Imperial College London
Department of Earth Science and Engineering

Developing and validating a flexible unstructured ocean model with adjoint capability for flow under ice shelf cavities using the Firedrake finite element framework

William Scott

Supervised by
Prof Matthew Piggott
Prof Martin Siegert
Dr Stephan Kramer
Dr Paul Holland
Dr Keith Nicholls

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computational Geoscience

March 2024

Statement of Originality

I hereby declare that the contents of this thesis are my own work and that any external resources that I have used have been clearly referenced in the text.

William Ivor Scott, 20th July 2023.

Copyright Declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-Non Commercial 4.0 International Licence (CC BY-NC).

Under this licence, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the licence terms clear to others by naming the licence and linking to the licence text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes. Please seek permission from the copyright holder for uses of this work that are not included in this licence or permitted under UK Copyright Law.

Abstract

Accurate modelling of basal melting beneath ice shelves is key to reducing the uncertainty in forecasts of ice shelf stability and, thus, the polar contribution to sea level rise. Numerical ocean models are important tools to investigate these critical and remote regions. The relative inflexibility of structured grid models means, however, that they can struggle to resolve these processes in realistic cavity geometries.

We present a new nonhydrostatic unstructured mesh model for flow under ice shelves built using the Firedrake finite element framework. Verification tests based on a novel Method of Manufactured Solutions (MMS) test, including melting, shows that the model discretisation is sound and second order accurate. This test may have wider applicability as a rigorous code verification procedure for the ice-ocean modelling community.

In detailed testing our results are shown to compare favorably, in terms of accuracy and robustness, against those obtained with the popular MITgcm model for a series of benchmarks of increasing complexity and different meshing techniques, culminating with a full 3D ice shelf cavity domain using the community standard ISOMIP+ *Ocean0* test case. We build on existing work to show how spurious numerical mixing dictates the sensitivity of the melt rate parameterisation to grid resolution.

A key motivation behind using Firedrake is the availability of an automatically generated adjoint model. Preliminary adjoint sensitivity calculations are promising and point to the ability to address important questions on ocean influence on ice shelf vulnerability in the future.

Finally, we apply the forward model to investigate the dynamics of ocean flow in the presence of complicated basal features. We extend the literature, previously confined to 2D, to consider the importance of crevasse orientation and rotational effects for a suite of 3D experiments.

Acknowledgements

I would like to thank all my supervisors Matt Piggott, Stephan Kramer and Martin Siegert at Imperial College London, as well as Paul Holland and Keith Nicholls at the British Antarctic Survey. They have all given me so much time and good advice in every aspect of my PhD.

I am especially grateful for being part of the International Thwaites Glacier Collaboration (ITGC) right from the beginning of my project when I attended an early career retreat in Bury St Edmunds in Suffolk, and later when I also had the chance to participate in a meeting in Boulder, Colorado. I met so many nice and engaging scientists through these trips and they fuelled my motivation for ice-ocean research. I also enjoyed and gained inspiration from meeting the FRISP ice shelf community. Similarly, interacting with the PRISM and Firedrake communities have widened my knowledge of best practices in numerical methods.

I would like to thank the Science and Solutions for a Changing Planet DTP for funding my work and the friends I made from Cohort 6. Thanks are also due to the Research Computing Service for providing HPC facilities and support for this work.

Finally, I would like to thank my friends and family, in particular my parents for supporting me all the way through.

Contents

Statement of Originality i

Abstract i

Acknowledgements iii

1 Introduction 1

2 Model Development and Verification 9

2.1 Model Equations 10

2.2 Weak Form of Model Equations 14

2.2.1 Tracers 14

2.2.2 Velocity-Pressure 15

2.3 Finite Element Discretisation: Fully Unstructured and 'Extruded' Meshes 17

2.3.1 Tracers 17

2.3.2 Velocity-Pressure 18

2.4 Discontinuous Galerkin Surface Integrals 21

2.5 Timestepping and Solver Strategy 30

2.6 Melt Parameterisation 36

2.7 Slope Limiters 41

2.8 Model Implementation 45

2.9 Method of Manufactured Solution (MMS) tests 46

2.10 Ekman Spiral 52

2.11 Code Availability 56

2.12 Conclusion 57

3 Model Benchmarking vs MITgcm 58

3.1 Introduction 59

3.2 2D Benchmark vs MITgcm for an Idealised Grounding Zone Domain 60

3.3 2.5D Benchmark vs MITgcm for an Idealised Grounding Zone Domain 65

3.4 2D Benchmark vs MITgcm for an Extended ISOMIP-like Domain 68

3.5 2D z vs σ-coordinate Model Comparison for an Extended ISOMIP-like Domain . 71

3.6 3D Benchmark vs MITgcm for the ISOMIP+ Ocean0 Test Case 75

3.6.1 Melt Sensitivity to Vertical Resolution 78

3.6.2 Melt Sensitivity to Horizontal Resolution 89

3.7 Performance Benchmarking vs MITgcm 95

3.8 Code Availability 96

3.9 Conclusion 98

4 Adjoint Sensitivity Calculations Applied to Ice Shelf Cavities **101**

4.1 Introduction . . . 102

4.2 Adjoint System of Equations . . . 104

4.3 Adjoint Sensitivity Calculations in a 2D Idealised Grounding Zone Domain . . . 109

4.4 Adjoint Sensitivity Calculations with a Depth Profile Objective Function . . . 119

4.5 Adjoint Performance . . . 128

4.6 Code Availability . . . 130

4.7 Conclusion . . . 133

5 Modelling 3D Ice Shelf Basal Crevasses **135**

5.1 Introduction . . . 136

5.2 Frazil Ice Dynamics . . . 138

5.3 2D Crevasse Simulation with Frazil Ice Dynamics . . . 141

5.4 Geometry and Rotational Effects for 3D Basal Crevasses . . . 146

5.5 Code Availability . . . 166

5.6 Conclusion . . . 167

6 Conclusion **168**

Bibliography **171**

List of Tables

2.1 Iteration counts for 2D z-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing. 27

2.2 Iteration counts for 2D σ-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing. 28

2.3 Iteration counts for 2D z-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing using the new penalty parameter formulation, as described in Section 2.4. 28

2.4 Iteration counts for 2D σ-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing using the new penalty parameter formulation, as described in Section 2.4. 29

2.5 General constants for model equations and melt parameterisation. 39

2.6 MMS convergence tests for temperature, salinity, melt, integrated melt, velocity and pressure. 52

2.7 Parameters and constants used for MMS convergence test in Section 2.9. 54

3.1 Parameters and constants used for the ISOMIP+ *Ocean0* test case in Section 3.6. 94

3.2 Performance benchmarking vs MITgcm for 2D simulations. 95

3.3 Performance benchmarking vs MITgcm for 3D simulations. 97

4.1 Taylor remainder convergence tests for temperature, salinity, viscosity, temperature diffusivity, salinity diffusivity and turbulent exchange coefficient control fields. 111

4.2 Parameters and constants used for adjoint sensitivity calculations in the idealised grounding zone domain from Section 4.3. 119

4.3 Taylor remainder convergence tests for temperature, salinity, viscosity, boundary temperature and boundary salinity control fields for the depth profile objective function. 121

4.4 Adjoint performance for simulations using the coarse mesh ($\Delta x = 500$m, $\Delta z = 2$ m) run for ten timesteps. 131

4.5 Adjoint performance for simulations using the fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m) run for ten timesteps. 131

4.6 Adjoint performance for simulations using the fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m) run for 100 timesteps. 131

4.7 Adjoint performance for simulations using the fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m) run for 100 timesteps using iterative solvers. 132

4.8 Adjoint memory consumption for simulations using the coarse ($\Delta x = 500$m, $\Delta z = 2$ m) and fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m). 132

List of Figures

2.1 Schematic of 3D extruded elements used in Section 3.6. 18

2.2 Pressure fields for a 2D early, pre-ISOMIP+ (see Section 3.6) simulation using the unstable velocity-pressure element pair DQ1-Q2 (top) and the stable RTCE2-Q2 pair (bottom). 20

2.3 Effect of penalty parameter for 1D test case for three resolutions. 24

2.4 Horizontal velocity fields at 50 days for a 2D early, pre-ISOMIP+ (see Section 3.6) simulation using the unstable velocity-pressure element pair DQ1Q2 (top), the stable RTCE2-Q2 pair before changes to the Interior Penalty (IP) term (middle), and RTCE2-Q2 pair after changes to the IP term (bottom). 29

2.5 2D vertical schematic of melt parameterisation and implementation of tracer boundary conditions. 40

2.6 Melt rate profile at 50 days for a 2D early, pre-ISOMIP+ (see Section 3.6) simulation showing the effects of limiters on a spurious freezing signal at the ice front. 42

2.7 Melt rate profile for Firedrake and MITgcm showing that applying limiters on velocity components significantly affects melt rate. 44

2.8 Example MMS solution and error fields. 48

2.9 MMS convergence tests for temperature, salinity, melt, integrated melt over the top boundary, velocity and pressure. 53

2.10 Ekman spiral results for 2.5D and 3D simulations compared with analytical solution. 56

3.1 Plots of temperature for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days. 61

3.2 Plots of salinity for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days. 62

3.3 Plots of horizontal velocity for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days. 62

3.4 Plots of vertical velocity for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days. 63

3.5 Melt rate profile along ice shelf base for the 2D idealised grounding zone comparison after 40 days. 64

3.6 Plots of temperature for the 2.5D idealised grounding zone comparison between MITgcm and Firedrake after 40 days. 66

3.7 Plots of u velocity (positive 'out of the page') for the 2.5D comparison idealised grounding zone between MITgcm and Firedrake after 40 days. 66

3.8 Plots of v velocity (positive to the right) for the 2.5D idealised comparison grounding zone between MITgcm and Firedrake after 40 days. 67

3.9 Plots of vertical velocity for the 2.5D idealised grounding zone comparison between MITgcm and Firedrake after 40 days. 67

3.10 Plots of temperature for the 2D pre-ISOMIP+ comparison between MITgcm and Firedrake after 100 days. 70

3.11 Plots of horizontal velocity for the 2D pre-ISOMIP+ comparison between MITgcm and Firedrake after 100 days. 70

3.12 Melt rate profile for the 2D pre-ISOMIP+ comparison between MITgcm and Firedrake at 50 days. 71

3.13 Example of z-layer (constant vertical resolution) and σ-layer (fixed number of layers) meshes built using Firedrake's 'extruded' mesh functionality. 74

