:598–618, November 1929.

[12] R. C. Beardsley. An experimental study of inertial waves in a closed cone. *Stud. Appl. Math.*, 49:187–196, 1970.

[13] D. J. Benney and J.C. Luke. On the interactions of permanent waves of finite amplitude. *J. Math. Phys.*, pages 309–313, 1964.

[14] G. P. Bewley, D. P. Lathrop, L. R. M. Maas, and K. R. Sreenivasan. Inertial waves in rotating grid turbulence. *Physics of Fluids*, 19:071701, 2007.

[15] J. Boisson, C. Lamriben, L. R. M Maas, P. Cortet, and F. Moisy. Inertial waves in rotating grid turbulence. *Physics of Fluids*, 24:076602, 2012.

[16] O. Bokhove. On balanced models in Geophysical Fluid Dynamics: Hamiltonian formulation, constraints and formal stability. In Ed.J. Norbury and I. Roulstone, editors, *Large-Scale Atmosphere-Ocean Dynamics: Vol II: Geometric Methods and Models*, pages 1–63. Cambridge University Press, Cambridge, 2002.

[17] O. Bokhove and V. R Ambati. Hybrid Rossby-shelf modes in a laboratory ocean. *J. Phys. Ocean.*, 39(10):2523–2542, 2009.

[18] O. Bokhove and E. R Johnson. Hybrid Coastal and Interior Modes for Two-Dimensional Flow in a Cylindrical Ocean. *J. Phys. Ocean*, 29:93–118, 1999.

[19] O. Bokhove and M. Oliver. Parcel Eulerian-Lagrangian fluid dynamics of rotating geophysical flows. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 462(2073):2575–2592, 2006.

[20] L. J. F. Broer. On the Hamiltonian theory of surface waves. *Applied Scientific Research*, 29(1):430–446, 1974.

[21] G. H. Bryan. The waves on a rotating liquid spheroid of finite ellipticity. *Philos. Trans. Roy. Soc.*, 180:187–219, 1889.

[22] A. Clebsch. *Über die Integration der hydrodynamischen Gleichungen.* 1858.

[23] A. Constantin. A Hamiltonian Formulation for Free Surface Water Waves with Non-Vanishing Vorticity. *Journal of Nonlinear Mathematical Physics*, 12, 2005.

[24] C. Cotter and O. Bokhove. Variational water-wave model with accurate dispersion and vertical vorticity. *Journal of Engineering Mathematics*, 67(1-2):33–54, 2010.

[25] W. Craig. Water waves, Hamiltonian systems and Cauchy integrals. *Institute for Mathematics and Its Applications*, 30:37, 1991.

[26] F. Cucker, P. G. Ciarlet, and J. L. Lions. *Handbook of Numerical Analysis.* Handbook of numerical analysis / general editor, P.G. Ciarlet. North-Holland Publishing Company, 2003.

[27] B. Dintrans, M. Rieutord, and L. Valdettaro. Gravito-inertial waves in a rotating stratified sphere or spherical shell. *Journal of Fluid Mechanics*, 398:271–297, 10 1999.

[28] P. A. M. Dirac. Generalized Hamiltonian dynamics. *Proc.Roy.Soc.Lond.*, A246:326–332, 1958.

[29] D. G. Dommermuth, D. K. P. Yue, W. M. Lin, R. J. Rapp, E. S. Chan, and W. K. Melville. Deep-water plunging breakers: a comparison between potential theory and experiments. *Journal of Fluid Mechanics*, 189:423–442, 3 1988.

[30] M. Feistauer, J. Felcman, and I. Straéskraba. *Mathematical and Computational Methods for Compressible Flow.* Numerical Mathematics and Scientific Computation. Clarendon Press, 2003.

[31] B. A. Finlayson. *The Method of Weighted Residuals and Variational Principles: With Applications in Fluid Mechanics, Heat and Mass Transfer.* Educational Psychology. Academic Press, 1972.