3.14 Melt rate profile for the 2D pre-ISOMIP+ comparison between z and σ-layer meshes using Firedrake after 50 days. 75

3.15 Integrated melt rate over deepest region, where ice draft is deeper than 300 m following Asay-Davis et al. [2016]. 76

3.16 Plan view of basal melt at 50 days for MITgcm at target ISOMIP+ resolution for z-layer models (Δx: 2 km, Δz: 20 m), along with finer vertical resolution (Δz: 10 m, 5 m, 2.5 m) and Firedrake at target ISOMIP+ resolution for terrain following model (Δx: 2 km, 36 levels with 30 levels inside the ice shelf cavity). . 80

3.17 Cross section across ice shelf cavity at $x = 480$ km, close to the grounding line. 81

3.18 Depth profile of temperature at, x = 480 km, y = 40 km, which corresponds to the middle of Figure 3.17. 83

3.19 Comparison of 3D ISOMIP+ meshes. 84

3.20 Depth profile of temperature at, x = 480 km, y = 40 km, which corresponds to the middle of Figure 3.17. 85

3.21 Integrated melt rate with Firedrake run at different vertical resolutions and vertical diffusivity values. The melt rate is integrated over the deepest region, where ice draft is deeper than 300 m following Asay-Davis et al. [2016]. 87

3.22 Plan view of base meshes used to investigate sensitivity to horizontal resolution. 90

3.23 Plan view of basal melt rates for different horizontal grids. 91

3.24 Integrated melt rate with MITgcm run at different vertical resolutions and horizontal resolutions. 93

3.25 Plan view melt rate after 50 days for MITgcm run with $\Delta z = 2.5$ m and varying horizontal resolution, $\Delta x = 4$ km, 2 km, 1 km and Firedrake $\Delta x = 2$ km, 36 layers (ISOMIP+ default). 94

4.1 Schematic for an adjoint advection problem from Equations 4.13 and 4.15 showing the spatial discretisation. 108

4.2 Temperature and velocity fields at 50 days after the forward model has spun up. 110

4.3 Adjoint sensitivity fields of final total melt flux with respect to temperature (left) and salinity (right) at three time intervals 113

4.4 Adjoint sensitivity fields of final total melt flux with respect to temperature (left) and salinity (right) scaled by the expansion coefficients for the equation of state ($\alpha_T = -2 \times 10^{-4}\,^{\circ}\mathrm{C}^{-1}$, $\beta_S = 7 \times 10^{-4}$). 114

4.5 Sensitivity of total basal melt at the final time step to vertical viscosity and eddy diffusivity for temperature and salinity calculated with respect to the L_2 norm . 116

4.6 Sensitivity of total basal melt at the final time step to the turbulent heat exchange coefficient Γ_T, calculated with respect to the L_2 'boundary' norm (see main text for explanation). 118

4.7 Melt profile at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (purple) and linear stratification (black). 122

4.8 Horizontal velocity at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (top) and linear stratification (bottom). 123

4.9 Salinity field at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (top) and linear stratification (bottom). 124

4.10 Depth profile of salinity at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (purple) and linear stratification (black). 125

4.11 Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 1 hour before the final time calculated with respect to the L_2 norm. 125

4.12 Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 6 hours before the final time calculated with respect to the L_2 norm. 126

4.13 Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 1 day before the final time calculated with respect to the L_2 norm. 126

4.14 Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 5 days before the final time calculated with respect to the L_2 norm. 127

4.15 Sensitivity of objective function based on depth profile to the static salinity (blue) and temperature (red) boundary conditions on the open boundary 20 days before the final time calculated with respect to the L_2 boundary norm. 128

4.16 Sensitivity of objective function based on depth profile to vertical viscosity (top) and eddy diffusivity for and salinity (top) temperature (bottom) calculated with respect to the L_2 norm. 129

5.1 Time averaged plots of frazil ice concentration and velocity vectors for the 2D crevasse based on the study by Jordan et al. [2014]. 143

5.2 Frazil ice simulations from carried out by Jordan et al. [2014] (Figure 5). Note the same counterclockwise circulation as our results in Figure 5.1, as well as Frazil ice production concentrated in the crevasse corners. 144

5.3 Temperature depth profile at 'comparison' and 'calibration' profile locations from Jordan et al. [2014]. 144

5.4 Temperature and salinity depth profile at 'comparison' and 'calibration' profile locations from Jordan et al. [2014]. 145

5.5 Plan view of meshes used in the 3D crevasse geometry experiment at a depth slice of 525 m. 148

5.6 Cross section of meshes used in the 3D crevasse geometry experiment along y = 2500 m. 149

5.7 Cross section of meshes used in the 3D crevasse geometry experiment along x = 2500 m. 149

5.8 Depth slice of velocity magnitude at 550 m for the 3D crevasse geometry experiment after 10 days. 152

5.9 Depth slice of velocity magnitude at 510 m for the 3D crevasse geometry experiment after 10 days. 153

5.10 Depth slice of velocity magnitude at 502.5 m for the 3D crevasse geometry experiment after 10 days. 154

5.11 Depth slice of velocity magnitude at 497.5 m for the 3D crevasse geometry experiment after 10 days. 155

5.12 Depth slice of velocity magnitude at 475 m for the 3D crevasse geometry experiment halfway inside the crevasse after 10 days. 156

5.13 Depth slice of vertical velocity at 550 m for the 3D crevasse geometry experiment after 10 days. 157

5.14 Depth slice of vertical velocity at 510 m for the 3D crevasse geometry experiment after 10 days. 158

5.15 Depth slice of vertical velocity at 502.5 m for the 3D crevasse geometry experiment after 10 days. 159

5.16 Depth slice of vertical velocity at 497.5 m for the 3D crevasse geometry experiment after 10 days. 160

5.17 Plan view of melt rates for the 3D crevasse geometry experiment after 10 days. 161

5.18 Transect of melt rates for the 3D crevasse geometry experiment along y = 2500 m after 10 days. 162

5.19 Transect of melt rates for the 3D crevasse geometry experiment along y = 2500 m after 10 days zoomed to the crevasse region. 163

5.20 Transect of melt rates for the 3D crevasse geometry experiment along x = 2500 m after 10 days. 164

5.21 Transect of melt rates for the 3D crevasse geometry experiment along x = 2500 m after 10 days zoomed to the crevasse region. 165

Chapter 1

Introduction

There are significant uncertainties surrounding how the polar ice sheets will react to climate change. Considering 40% of the world's population live in coastal regions [Millennium ecosystem assessment, 2005], and that there is enough ice locked up in Greenland to raise sea level by 7 metres [Morlighem et al., 2021] and in Antarctica by 58 metres [Fretwell et al., 2013], it is important to quantify how fast, and how much ice could be lost.

There are three main processes by which ice is lost from polar ice sheets: surface melting, calving of icebergs, and melting beneath ice shelves. Surface melting is very low for most parts of Antarctica because surface temperatures are so cold [Cuffey and Paterson, 2010]. In Greenland however this is an important process, as seasonal melt percolates to the bed and feeds subglacial channels, which can affect ice velocity [Sole et al., 2011]. Many glaciers in Antarctica flow out into the ocean [Jacobs et al., 1992]. If the ocean is deep enough to support the weight of the glacier, then a floating ice shelf can form. Glaciers in these environments are referred to as marine terminating glaciers [Cuffey and Paterson, 2010].

Recent observations have shown that outlet glaciers and ice streams in West Antarctica are retreating at an alarming rate [Mouginot et al., 2014, Scambos et al., 2017]. In total there is enough ice in West Antarctica to raise sea level by approximately 4.3 m [Fretwell et al., 2013]. Evidently, this has serious implications for communities that live in low lying coastal regions. There are still major uncertainties, however, in predictions of how much ice could be

lost and, importantly, on what time scales this may occur. Increased basal melting beneath ice shelves, due to increased ocean heat, is thought to be the main cause for ice loss in West Antarctica [Rignot et al., 2013]. Wherever ice shelves are topographically confined, they provide buttressing forces to the grounded glaciers [Shepherd et al., 2018]. When ice shelves thin, a decrease in buttressing can lead to an increase in ice flow speeds and hence the rate at which grounded ice is lost increases. Once ice flows from the grounded ice sheet to the floating ice shelf it is effectively lost to the ocean, due to Archimedes principle [Jenkins and Holland, 2007]. Ice that is initially grounded above sea level will raise sea level once it is part of the floating ice shelf. Predicting sea level rise is a pressing issue, but there are also questions about how changes in melt water flux to the ocean can affect global ocean dynamics [Dinniman et al., 2016]. Improving understanding of ocean processes beneath ice shelves and their impact on ice shelf vulnerability is therefore an important problem in glaciology and climate science.

Numerical models of ocean flow beneath ice shelves play a key role in our understanding of ocean cavities, since direct observations are so limited. The model needs to resolve the interaction between salinity, temperature and the flow, along with the thermodynamics at the complex, sloping ice-ocean interface that characterise ice shelf cavity environments. Melting ice injects buoyant, fresh water into the domain, which in turn drives overturning ocean circulation within the cavity. The freezing point of water decreases with depth so, in some cases, ice that melts at depth in the cavity can refreeze closer to the ocean surface, thus transporting ice up the underside of the ice shelf. This is known as the 'ice pump' mechanism [Lewis and Perkin, 1986, Jenkins and Bombosch, 1995].

Another key piece of theory which has motivated the study of marine terminating glaciers is known as the Marine Ice Sheet Instability (MISI) [Mercer, 1978, DeConto and Pollard, 2016]. MISI is a mechanism for rapid loss of ice into the ocean. It is a positive feedback between ice flux and geometry. The flux of ice through the grounding zone is a function of the ice thickness above the grounding line. If the bedrock slopes inland then the thickness above the grounding line will increase as the ice retreats, resulting in a higher ice flux through the grounding zone. If this is not balanced by a corresponding increase in the rate of snowfall, then the glacier will retreat until equilibrium is reached [Cuffey and Paterson, 2010].

West Antarctica is predominantly below sea level and bathymetry deepens towards the centre of Antarctica [Jacobs et al., 1992]. Thwaites glacier is retreating particularly rapidly, with a significant acceleration since 2006 [Mouginot et al., 2014]. This has prompted an international investigation to conduct the largest fieldwork campaign in polar science history to study this glacier [Scambos et al., 2017]. It is not just West Antarctica which deserves attention. Updated estimates of bed topography for Totten glacier in East Antarctica has suggested it also could be at risk of MISI [Greenbaum et al., 2015]. This is particularly worrying because the catchment of Totten Glacier alone has the potential to raise sea level by 3.5 m [Greenbaum et al., 2015].

In coastal regions where changes in coastlines and bathymetry can have important effects on dynamics, horizontally unstructured grid models have proven to be extremely useful tools [Kärnä et al., 2018]. In an ice shelf cavity setting, this is doubly important since the base of the ice shelf can vary as much as the seafloor, with the formation of basal crevasses and channels that evolve with the ocean, and indeed ice, flow. The lack of flexibility in traditional structured grid ocean models in resolving these features means that there is still large uncertainty in flow dynamics beneath ice shelves [Dinniman et al., 2016]. This is compounded by the fact that results using the commonly employed 'three-equation melt parameterisation' depend significantly on vertical resolution and implicitly on the model's choice of vertical discretisation [Gwyther et al., 2020] – an issue that will be explored further in Section 3.6.

Satellite measurements indicate that spatial melt rate patterns near grounding zones (the region separating the grounded ice sheet from the floating ice shelf) are highly variable [Milillo et al., 2019]. This implies that there must be a complex combination of processes in the grounding zone that need to be resolved to accurately model melt rates. Due to the limitations on grid resolution imposed by structured grids used by traditional ocean models, it has not been possible to investigate in detail melting at grounding zones. Ensemble simulations from glacial flow models have shown that the largest uncertainties in projected ice loss come from estimating melt rates at the grounding zone [Arthern and Williams, 2017]. Similarly, Goldberg et al. [2019] used an ice flow model with an associated adjoint model to investigate how sensitive the volume of ice loss was to spatially varying melt for the Dotson and Crosson ice shelves. Again, they found that ice loss was most sensitive to melting at the grounding zone. An extension of this

study to the wider Amundsen Sea Embayment confirmed the high sensitivity of ice loss to melting at the grounding zone [Morlighem et al., 2021]. Therefore, accurately simulating ocean dynamics near the grounding zone seems to be one of the most important tasks required to help reduce uncertainty in ice loss estimates, and consequently reduce uncertainty in sea level projections.

The finite element method is particularly well suited to solving problems in complicated domains [Elman et al., 2014]. The power of the finite element method comes from being able to define the solution using piecewise functions which can be defined on completely arbitrary meshes. This is ideal for modelling flow in complicated domains, such as ice shelf cavities. A fully unstructured finite element ocean model, Fluidity [Piggott et al., 2008a], was previously adapted to enable simulation of ice shelf cavities [Kimura et al., 2013]. Due to its ability to run on fully unstructured grids, Fluidity was able to resolve all the key features of an ice shelf cavity (pinching grounding lines, sloping ice base and the steep vertical ice front; Kimura et al. 2013), as well as investigate flow within basal crevasses [Jordan et al., 2014].

The model presented here is similarly motivated by the applicability of finite elements on unstructured meshes and the need to retain a full representation of physics, valid at order one aspect ratios, for buoyancy driven flows in domains with complicated geometries, such as ice shelf cavities. A significant departure from previous work is use here of the Firedrake toolkit to implement the underlying discretisation of the finite element method [Rathgeber et al., 2016]. The philosophy behind Firedrake has several advantages. By separating the underlying software implementation from the physics of the system, it enables a Firedrake user to quickly develop and test new discretisation methods on the problem in question. The syntax of the Unified Form Language (UFL) that Firedrake employs is designed to closely mimic the mathematical description of the weak form prescribing the finite element discretisation [Alnæs et al., 2014]. This, combined with the readability of the Python programming language, helps to improve the longevity of the code by reducing the learning curve required for new users before they can start using and implementing features in the model. 'Icepack', a glacier flow modelling package built on top of Firedrake, is a good example of this [Shapero et al., 2021]. The Firedrake framework automatically translates this high-level mathematical description of the numerics

into highly optimized low-level C code which leads to an efficient implementation of the finite element method. In addition, Firedrake uses the Portable, Extensible Toolkit for Scientific Computation (PETSc) a state-of-the-art library to efficiently solve the systems that arise from the finite element discretisation [Balay et al., 1997].

There are already ocean models that use horizontally unstructured grids to investigate ice shelf cavities, such as FESOM [Timmermann et al., 2012], FVCOM [Zhou and Hattermann, 2020] and MPAS-O [Ringler et al., 2013, Gwyther et al., 2020]. These models typically retain a distinct difference in how they deal with horizontal and vertical motion in the model. In part this means making the hydrostatic approximation: provided the horizontal length scales are much larger than the vertical, the vertical acceleration terms can be neglected, which reduces the computational cost. This approximation is evidently true for the open ocean when typically the meshes used for simulations have horizontally stretched, anisotropic grids. To investigate interesting basal features under ice shelves though, such as crevasses [Jordan et al., 2014] or in the proximity of topographically complex grounding zone regions, one may want to have more flexibility by using meshes where the aspect ratio of the grid becomes close to order one, i.e., the horizontal grid sizes are comparable to the vertical grid cell sizes. In this case the hydrostatic approximation may no longer be valid. Therefore we include the nonhydrostatic vertical acceleration terms as part of the solve. Moreover, by treating the horizontal and vertical equations on the same footing, our model has been designed so that it can easily incorporate isotropic turbulence closure schemes from the CFD literature, similar to work carried out by Yeager [2018], whilst at the same time being able to transition to more traditional GFD turbulence closures in high aspect ratio domains, where vertical and horizontal mixing is parameterised separately.

A key advantage of using a framework such as Firedrake, or similar projects such as Fenics [Alnæs et al., 2015], is the fact that it readily facilitates the use of an automatically generated adjoint model, thanks to the work of the dolfin-adjoint project [Farrell et al., 2013, Mitusch et al., 2019]. Adjoint modelling represents an efficient means to calculate gradients of model output functionals with respect to model inputs [Errico, 1997]. By solving the adjoint system the gradient of an output functional can be calculated with respect to any number of input

parameters, independent of the number of these input parameters. Adjoint modelling is often the only tractable method for calculating gradient information for numerical models that rely on grid-based discretisations, like ocean models, because the number of input parameters scales with the grid resolution [Errico, 1997]. Early implementations of adjoint models were developed by hand separately from forward models [Kalnay, 2003]. As the complexity of the forward code develops this approach becomes time consuming and error prone. A more sophisticated method uses what is sometimes referred to as automatic differentiation (AD) of the forward code, by using the chain rule on each line of forward code repeatedly to generate an adjoint model [Errico, 1997]. Usually, careful intervention by the user is required during development and maintenance, especially with regards to ensuring the computational efficiency of the adjoint model. Hence it is common to refer to this process as algorithmic, not automatic, differentiation. The approach implemented by the dolfin-adjoint project makes use of the high-level symbolic representation of the discrete mathematical problem written in UFL [Mitusch et al., 2019]. Since the mathematical problem is separate from its software implementation, provided the mathematical form of the equations is differentiable then the process is automatic. The adjoint model is consistent with the forward discretisation and automatically inherits the solver strategies and the parallelisability of the forward model [Mitusch et al., 2019].

Sensitivities of a functional with respect to different inputs are very useful in their own right [Errico, 1997]. In a time-dependent forward model, the equivalent adjoint model propagates sensitivities backwards through time, from the final time to the initial time. This complimentary information can be very useful when trying to understand modelled processes that may not be obvious from considering the forward model alone. Gradient information is also very useful for optimisation problems, in particular parameter estimation and data assimilation. In many ocean modelling studies there are many parameters that are unknown, but can be constrained by observations. Gradient information can be used to update these parameters in an efficient manner by minimising a functional that measures the error between observed and modelled values.

There have been a limited number of studies of an ocean adjoint in an ice shelf cavity context. The first used the MITgcm ocean model to simulate the Pine Island Ice Shelf cavity [Heimbach

and Losch, 2012]. Goldberg et al. [2020] extended MITgcm's adjoint model, such that it was able to calculate the sensitivity of melt rate to bathymetry. Notably, this was using the open source AD tool, openAD [Utke et al., 2008]. Most recently the ECCO2 assimilation framework (which also relies on MITgcm's adjoint capability) was applied to the Amundsen and Bellingshausen seas and ice shelf cavities [Nakayama et al., 2021b]. Since all the previous adjoint modelling work for ice shelf ocean cavities have used the MITgcm framework, they inherit the relative inflexibility of MITgcm's structured grid. Although MITgcm's resolution can vary in space (Δx can vary horizontally) Δz is limited to varying with depth. This has limited practical application for ice shelf cavities since the ocean cavity thickness varies significantly from the grounding zone to the open ocean. We see our model, which can combine flexible unstructured meshes with the adjoint capability, as a tool with significant potential for investigating ocean conditions in the complex domains that are ice shelf cavities. We believe this unique capability justifies developing a new model to investigate ice ocean interactions and, crucially, that this is a sustainable endeavour due to Firedrake's philosophy of separating the different concerns of model development between scientists, mathematicians and engineers to produce highly effective code in an efficient manner.

The outline for the rest of this thesis is as follows: Chapter 2 covers the model equations, discretisation choices and the verification testing that is vital for the new model developments implemented as part of this work. This includes a novel Method of Manufactured Solutions (MMS) test with melting verifying the accuracy of the second order spatial discretisation. In Chapter 3 we benchmark our model against MITgcm demonstrating the model's capability to simulate a range of scenarios, using a variety of meshing strategies, including a full 3D simulation of an entire ice shelf cavity for the ISOMIP+ test case. This includes an investigation into the sensitivity of the melt parameterisation to vertical mixing and resolution choice. We present adjoint sensitivity calculations in an ice-ocean context in Chapter 4. Although the domains are still idealised, the simulations highlight the sensitivity to ocean conditions at the grounding zone and there is exciting future scope to apply this tool to future ice-ocean research. The final chapter applies the forward model to simulate basal crevasses extending previous work to 3D and considering the effects of rotation and crevasse geometry on the melt. The experiments

are motivated by observations of crevasses made at the Thwaites grounding zone by the MELT team as part of the Thwaites International Glacier Collaboration (ITGC) [Schmidt et al., 2023, Davis et al., 2023].