[32] P. D. Fricker and H. M. Nepf. Bathymetry, stratification, and internal seiche structure. *Journal of Geophysical Research: Oceans*, 105(C6):14237–14251, 2000.

[33] Lee-Lueng Fu. Observations and models of inertial waves in the deep ocean. *Reviews of Geophysics*, 19(1):141–170, 1981.

[34] D. Fultz. A note on overstability and the elastoid-inertia oscillations of kelvin, solberg and bjerknes. *J. Meteorol.*, 16:199–208, 1959.

[35] G. P. Galdi. An introduction to the Navier-Stokes initial-boundary value problem. In Giovanni P. Galdi, John G. Heywood, and Rolf Rannacher, editors, *Fundamental Directions in Mathematical Fluid Mechanics*, Advances in Mathematical Fluid Mechanics, pages 1–70. Birkhauser Basel, 2000.

[36] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software.* Pearson Education, 1994.

[37] F. R. Gantmakher. *The theory of matrices. 1.* The Theory of Matrices. Chelsea Publishing Company, 2000.

[38] I. M. Gelfand. *Lectures on Linear Algebra.* Dover Publications Inc., New York, second edition edition, 1989.

[39] T. Gelhard, G. Lube, M. A. Olshanskii, and J. H. Starcke. Stabilized finite element schemes with LBB-stable elements for incompressible flows. *Journal of Computational and Applied Mathematics*, 177(2):243 – 267, 2005.

[40] A. E. Gill. *Atmosphere-Ocean Dynamics.* International Geophysics Series. Academic Press, 1982.

[41] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics - Theory and application to non-spherical stars. 181:375–389, November 1977.

[42] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 1996.

[43] H. P. Greenspan. *The theory of rotating fluids.* Cambridge Monographs on Mechanics and Applied Mathematics. At the University Press, 1990.

[44] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-preserving Algorithms for Ordinary Differential Equations.* Springer series in computational mathematics. Springer-Verlag Berlin and Heidelberg GmbH & Company KG, 2006.

[45] R. Hargreaves. A pressure-integral as kinetic potential. *Phil. Mag.*, 16:436444, 1908.

[46] J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications.* Springer Publishing Company, Incorporated, 1st edition, 2007.

[47] C. W. Hirt and B. D. Nichols. Volume of fluid /VOF/ method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, January 1981.

[48] http://openfvm.sourceforge.net/.

[49] S. R. Idelsohn, E. Oñate, and F. Del Pin. The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves. *International Journal for Numerical Methods in Engineering*, 61(7):964–989.

[50] J. T. Katsikadelis. *Boundary Elements: Theory and Applications: Theory and Applications.* Elsevier Science, 2002.

[51] Lord Kelvin. Vibrations of a columnar vortex. *Philos. Mag.*, 10:155–168, 1880.

[52] B. S. Kirk, J. W. Peterson, R. H. Stogner, and G. F. Carey. `libMesh`: A C++ Library for Parallel Adaptive Mesh Refinement/Coarsening Simulations. *Engineering with Computers*, 22(3–4):237–254, 2006. `http://dx.doi.org/10.1007/s00366-006-0049-3`.

[53] J. J. Kobine. Inertial wave dynamics in a rotating and precessing cylinder. *J. Fluid Mech.*, 303:233–252, 1995.

[54] C. Lamriben, P. P. Cortet, F. Moisy, and L. R. M. Maas. Excitation of inertial modes in a closed grid turbulence experiment under rotation. *Physics of Fluids*, 23:015102, 2011.

[55] P. H. LeBlond and L. A. Mysak. *Waves in the Ocean.* Elsevier, Amsterdam, 1978.

[56] D. Lewis, J. Marsden, R. Montgomery, and T. Ratiu. The Hamiltonian structure for dynamic free boundary problems. *Physica D Nonlinear Phenomena*, 18:391–404, January 1986.