Publications as a result of work in this PhD Thesis

Condensed versions of the model description (Sections 2.1–2.8), MMS verification test with melting (Section 2.9), 3D ISOMIP+ comparison with MITgcm (Section 3.6) and preliminary adjoint sensitivity calculations from an idealised grounding zone (Section 4.3) from this Thesis have been published in the following paper:

> William I. Scott, Stephan C. Kramer, Paul R. Holland, Keith W. Nicholls, Martin J. Siegert, Matthew D. Piggott (2023). 'Towards a fully unstructured ocean model for ice shelf cavity environments: Model development and verification using the Firedrake finite element framework', Ocean Modelling, Volume 182, 102178, https://doi.org/10.1016/j.ocemod.2023.102178.

Where appropriate, please refer to the 'Publications and Disclaimer' section at the beginning of each chapter for more details on the overlap between the publication and the work in that chapter.

Chapter 2

Model Development and Verification

Highlights

- We implement a new discontinuous Galerkin (DG) finite element model for ocean flow under ice shelves in Firedrake.
- We discuss flexible meshing options, including new velocity-pressure tensor product elements pairs for large scale ocean applications in anisotropic domains.
- We present a novel MMS test that rigorously verifies our second order accurate spatial discretisation.

Summary

This chapter is primarily split into two parts. The first part summarises the model equations and finite element discretisation choices made as part of the model development. In particular we emphasise the DG aspects to help with documentation and reproducibility. We present a new discretisation suitable for ocean models with anisotropic domains based on tensor product elements and Firedrake's 'extruded' mesh functionality. The chapter also describes the

solver choices made during timestepping, the melt parameterisation and slope limiters, used to enhance model stability, especially regarding their impact on melt rates.

The second part of this chapter focuses on code verification. A particular highlight is a new Method of Manufactured Solution (MMS) test, that includes melting, rigorously showing that our spatial discretisation is sound and achieves the expected second order convergence.

Publications and Disclaimer

A condensed version of this chapter appears in Scott et al. [2023], though the model description and verification tests are expanded here. Throughout the model development effort I have worked closely with, and had a lot of help from, Stephan Kramer. The initial development of the Navier-Stokes solver was carried out by Stephan Kramer and the design choices and some functionality were strongly influenced by the Thetis project, a Firedrake based coastal ocean model developed in the Earth Science and Engineering (ESE) department at Imperial College London. I initially implemented the melt parameterisation as part of my MSci thesis, also in ESE. During my PhD project I have further developed the model in collaboration with Stephan Kramer and I have led on the testing, verification and development of our ability to model ice-ocean interactions.

2.1 Model Equations

As emphasised in Section 1, one of the key motivations is to develop a model with the ability to treat vertical dynamics in the same way as horizontal dynamics, when the mesh cells become close to isotropic, on fully unstructured meshes. This means we solve the full incompressible Navier-Stokes equations for velocity and pressure, and we do not make the hydrostatic approximation.

Conservation of momentum (in strong form) for a rotating fluid under the Boussinesq approx-

imation can be written as

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + 2\boldsymbol{\Omega} \times \mathbf{u} = -\frac{1}{\rho_0} \nabla p + \nabla \cdot \boldsymbol{\tau} - \frac{\rho'}{\rho_0} g\mathbf{k} + \mathbf{F}, \tag{2.1}$$

where

$$\boldsymbol{\tau} = \boldsymbol{\nu} \cdot \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right]. \tag{2.2}$$

The rotation vector of the Earth is given by $\boldsymbol{\Omega}$ and in the applications presented in this work Coriolis is implemented using an f-plane assumption, with $f = -1.409 \times 10^{-4}$ s^{-1}, corresponding to a latitude of 75 °S [Vallis, 2017]. ρ_0 is the reference density of sea water, taken as 1027.51 kg m^{-3} following Asay-Davis et al. [2016]. Kinematic eddy viscosity, $\boldsymbol{\nu}$, is a rank two tensor and can be spatially variant

$$\boldsymbol{\nu} = \begin{pmatrix} \nu_h(x,y,z) & 0 & 0 \\ 0 & \nu_h(x,y,z) & 0 \\ 0 & 0 & \nu_v(x,y,z) \end{pmatrix}, \tag{2.3}$$

where ν_h is the horizontal viscosity and ν_v is the vertical viscosity. $\mathbf{F}$ is a generic vector body force.

We make the Boussinesq approximation, where perturbations in density are neglected in all terms except the buoyancy term. $g = 9.81$ m s^{-2} is the gravitational acceleration and $\mathbf{k}$ is the unit vector pointing in the vertical direction. ρ' is a density perturbation related to temperature, T, and salinity, S, by a linear equation of state, given by

$$\rho' = \rho_0(\alpha_T(T - T_0) + \beta_S(S - S_0)), \tag{2.4}$$

where $\alpha_T = -3.733 \times 10^{-5}\,^\circ\text{C}^{-1}$ and $\beta_S = 7.843 \times 10^{-4}$ are respectively the expansion and contraction coefficients for temperature and salinity. $T_0 = -1\,^\circ$C and $S_0 = 34.2$ are reference temperature and salinity values used in the linear equation of state again with values from Asay-Davis et al. [2016]. Note that a constant hydrostatic pressure term has been subtracted from the momentum equations and incorporated into the definition of pressure, so that p is a

perturbation pressure

$$p = P - P_{\text{hyd}}, \tag{2.5}$$

where P is the full pressure and P_{hyd} is given by

$$P_{\text{hyd}} = -\rho_0 g z, \tag{2.6}$$

and z is the vertical coordinate.

The velocity, $\mathbf{u} = (u, v, w)^T$, and pressure, p, can be solved for using the continuity equation as a constraint to enforce the incompressibility condition

$$\nabla \cdot \mathbf{u} = 0. \tag{2.7}$$

In this work we have not implemented a free surface, i.e. through a kinematic boundary condition on the top surface and an evolution equation for the free surface elevation. The top surface of the model is treated as a rigid lid. Flows with more complicated tidal forcing may require a free surface to accurately capture the dynamics though and we anticipate implementing this feature in future work.

To complete the description of the problem boundary conditions need to be imposed, which are given as follows

$$\mathbf{u} = \mathbf{u}_D = u_n \mathbf{n} + u_t \mathbf{t} \qquad \text{on } \Gamma_D, \tag{2.8}$$

$$\mathbf{u} \cdot \mathbf{n} = u_n, \quad \tau_{nt} = f_{\text{tang}} \qquad \text{on } \Gamma_{D_n}, \tag{2.9}$$

$$-\rho_0 \tau_{nn} + p = p_{\text{ext}}, \quad \tau_{nt} = 0 \qquad \text{on } \Gamma_{p_{\text{ext}}}, \tag{2.10}$$

Unit normal and tangential vectors are given by $\mathbf{n}$ and $\mathbf{t}$ respectively. On Γ_D Dirichlet boundary conditions for both the normal and tangential components of velocity, u_n and u_t, are specified. This type of boundary can be used for rigid walls by setting $u_n = u_t = 0$ for no-normal, no-slip conditions. Although these boundaries are not used in this work they are included for completeness to ensure the correctness of the weak forms in Section 2.2. On Γ_{D_n} boundaries

only the normal component of velocity is specified and is combined with a condition for the shear stress in the tangential direction

$$\tau_{nt} = \mathbf{n} \cdot \boldsymbol{\tau} \cdot \mathbf{t}. \tag{2.11}$$

Free slip boundaries are imposed by setting f_{tang} and u_n to zero. On the top and bottom boundaries, namely the floating ice shelf and the seabed, we apply quadratic wall drag

$$f_{\text{tang}} = C_D|\mathbf{u}|\mathbf{u} \cdot \mathbf{t}, \tag{2.12}$$

where C_D is a drag coefficient. For open boundaries, $\Gamma_{p_{\text{ext}}}$, only an external pressure is specified in (2.10), with τ_{nn} the normal component of the stress vector on the boundary given by

$$\tau_{nn} = \mathbf{n} \cdot \boldsymbol{\tau} \cdot \mathbf{n}. \tag{2.13}$$

These types of boundaries are necessary in open domains, where an external hydrostatic pressure field is applied to balance the stress on either side of the open boundary, thus preventing fluid from 'falling' out of the domain.

Temperature and salinity are governed by scalar advection diffusion equations, which can be written in strong form as

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \nabla \cdot (\boldsymbol{\kappa}_T \cdot \nabla T) + r_T(T_{\text{res}} - T), \tag{2.14}$$

and

$$\frac{\partial S}{\partial t} + \mathbf{u} \cdot \nabla S = \nabla \cdot (\boldsymbol{\kappa}_S \cdot \nabla S) + r_S(S_{\text{res}} - S), \tag{2.15}$$

where $\boldsymbol{\kappa}_T$ and $\boldsymbol{\kappa}_S$ are the (eddy) diffusivity of temperature and salinity respectively. These are spatially variant rank two tensors similar to viscosity. r_T and r_S represent restoring frequencies when relaxing the solution in a sponge region to specified values, T_{res} and S_{res}.

Boundary conditions for temperature and salinity are given as

$$T = T_D \qquad \text{on } \Gamma_{D_T}, \tag{2.16}$$

$$(\boldsymbol{\kappa}_T \cdot \nabla T) \cdot \mathbf{n} = \Phi_T \qquad \text{on } \Gamma_{\Phi_T}, \tag{2.17}$$

$$S = S_D \qquad \text{on } \Gamma_{D_S}, \tag{2.18}$$

$$(\boldsymbol{\kappa}_S \cdot \nabla S) \cdot \mathbf{n} = \Phi_S \qquad \text{on } \Gamma_{\Phi_S}, \tag{2.19}$$

where T_D and S_D are the Dirichlet boundary values for temperature and salinity respectively on walls Γ_{D_T} and Γ_{D_S} given in (2.16) and (2.18). Neumann boundary fluxes for temperature (2.17) and salinity (2.19) are given by Φ_T and Φ_S respectively. Note the definition of Φ includes diffusivity, so if the tracer is temperature, the units of Φ_T would be m K/s.

The next section gives the weak forms of (2.1), (2.7), (2.14) and (2.15) necessary for the finite element discretisation.

2.2 Weak Form of Model Equations

2.2.1 Tracers

Choosing a suitable scalar function space, $\mathcal{Q}$, for a generic tracer, q, and a test function, ϕ, the weak form of the advection diffusion equation ((2.14) and (2.15)) can be written as

$$M_q + A_q + K = F_q - S, \quad \forall \phi \in \mathcal{Q}, \tag{2.20}$$

where

$$M_q = \int_\Omega \frac{\partial q}{\partial t} \phi \, dx, \tag{2.21}$$

$$A_q = \int_\Omega -q \nabla \cdot (\phi \mathbf{u}) \, dx + \int_\Gamma \phi \tilde{q} \mathbf{u} \cdot \mathbf{n} \, ds, \tag{2.22}$$

$$K = \int_\Omega \nabla\phi \cdot (\boldsymbol{\kappa} \cdot \nabla q)\, dx - \int_{\Gamma_{D_q}} \phi \mathbf{n} \cdot \boldsymbol{\kappa} \cdot \nabla q\, ds - \int_{\Gamma_{\Phi_q}} \phi \Phi_q\, ds, \tag{2.23}$$

$$F_q = \int_\Omega r_q q_{\text{res}} \phi\, dx, \tag{2.24}$$

$$S = \int_\Omega r_q q \phi\, dx, \tag{2.25}$$

where the individual terms are denoted: M_q – time derivative, A_q – advection, K – (eddy) diffusivity, F_q – source and S – sink. Dirichlet boundary values for the tracer are denoted q_D and Neumann boundary conditions as Φ_q. Note $\tilde{q}$ is the 'advected' scalar value defined in Section 2.4 in the context of surface integrals and Discontinuous Galerkin (DG) finite elements.

2.2.2 Velocity-Pressure

To derive the weak forms of (2.1) and (2.7) we choose separate, suitable function spaces $\boldsymbol{\mathcal{V}}$ and $\mathcal{W}$ for velocity, $\mathbf{u} = (u, v, w)^T$, and pressure, p respectively. Multiplying (2.1) by a vector test function $\boldsymbol{\phi} = (\phi_u, \phi_v, \phi_w)^T \in \boldsymbol{\mathcal{V}}$ and integrating over the domain, Ω, gives

$$M + A + C + P + V = B + F \qquad \forall \boldsymbol{\phi} \in \boldsymbol{\mathcal{V}}, \tag{2.26}$$

where

$$M = \int_\Omega \frac{\partial \mathbf{u}}{\partial t} \cdot \boldsymbol{\phi}\, dx, \tag{2.27}$$

$$A = \int_\Omega -\mathbf{u} \cdot [\nabla \cdot (\mathbf{u} \otimes \boldsymbol{\phi})]\, dx + \int_\Gamma (\tilde{\mathbf{u}} \cdot \boldsymbol{\phi})(\mathbf{u} \cdot \mathbf{n})\, ds, \tag{2.28}$$

$$C = \int_\Omega -fv\phi_u + fu\phi_v\, dx, \tag{2.29}$$

$$P = \int_\Omega \frac{1}{\rho_0} \boldsymbol{\phi} \cdot \nabla p\, dx - \int_{\Gamma_{p_{\text{ext}}}} \frac{1}{\rho_0} (p - p_{\text{ext}})(\boldsymbol{\phi} \cdot \mathbf{n})\, ds, \tag{2.30}$$

$$V = \int_\Omega \nabla \boldsymbol{\phi} \cdot \boldsymbol{\tau}\, dx - \int_{\Gamma_D \cup \Gamma_{D_n}} \boldsymbol{\phi} \cdot (\mathbf{n} \cdot \boldsymbol{\tau})\, ds \tag{2.31}$$
$$+ \int_{\Gamma_{D_n}} (\mathbf{n} \cdot \boldsymbol{\tau} \cdot \mathbf{t})(\boldsymbol{\phi} \cdot \mathbf{t})\, ds - \int_{\Gamma_{D_n}} f_{\text{tang}} \boldsymbol{\phi} \cdot \mathbf{t}\, ds,$$

$$B = \int_\Omega -\frac{\rho'}{\rho_0} g\mathbf{k} \cdot \boldsymbol{\phi}\, dx, \tag{2.32}$$

$$F = \int_\Omega \mathbf{F} \cdot \boldsymbol{\phi}\, dx, \tag{2.33}$$

the individual terms being: M – time derivative, A – advection, C – rotation, P – pressure gradient, V – (eddy) viscosity, B – buoyancy and F – source. Note $\tilde{\mathbf{u}}$, similar to $\tilde{q}$, is an 'advected' velocity also defined in Section 2.4. The symbol $\otimes$ represents an outer product between two vector quantities.

Multiplying (2.7) by a test function $\psi \in \mathcal{W}$ and integrating by parts gives the weak form of the continuity equation

$$\int_\Omega -\nabla\psi \cdot \mathbf{u}\, dx + \int_{\Gamma_{p_{\text{ext}}}} \psi \boldsymbol{u} \cdot \mathbf{n}\, ds + \int_{\Gamma_D \cup \Gamma_{D_n}} \psi u_n\, ds = 0, \qquad \forall \psi \in \mathcal{W}. \tag{2.34}$$

Defining ψ in the pressure space ensures that the discretised divergence and gradient operators are the negative transpose of each other and it means there are the same number of discrete continuity equations as pressure degrees of freedom. This is natural, since the pressure can be thought of as a Lagrange multiplier that enforces the incompressibility constraint [Strang, 2007].

2.3 Finite Element Discretisation: Fully Unstructured and 'Extruded' Meshes

2.3.1 Tracers

On fully unstructured meshes (2D: triangle elements, 3D: tetrahedral elements) we use linear discontinuous basis functions (P1DG) for the tracer test and trial space, $\mathcal{Q}$. This is because P1DG elements are well suited to advection dominated problems [Kärnä et al., 2018].

For high aspect ratio domains, where the horizontal extent of the domain is much greater than the depth, for instance when simulating an entire ice shelf cavity, an alternative option to using a fully unstructured mesh is to make use of Firedrake's in-built 'extruded' mesh feature. Extruded meshes offer increased performance and robustness when elements are highly anisotropic, yet retain the flexibility of a horizontally unstructured mesh [Bercea et al., 2016, McRae et al., 2016].

An extruded 2D vertical slice mesh is made up of quadrilateral elements, which can be thought of as the tensor product of an element interval in the horizontal and an element interval in the vertical. For 3D domains the elements are triangular prisms, which are formed by the tensor product of a triangular element in the horizontal with an interval element in the vertical. This means the horizontal part of the mesh can be fully unstructured. In the vertical direction the mesh is structured. It is possible to choose between a fixed number of columns in the vertical, or a fixed cell height, as well as 'shaving' cells in the top and bottom columns to accurately fit the domain geometry.

Finite elements on extruded meshes are defined in terms of a tensor product of a finite element on the horizontal mesh and finite element on the vertical mesh. This approach controls how the element behaves in the horizontal direction and how the element behaves in the vertical. This is particularly useful for more 'exotic' element combinations that have different continuity requirements in the horizontal and vertical directions.

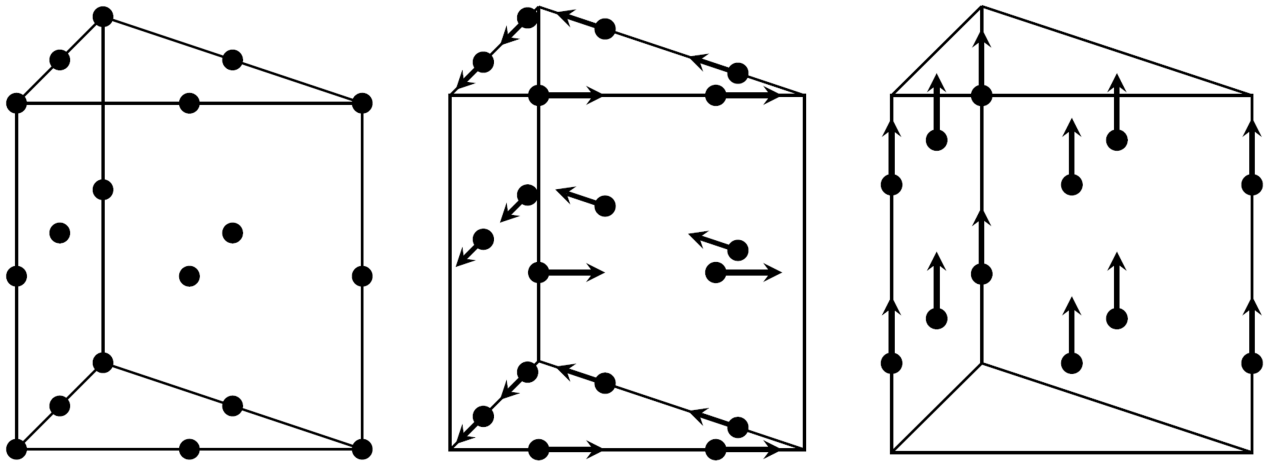


Figure 2.1: Schematic of 3D extruded elements used in Section 3.6. Degrees of Freedom (DOF) are represented by black dots (with arrows representing vector quantities). DOF situated on the edges of the element indicate continuity between elements, whereas DOF displaced into the element indicate discontinuity between elements. The extruded triangular prism elements are created by taking the product of a triangular element in the horizontal with an interval element in the vertical. The element on the left, used to discretise pressure, is a tensor product of P2 in the horizontal with P2 in the vertical. The element in the middle, representing the horizontal velocity component, is formed by taking a product of N2 in the horizontal with P2 in the vertical. This ensures the horizontal component of velocity is discontinuous in the normal direction but continuous in the tangential direction. The element on the right is the vertical velocity component formed from taking a product of P2 in the horizontal with P1DG in the vertical. This ensures that the vertical velocity component is discontinuous in the vertical direction but continuous in the horizontal direction.

On extruded meshes we choose the scalar finite element which mimics the P1DG element. This is formed by taking the tensor product of P1DG on the horizontal element with P1DG on the vertical interval.

2.3.2 Velocity-Pressure

The choice of a suitable finite element velocity-pressure element pair is crucial to ensuring an accurate and stable discretisation of the continuous equations [Elman et al., 2014]. On fully unstructured meshes we use (vector) P1DG basis functions for the velocity space, $\boldsymbol{\mathcal{V}}$, and quadratic continuous basis functions (P2) for the pressure space, $\mathcal{W}$. The resulting P1DG-P2 velocity-pressure element pair has been used with the finite element ocean model Fluidity [Piggott et al., 2008b, 2009, Kramer et al., 2010, Kimura et al., 2013, Jordan et al., 2014] and more recently implemented within the Thetis (coastally-focused) ocean model developed using

Firedrake [Kärnä et al., 2018, Wallwork et al., 2020]. The element pair has been shown to maintain geostrophic balance, because the skew gradient of the pressure maps to the velocity space [Cotter et al., 2009b,a, Cotter and Ham, 2011].