[57] B. Q. Li. *Discontinuous Finite Elements in Fluid Dynamics and Heat Transfer*. Computational Fluid and Solid Mechanics Series. Springer-Verlag London Limited, 2006.

[58] C. C. Lin. Liquid Helium. *Hydrodynamics of helium II, Proc. Int. School of Physics Enrico Fermi*, XXI:93–146, 1963.

[59] M. S. Longuet-Higgins and E. D. Cokelet. The deformation of steep surface waves on water. Part I. A numerical method of computation. *Proc. Roy. Soc. Lond. A*, 350:1–26, 1976.

[60] J. C. Luke. A variational principle for a fluid with a free surface. *Journal of Fluid Mechanics*, 27:395–397, 1967.

[61] L. R. M. Maas. Wave focussing and ensuing mean flow due to symmetry breaking in rotating fluids. *J. Fluid Mech.*, 437:13–28, 2001.

[62] L. R. M. Maas. On the amphidromic structure of inertial waves in a rectangular parallelepiped. *Fluid Dyn. Res.*, 33:373–401, 2003.

[63] J. H. Maddocks and R. L. Pego. An unconstrained hamiltonian formulation for incompressible fluid flow. *Communications in Mathematical Physics*, 170(1):207–217, 1995.

[64] W. V. R. Malkus. Precession of the Earth as the Cause of Geomagnetism: Experiments lend support to the proposal that precessional torques drive the earth's dynamo. *Science*, 160(3825):259–264, 1968.

[65] R. Manasseh. Breakdown regimes of inertia waves in a precessing cylinder. *J. Fluid Mech.*, 243:26, 1992.

[66] R. Manasseh. Visualization of the flows in precessing tanks with internal baffles. *Am. Inst. Aeronaut. Astronaut*, 31:312–318, 1993.

[67] R. Manasseh. Distortions of inertia waves in a rotating fluid cylinder forced near its fundamental mode resonance. *J. Fluid Mech.*, 265:345–370, 1994.

[68] A. M. M. Manders. *Internal wave patterns in enclosed density-stratified and rotating fluids.* 2003.

[69] A. M. M. Manders and L. R. M. Maas. Observations of inertial waves in a rectangular basin with one sloping boundary. *Journal of Fluid Mechanics*, 493:59–88, October 2003.

[70] J. E. Marsden and T. S. Ratiu. *Introduction to Mechanics and Symmetry: A Basic Exposition of Classical Mechanical Systems.* Texts in Applied Mathematics. Springer, 1999.

[71] G. Matthies and G. Lube. On streamline-diffusion methods of inf-sup stable discretisations of the generalised Oseen problem. *Institut für Numerische und Angewandte Mathematik*, Preprint 2007-02, 2007.

[72] J. M. McDonough. *Lectures in elementary fluid dynamics*, 2009.

[73] A. D. McEwan. Inertial oscillations in a rotating fluid cylinder. *J. Fluid Mech.*, 40:603–640, 1970.

[74] D. Michael Milder. A note regarding on hamilton's principle for surface waves. *Journal of Fluid Mechanics*, 83:159–161, 11 1977.

[75] J. W. Miles. On Hamilton's principle for surface waves. *Journal of Fluid Mechanics*, 83:153–158, November 1977.

[76] P. J. Morrison. Hamiltonian description of the ideal fluid. *Rev. Mod. Phys.*, 70:467–521, Apr 1998.

[77] P. J. Morrison and J. M. Greene. Noncanonical Hamiltonian Density Formulation of Hydrodynamics and Ideal Magnetohydrodynamics. *Phys. Rev. Lett.*, 45:790–794, Sep 1980.

[78] S. Nurijanyan, O. Bokhove, and L. R. M. Maas. New semi-analytical solution for inertial waves in a rectangular parallelepiped. *Phys. of Fluids*, 2013.

[79] S. Nurijanyan, J. J. W. van der Vegt, and O. Bokhove. Hamiltonian discontinuous Galerkin FEM for linear, rotating incompressible Euler equations: Inertial waves. *Journal of Computational Physics*, 241(0):502 – 525, 2013.