On extruded meshes we have chosen an element pair that has similar properties to P1DG-P2, so that as the horizontal mesh is refined the discretisation is able to deal with order one aspect ratios. Defining the equivalent of P2 on extruded meshes is relatively straightforward: it is the tensor product of a P2 horizontal element with the equivalent P2 element on the vertical interval, as shown on the left side of Figure 2.1.

Forming the required velocity element on a vertically extruded mesh is more complicated though. Considering an extruded 2D vertical slice mesh first for simplicity (i.e. a quadrilateral mesh), a possible equivalent discretisation choice for P1DG-P2 is DQ1-Q2, where Q indicates quadrilateral elements instead of triangles. However, this element pair is not LBB (Ladyzhenskaya–Babuška–Brezzi) stable (also referred to as the inf-sup condition). As the momentum and continuity equations form a saddle point system with positive and negative eigenvalues (see Section 2.5), if a discretisation is not LBB stable, then spurious zero-value eigenvalues are introduced that will produce grid scale noise that dominate the solution as the grid is refined, instead of producing a result that converges towards the solution of the continuous equation as the grid is refined. The equal order velocity-pressure pair, P1-P1, is an example of an unstable pair that does not meet the inf-sup condition [Strang, 2007]. Figure 2.2 show the pressure field for the DQ1-Q2 pair has an obvious checkerboard pattern that will get worse as the grid is refined compared with the stable velocity element pair described below that has no checkerboard pattern.

Our solver strategy, outlined in Section 2.5, relies on the gradient of functions in the pressure function space mapping to functions in the velocity trial space. Given our discretisation choice for pressure this places a constraint on the continuity of the velocity element. Along the element boundary the gradient of the pressure is continuous across cells in the tangential direction and varies quadratically. In the normal direction the pressure gradient is discontinuous between cells and varies linearly.

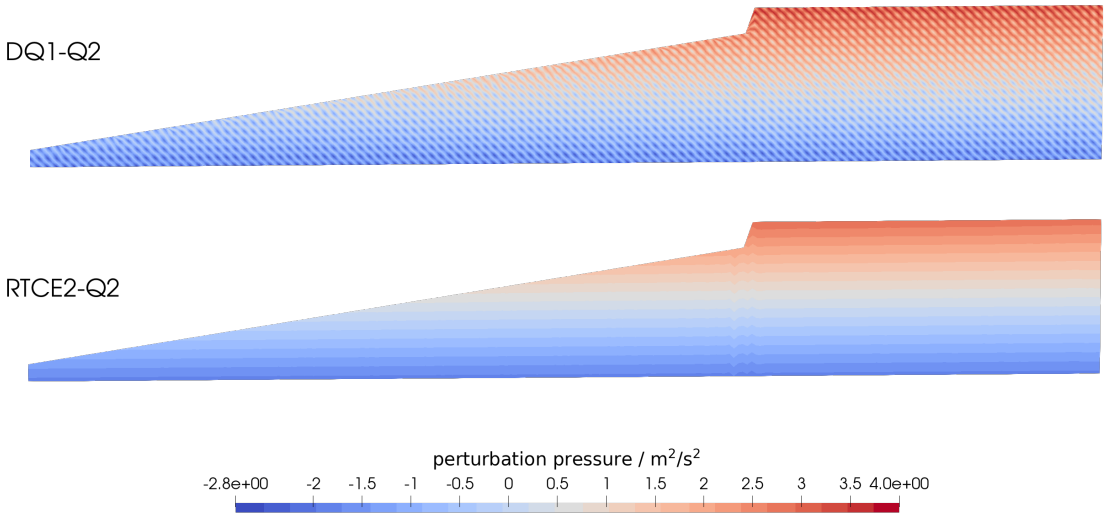


Figure 2.2: Pressure fields for a 2D early, pre-ISOMIP+ (see Section 3.6) simulation using the unstable velocity-pressure element pair DQ1-Q2 (top) and the stable RTCE2-Q2 pair (bottom). Note the obvious checkerboard pattern in the pressure field for the unstable element pair.

For a 2D vertical slice the horizontal component of this velocity element, u, is formed by taking the tensor product of P1DG on a horizontal interval and P2 on a vertical interval. This ensures that the horizontal component of velocity is discontinuous in the horizontal direction, but continuous between elements on vertical facets. The vertical component of the velocity, w, is defined the other way round, as a tensor product of P2 in the horizontal and P1DG in the vertical. Again, this ensures that the vertical velocity is continuous in the horizontal direction, but discontinuous in the vertical direction. This 2D vector element is actually referred to as RTCE2 using the FEM periodic table [Kirby et al., 2012] and is supported natively by Firedrake on quadrilateral meshes.

The 3D equivalent of the 2D velocity element is built up as follows: the horizontal velocity components, u and v, are formed by taking the tensor product of a Nedelec element of the second kind (N2) defined on a triangular element with a P2 element in the vertical. N2 elements are inherently (2D) vector elements, that are tangentially continuous between cells and discontinuous in the normal direction [Kirby et al., 2012]. The middle panel of Figure 2.1 shows

this element discretising the horizontal velocity components.

The vertical component of velocity is defined in the same way as on the 2D vertical slice mesh with a tensor product of P2 in the horizontal and P1DG in the vertical and is shown on the right hand side of Figure 2.1. Similarly Figure 2.5 from Section 2.6 detailing the melt parametisation shows a 2d vertical schematic representation of the P1DG discretisation on the fully unstructured mesh versus the tensor product discretisation on the vertically structured extruded mesh.

2.4 Discontinuous Galerkin Surface Integrals

Various terms in the weak form have been integrated by parts, resulting in additional surface integrals, which can be exploited to enforce boundary conditions weakly. With continuous finite element basis functions, contributions from either side of the same internal boundary between elements cancel. This means only surface integrals defined on exterior boundaries need to be considered. In the Discontinuous Galerkin (DG) method however, the surface contributions from either side of the internal boundaries do not cancel, because of the 'jump' in the solution value across the boundary.

For the advection term in (2.28) we choose the advected velocity, $\tilde{\mathbf{u}}$, to be the upwind value. On an exterior outflow boundary this simply means $\tilde{\mathbf{u}}$ is the solution inside the domain and on inflow boundaries, $\tilde{\mathbf{u}}$ is set using the Dirichlet values at the boundary. In the interior, the solution fields on either side of a facet between two adjacent elements are arbitrarily labelled with '+' and '-'. Integrals over interior facets are given by $\int_{\Gamma_{\text{int}}} dS$. Averages across interfaces are shown by $\{\{\cdot\}\}$ and $[[\cdot]]$ denotes a jump across an interface with unit normal, $\mathbf{n}$, which for scalar a and vector quantity $\mathbf{b}$ are given by

$$\{\{a\}\} = \frac{1}{2}(a^+ + a^-), \tag{2.35}$$

$$\{\{\mathbf{b}\}\} = \frac{1}{2}(\mathbf{b}^+ + \mathbf{b}^-), \tag{2.36}$$

$$[[a\mathbf{n}]] = a^+\mathbf{n}^+ + a^-\mathbf{n}^-, \tag{2.37}$$

$$[[\mathbf{b}\cdot\mathbf{n}]] = \mathbf{b}^+\cdot\mathbf{n}^+ + \mathbf{b}^-\cdot\mathbf{n}^-, \tag{2.38}$$

$$[[\mathbf{b}\otimes\mathbf{n}]] = \mathbf{b}^+\otimes\mathbf{n}^+ + \mathbf{b}^-\otimes\mathbf{n}^-. \tag{2.39}$$

Using this notation, the additional term for DG advection is

$$\int_{\Gamma_{\text{int}}} \tilde{\mathbf{u}}\cdot([[\boldsymbol{\phi}(\mathbf{u}\cdot\mathbf{n})]])\,dS, \tag{2.40}$$

where the upwind velocity, $\tilde{\mathbf{u}}$ is given by

$$\tilde{\mathbf{u}} = \mathbf{u}^+ \qquad \text{if } \mathbf{u}\cdot\mathbf{n}^- \leq 0, \tag{2.41}$$

$$\tilde{\mathbf{u}} = \mathbf{u}^- \qquad \text{if } \mathbf{u}\cdot\mathbf{n}^- > 0. \tag{2.42}$$

The discontinuous nature of the DG solution is evidently highly beneficial for advection dominated problems with sharp gradients in flow properties, such as modelling ocean fronts. However, the DG method requires some care when dealing with the discontinuity between cells for the diffusive terms in a consistent manner, even if these are not the dominant terms for ocean models. The viscosity term is discretised using the Symmetric Interior Penalty Galerkin (SIPG) method [Epshteyn and Rivière, 2007]. After integrating by parts the discontinuous velocity field leads to an interior facet term

$$-\int_{\Gamma_{\text{int}}} [[\boldsymbol{\phi}\otimes\mathbf{n}]]\cdot\{\{\boldsymbol{\nu}\cdot(\nabla\mathbf{u}+(\nabla\mathbf{u})^T)\}\}\,dS. \tag{2.43}$$

Adding only this term to (2.31) would not give a stable scheme. An example is shown by the top panel of Figure 2.3 where oscillating instabilities develop in the solution and it does not converge to the analytical solution as the grid is refined. Note that this is a spatial discretisation issue, not a timestepping instability similar to an advection based CFL limit. Diffusion based CFL limits (especially in the horizontal direction) are not normally an issue for ocean modelling given that the grid size is typically much larger than the distance a signal propagates by diffusion in

one timestep, i.e. $\Delta x \gg \sqrt{2\kappa\Delta t}$.