[80] L. Pesch, A. Bell, H. Sollie, V. R. Ambati, O. Bokhove, and J. J. W. van der Vegt. hpGEM A software framework for discontinuous Galerkin finite element methods. *ACM Trans. Math. Softw*, 2007.

[81] Charles S. Peskin. The immersed boundary method. *Acta Numerica*, 11(-1):479–517, 2002.

[82] O. M. Phillips. Energy Transfer in Rotating Fluids by Reflection of Inertial Waves. *Physics of Fluids*, 6(4):513–520, 1963.

[83] D. A. Di Pietro and A. Ern. Mathematical Aspects of Discontinuous Galerkin Methods, 2012.

[84] J. Proudman. On the dynamic equation of the tides. Parts 1-3. *Proceedings of the London Mathematical Society, Series 2*, 18:168, 1917.

[85] J. Proudman. Note on the free tidal oscillations of a. sea. with slow rotation. *Proc. Lond. Math. 800*, 35:75–82, 1933.

[86] D. Rao. Free gravitational oscillations in rotating rectangular basins. *J. Fluid Mech.*, 25:523–555, 1966.

[87] Lord Rayleigh. Notes concerning tidal oscillations upon a rotating globe. *Proc. R. Soc. Lond.*, 26:448–464, 1909.

[88] M. Rieutord. Inertial modes in the liquid core of the Earth. *Physics of the Earth and Planetary Interiors*, 91(1-3):41–46, 1995.

[89] M. Rieutord and L. Valdettaro. Inertial waves in a rotating spherical shell. *Journal of Fluid Mechanics*, 341:77–99, 5 1997.

[90] H. Rund. *The Hamilton-Jacobi theory in the calculus of variations: its role in mathematics and physics.* New university mathematics series. Van Nostrand, 1966.

[91] K. S. Balay, Buschelman, D. W. Gropp, D. Kaushik, G. M. Knepley, C. L. Mcinnes, F. B. Smith, and H. Zhang. PETSc Web page, 2001.

[92] R. Salmon. Hamiltonian fluid mechanics. *Annu. Rev. Fluid Mech*, 20:225pp, 1988.

[93] R. Salmon. Semigeostrophic theory as a Dirac-bracket projection. *Journal of Fluid Mechanics*, 196:345–358, 10 1988.

[94] R. Scardovelli and S. Zaleski. Direct Numerical Simulation of Free-Surface and Interfacial Flow. *Annual Review of Fluid Mechanics*, 31:567–603, January 1999.

[95] J. Serrin. Mathematical Principles of Classical Fluid Mechanics. *Handbuch der Physik*, 8:125–263, 1959.

[96] S. J. Sherwin and G. E. Karnaidakis. Spectral/hp Element Methods for Computational Fluid Dynamics. *Oxford Science Publications*, 2005.

[97] K. Sheshadri and P. Fritzson. `libMesh`: MathPDE: A Package to Solve PDEs by Finite Differences. *The Mathematica journal*, 13, 2011.

[98] K. Stewartson. On trapped oscillations of a rotating fluid in a thin spherical shell. *Tellus*, 23(6):506–510, 1971.

[99] K. Stewartson. On trapped oscillations of a rotating fluid in a thin spherical shell II. *Tellus*, 24(4):283–287, 1972.

[100] B. R. Sutherland. *Internal Gravity Waves.* Cambridge University Press, 2010.

[101] G. I. Taylor. Tidal oscillations in gulfs and basins. *Proceedings of the London Mathematical Society, Series 2, 148–181*, 1921.

[102] R. O. R. Y. Thompson. A mechanism for angular momentum mixing. *Geophys. Astrophys. Fluid Dyn.*, 12:221–234, 1979.

[103] J. J. W. van der Vegt, F. Izsák, and O. Bokhove. Error Analysis of a Continuous-Discontinuous Galerkin Finite Element Method for Generalized 2D Vorticity Dynamics. *SIAM Journal on Numerical Analysis*, 45(4):1349–1369, 2007.

[104] H. van Haren and C. Millot. Rectilinear and circular inertial motions in the Western Mediterranean Sea. *Deep Sea Research Part I: Oceanographic Research Papers*, 51(11):1441 – 1455, 2004.

[105] J. Vanneste and O. Bokhove. Dirac-bracket approach to nearly geostrophic Hamiltonian balanced models. *Physica D: Nonlinear Phenomena*, 164(34):152 – 167, 2002.

[106] J. Vanyo, P. Wilde, P. Cardin, and P. Olson. Experiments on precessing flows in the earth's liquid core. *Geophys. J. Int.*, 121:136–142, 1995.

[107] H. H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method.* Pearson Education Limited, 2007.

[108] G. B. Whitham. Variational Methods and Applications to Water Waves. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 299(1456):6–25, 1967.

[109] Y. Xu, J. J. W. van der Vegt, and O. Bokhove. Discontinuous Hamiltonian Finite Element Method for Linear Hyperbolic Systems. *J. Sci. Comput.*, 35(2-3):241–265, June 2008.

[110] V. E. Zakharov. Stability of periodic waves of finite amplitude on the surface of a deep fluid. *Journal of Applied Mechanics and Technical Physics*, 9(2):190–194, 1968.

Acknowledgements

“We do not know what we want and yet
we are responsible for what we are – that is the fact.”
Jean-Paul Sartre

This thesis was a collaborative effort and it would have not reached its completion without the guidance of my supervisors, the assistance of our collaborators, help from colleagues and friends, and support from my family and fiancé.

First, I want to thank Jaap van der Vegt and Onno Bokhove for giving me the opportunity to try myself in the scientific world and for the never ending motivation and encouragement during ups and downs of this endeavour. Jaap, I am deeply grateful for your patience and wisdom, which helped me to get to the end of this interesting quest. You have always shown me the way to improve in all aspects. Onno, your guidance and support have helped me through these years, your enthusiasm and optimism have always made me to go forward even in, from the first glance, impossible situations. Here, I also want to thank Leo Maas for the exciting and fruitful collaboration. Leo, you have always shown a great interest in my work and I have enjoyed every minute of working with you; the all-around-the-Netherlands journeys from Enschede to Texel were well worth it.

Besides my supervisors, other members of the MACS group have provided an invaluable scientific, practical and emotional support; I will avoid to name everybody as I am afraid to leave someone accidentally out, but I really appreciated you all guys. Furthermore, I would like to express my gratitude to our secretaries Marielle and Linda, who could solve almost every problem spiced with a cheerful smile.

Many thanks to my university lunch-group; you have made an entertaining and enlightening act out of boring food consumption process.

It is impossible to underestimate the importance of Stefan Luding in the existence of the written version of this thesis. Stefan sheltered me for half a year in his wonderful group nurtured by lovely Sylvia, giving me time to successfully finish the writing. Thank you MSM group.

A big portion of gratitude goes to my paranymphs–friends, who I know will be by my side on the 9th of October, as they have always been there to give me a hand, whenever I really needed it. Thanks Ove (+Elena) and Anthony.

I am also gracefully thankful for the opportunity to gain a lot of friends not only in Enschede but all over the world. Guys, you have helped me to properly enjoy the small amount of free time that this thesis left me. I will never forget the trips, concerts, diners, football matches in the virtual and real fields, numerous BBQs and other stupid adventures.

Many words of appreciation to two Manukyan families, who ease my homesick in the difficult times and *"Hello* to *world!"* the wonderful new members of their families (Levon & Eva), who immediately earned my honest friendship.

Further, I would like to thank my parents and my sister, for their constant support and encouragement. Last but not least I am very grateful to my fiancé for her patience and belief in me. Lilit, whatever I put in words here, will not be enough. Thank you.