Instead, two additional terms are included which ensure that the form of the diffusivity term is symmetric positive definite given by

$$
\begin{aligned}
&- \int_{\Gamma_{\text{int}}} ([[\mathbf{u} \otimes \mathbf{n}]] + [[\mathbf{u} \otimes \mathbf{n}]]^T) \cdot \{\{\boldsymbol{\nu} \cdot \nabla \boldsymbol{\phi}\}\} \, dS \\
&+ \int_{\Gamma_{\text{int}}} \sigma [[\boldsymbol{\phi} \otimes \mathbf{n}]] \cdot (\{\{\boldsymbol{\nu}\}\} \cdot ([[\mathbf{u} \otimes \mathbf{n}]] + [[\mathbf{u} \otimes \mathbf{n}]]^T)) \, dS.
\end{aligned} \tag{2.44}
$$

The first term in (2.44) makes the scheme symmetric. The second term is a penalty term, penalizing jumps in the solution to ensure that the operator remains coercive which is required for stability. Equations 3.20 and 3.23 from Hillewaert [2013] give a minimal value for the penalty parameter, σ as

$$
\sigma = \alpha C(p) n_e \{\{A_f / V_e\}\}, \tag{2.45}
$$

where n_e is the number of facets for the element type, α is a constant (the theoretical minimum is 1 for stability, but after experimentation we found using a value of 2 improves robustness), A_f is the area of the facet, V_e is the volume of the element. Equation 3.7 from Hillewaert [2013] defines C as a constant that depends on polynomial degree, p, and element type such that

$$
\int_f u^2 \, dS \leq C A_f / V_e \int_e u^2 \, dx, \tag{2.46}
$$

is true for all polynomials u of degree p. Values for C are given in Table 3.1 from Hillewaert [2013]. The additional SIPG terms are also added on exterior surfaces

$$
\begin{aligned}
&- \int_{\Gamma_D} ((\mathbf{u} - \mathbf{u}_D) \otimes \mathbf{n} + ((\mathbf{u} - \mathbf{u}_D) \otimes \mathbf{n})^T) \cdot (\boldsymbol{\nu} \cdot \nabla \boldsymbol{\phi}) \, ds \\
&+ \int_{\Gamma_D} 2\sigma \boldsymbol{\phi} \otimes \mathbf{n} \cdot (\boldsymbol{\nu} \cdot ((\mathbf{u} - \mathbf{u}_D) \otimes \mathbf{n} + ((\mathbf{u} - \mathbf{u}_D) \otimes \mathbf{n})^T)) \, ds.
\end{aligned} \tag{2.47}
$$

The bottom panel of Figure 2.3 shows that with the sufficient penalty term the SIPG scheme converges to the analytical solution as the grid is refined.

Scalar advection is treated similarly to momentum, with the additional interior facet term, due

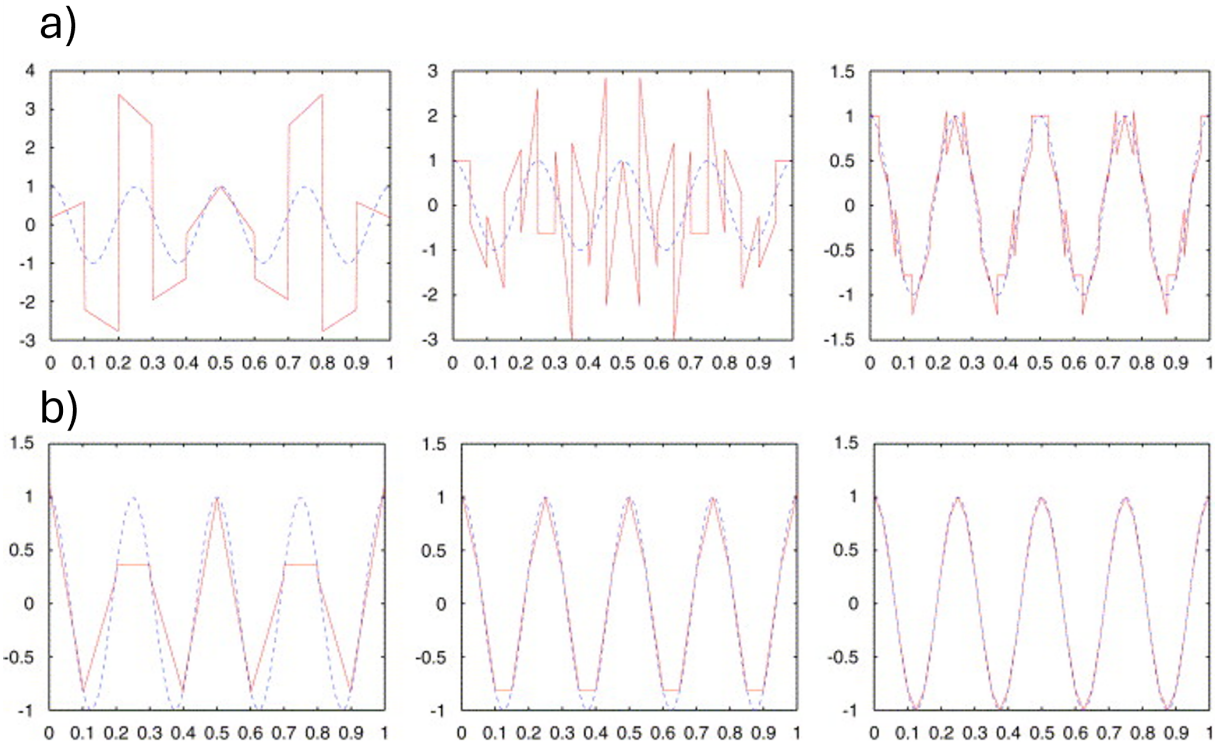


Figure 2.3: Effect of penalty parameter for 1D test case for three resolutions. The solid lines show the numerical results and the dashed line is the analytical solution. The top panel (a) uses a penalty value that is smaller than the critical penalty and oscillations occur for all three meshes giving a large numerical error. The bottom panel (b) shows that with a sufficient value for the penalty parameter, as the grid is refined the numerical solution converges towards the analytical solution. This figure is adapted from Figures 3 and 4 from Epshteyn and Rivière [2007] and the test parameters can be found within.

to the discontinuous finite element space, given by

$$\int_{\Gamma_{\text{int}}} [[\phi \mathbf{u} \cdot \mathbf{n}]] \tilde{q} \, dS, \tag{2.48}$$

where the upwind value of the tracer, $\tilde{q}$, is defined as

$$\tilde{q} = q^+ \qquad \text{if } \mathbf{u} \cdot \mathbf{n}^- \leq 0, \tag{2.49}$$

$$\tilde{q} = q^- \qquad \text{if } \mathbf{u} \cdot \mathbf{n}^- > 0. \tag{2.50}$$

On Dirichlet boundaries $\tilde{q}$ is again only replaced if the the flow is into the domain. Scalar diffusion in (2.23) is also implemented using the SIPG formulation, where the additional terms are given by

$$\begin{aligned} -\int_{\Gamma_{\text{int}}} [[\phi \mathbf{n}]] \cdot \{\{\boldsymbol{\kappa} \cdot \nabla q\}\} \, dS - \int_{\Gamma_{\text{int}}} [[q \mathbf{n}]] \cdot \{\{\boldsymbol{\kappa} \cdot \nabla \phi\}\} \, dS \\ + \int_{\Gamma_{\text{int}}} \sigma [[\phi \mathbf{n}]] \cdot (\{\{\boldsymbol{\kappa}\}\} \cdot [[q \mathbf{n}]]) \, dS \\ - \int_{\Gamma_{D_q}} (q - q_D) \mathbf{n} \cdot \boldsymbol{\kappa} \cdot \nabla \phi \, ds + \int_{\Gamma_{D_q}} 2\sigma \phi \mathbf{n} \cdot \boldsymbol{\kappa} \cdot (q - q_D) \mathbf{n} \, ds, \end{aligned} \tag{2.51}$$

and similar interpretations of each term can be made as for momentum viscosity in (2.31).

During the model development the penalty term was frequently the cause of a number of instability and solver convergence issues. For example, in anticipation of running 3D simulations the prediction solve (see Section 2.5) was changed from using a direct (MUMPS) to an iterative (GMRES) solver for both run time and memory savings. Using direct solvers was not feasible for 3D ISOMIP+ simulations (see Section 3.6) due to memory constraints. Whilst testing the iterative solver on 2D extruded meshes, the number of iterations required for the velocity prediction solve was very sensitive to changes to the penalty term, as shown by Table 2.1. For example, changing the penalty factor from 10 to 50 (original default), doubled the number of velocity iterations from 54 to 105 (using the RTCE2-Q2 element with viscosity of 1×10^{-3} m^2/s), whilst having no qualitative effect on the solution. The iteration counts also depended significantly on the value of vertical viscosity, especially for the σ meshes as shown by Table 2.2.

The original penalty term we used was based on a formulation by Epshteyn and Rivière [2007] that was intended to be used on simplices (triangles or tetrahedrals), not on quadrilateral meshes. As a result, a new penalty term formulation was implemented (Stephan Kramer, personal communication) with a penalty factor that took into account the type of mesh element based on the thesis by Hillewaert [2013] as described above. Tables 2.3 and 2.4 show iteration counts for both z and σ meshes after the new penalty formulation had dropped by nearly a factor of ten, requiring only 8 iterations for the velocity solve to converge. In addition, the new penalty formulation solved an existing problem, where stripes on the order of the grid scale appeared for the nominally stable RTCE2-Q2 velocity element pair using the old formulation, but disappeared after using the new scheme as shown in Figure 2.4. Incidentally, the LBB unstable DQ1-Q2 pair, which had been robust in practice (ignoring the pressure checkerboard error referred to earlier), became unstable with no simulation reaching 10 days before blowing up. Note that by 100 days the simulations were spun up and effectively stable indefinitely.

The penalty term also caused initial problems for convergence in 3D. The routine that calculates the constant, $C(p)$, as a function of polynomial degree, p, also requires the degree of the gradient field. The default setting was to subtract off one degree from p, as this is the degree of the gradient field for polynomials on triangles. For the tensor product element in 3D used for the ISOMIP+ study (see Section 3.6) this approach led to the degree of the gradient field being too low.

The horizontal velocity for the tensor product element is a product of N2 in the horizontal (degree of 1, since this vector is a really a subset of $P1^2$) and P2 in the vertical. This should make the gradient of the horizontal velocity, in the horizontal direction P0 $\times$ P2, i.e. a degree of 2. Similarly, the vertical gradient of the horizontal field should be N2 $\times$ P1, which gives a degree of 2.

The vertical velocity field is a product of P2 $\times$ P1DG. The horizontal gradient of the vertical velocity becomes P1 $\times$ P1DG, i.e. degree of 2 and likewise the vertical gradient of the vertical velocity field is P2 $\times$ P0, again giving a degree of 2.

Therefore, subtracting a degree from the default led to the penalty parameter being too low

Velocity FS	IP	ν_v	$\underline{\mathbf{u}}^*$	p	q	stable to 100 days ?
DQ1	50	0.01	77	28	25	yes
DQ1	50	0.001	69	30	24	yes
DQ1	50	0.0001	46	29	15	no
DQ1	10	0.01	40	27	15	yes
DQ1	10	0.001	35	30	13	yes
DQ1	10	0.0001	26	29	11	no
RTCE2	50	0.01	83	27	26	yes
RTCE2	50	0.001	105	27	24	yes
RTCE2	50	0.0001	87	29	15	no
RTCE2	10	0.01	64	27	15	yes
RTCE2	10	0.001	54	29	13	yes
RTCE2	10	0.0001	46	29	11	-

Table 2.1: Iteration counts for 2D z-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing. Column 4 shows the iteration counts when solving for the intermediate velocity, $\underline{\mathbf{u}}^*$, during the momentum prediction solve (Equation 2.68), column 5 the number of iterations for pressure, $\underline{p}$, as part of the pressure correction step (Equation 2.70) and column 6 shows the counts when solving for the tracer degrees of freedom, $\underline{q}$. There are two choices of velocity function space (FS), two values of the Interior penalty (IP) parameter using the old formulation and three values of the vertical viscosity, ν_v. N.b. horizontal viscosity is kept fixed at 6 m^2/s. The final column indicates whether the solution robustly ran past 100 days, as beyond this stage the dynamics were spun up from a steady state.

with the new formulation in 3D. By itself this was not enough, as the penalty safety factor had to be set to at least 2.5 in order to achieve convergence. The number of iterations for the momentum solve in the first 20 steps was still quite sensitive to the IP factor in 3D. With a penalty term of 3 there were 20 fewer iterations required ($\sim$50) compared with a penalty term of 4 ($\sim$70).

The above change was accidentally hard coded for all element types and for the MMS test case in 2D on triangle elements (see Section 2.9) this changed needed to be reversed to obtain better convergence for the pressure. Otherwise, there was a trade off between reducing the magnitude of viscosity to improve the pressure convergence at the expense of the velocity convergence. The main point to underline is that some care had to be taken with the magnitude of the penalty term depending on the type of mesh and choice of velocity element being used.

Velocity FS	IP	ν_v	$\underline{\mathbf{u}}^*$	$\underline{p}$	$\underline{q}$	stable to 100 days ?
DQ1	50	0.01	128	31	46	yes
DQ1	50	0.001	77	32	33	yes
DQ1	50	0.0001	148	29	30	-
DQ1	10	0.01	55	30	28	yes
DQ1	10	0.001	42	32	21	yes
DQ1	10	0.0001	58	29	17	-
RTCE2	50	0.01	85	43	26	yes
RTCE2	50	0.001	108	26	24	yes
RTCE2	50	0.0001	292	26	15	no
RTCE2	10	0.01	73	27	15	yes
RTCE2	10	0.001	53	17	13	yes
RTCE2	10	0.0001	79	15	10	-

Table 2.2: Iteration counts for 2D σ-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing. Column 4 shows the iteration counts when solving for the intermediate velocity, $\underline{\mathbf{u}}^*$, during the momentum prediction solve (Equation 2.68), column 5 the number of iterations for pressure, $\underline{p}$, as part of the pressure correction step (Equation 2.70) and column 6 shows the counts when solving for the tracer degrees of freedom, $\underline{q}$. There are two choices of velocity function space (FS), two values of the Interior penalty (IP) parameter using the old formulation and three values of the vertical viscosity, ν_v. N.b. horizontal viscosity is kept fixed at 6 m^2/s. The final column indicates whether the solution robustly ran past 100 days, as beyond this stage the dynamics were spun up from a steady state.

Velocity FS	ν_v	$\underline{\mathbf{u}}^*$	$\underline{p}$	$\underline{q}$	stable to 100 days ?
DQ1	0.01	8	26	4	no
DQ1	0.001	-	-	-	no **
DQ1	0.0001	-	-	-	no **
RTCE2	0.01	10	24	4	yes
RTCE2	0.001	8	29	3	no $^+$
RTCE2	0.0001	8	30	3	no

Table 2.3: Iteration counts for 2D z-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing using the new penalty parameter formulation, as described in Section 2.4. Column 3 shows the iteration counts when solving for the intermediate velocity, $\underline{\mathbf{u}}^*$, during the momentum prediction solve (Equation 2.68), column 4 the number of iterations for pressure, $\underline{p}$, as part of the pressure correction step (Equation 2.70) and column 5 shows the counts when solving for the tracer degrees of freedom, $\underline{q}$. There are two choices of velocity function space (FS) and three values of the vertical viscosity, ν_v. N.b. horizontal viscosity is kept fixed at 6 m^2/s. The final column indicates whether the solution robustly ran past 100 days, as beyond this stage the dynamics were spun up from a steady state. ** means the simulation diverged before 10 days. $^+$ Although the simulation diverged at 23 days this turned out to be due to an issue with the RHS boundary conditions. Applying stably stratified boundary conditions as per the ISOMIP+ protocol (see Section 3.6) enabled the simulation with $\nu_v = 1\times10^{-3}$ m^2/s to work.

Velocity FS	ν_v	$\underline{\mathbf{u}}^*$	$\underline{p}$	$\underline{q}$	stable to 100 days ?
DQ1	0.01	11	26	5	no
DQ1	0.001	-	-	-	no **
DQ1	0.0001	-	-	-	no **
RTCE2	0.01	9	30	4	yes
RTCE2	0.001	7	29	3	no $^+$
RTCE2	0.0001	-	-	-	no **

Table 2.4: Iteration counts for 2D σ-layer extruded meshes, as part of early, pre-ISOMIP+ (see Section 3.6) testing using the new penalty parameter formulation, as described in Section 2.4. Column 3 shows the iteration counts when solving for the intermediate velocity, $\underline{\mathbf{u}}^*$, during the momentum prediction solve (Equation 2.68), column 4 the number of iterations for pressure, $\underline{p}$, as part of the pressure correction step (Equation 2.70) and column 5 shows the counts when solving for the tracer degrees of freedom, $\underline{q}$. There are two choices of velocity function space (FS) and three values of the vertical viscosity, ν_v. N.b. horizontal viscosity is kept fixed at 6 m^2/s. The final column indicates whether the solution robustly ran past 100 days, as beyond this stage the dynamics were spun up from a steady state. ** means the simulation diverged before 10 days. $^+$ Although the simulation diverged at 23 days this turned out to be due to an issue with the RHS boundary conditions. Applying stably stratified boundary conditions as per the ISOMIP+ protocol (see Section 3.6) enabled the simulation with $\nu_v = 1\times10^{-3}$ m^2/s to work.

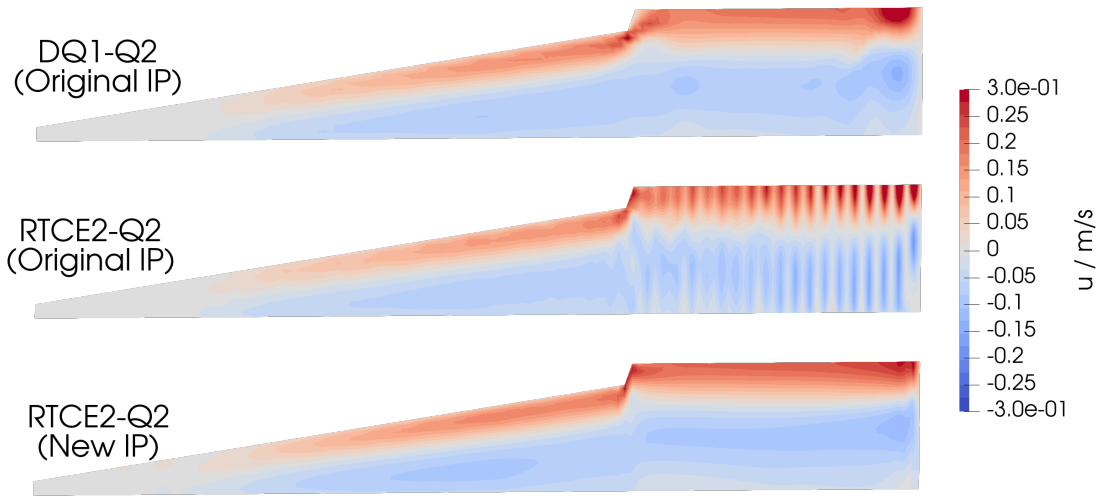


Figure 2.4: Horizontal velocity fields at 50 days for a 2D early, pre-ISOMIP+ (see Section 3.6) simulation using the unstable velocity-pressure element pair DQ1Q2 (top), the stable RTCE2-Q2 pair before changes to the Interior Penalty (IP) term (middle), and RTCE2-Q2 pair after changes to the IP term (bottom).

2.5 Timestepping and Solver Strategy

As outlined in the previous section, (2.26), (2.34) and (2.20), are the conservation of momentum, the incompressibility constraint and advection diffusion equations, and together they form a coupled system that needs to be solved for velocity, pressure, temperature and salinity. As stated before we are solving for the full nonhydrostatic dynamics and we do not carry out any mode splitting to separate the barotropic and baroclinic components of the flow.

Restricting the velocity test and trial functions to a finite dimensional subspace of size N and rewriting the continuous momentum equation, Equation 2.26, as a discrete matrix-vector system gives

$$\mathrm{M}\frac{\mathrm{d}\underline{\mathbf{u}}}{\mathrm{d}t} + \mathrm{A}(\mathbf{u})\underline{\mathbf{u}} + \mathrm{Cor}\ \underline{\mathbf{u}} + \mathrm{K}\underline{\mathbf{u}} + \mathrm{G}\underline{p} = \underline{\mathbf{B}} + \underline{\mathbf{F}} \qquad \forall \boldsymbol{\phi} \in \boldsymbol{\mathcal{V}}^N, \tag{2.52}$$

where

$$\mathrm{M}_{ij} = \int_\Omega \boldsymbol{\phi}_i \cdot \boldsymbol{\phi}_j, \tag{2.53}$$

$$\mathrm{A}_{ij} = \int_\Omega -\nabla \cdot (\boldsymbol{\phi}_i \mathbf{u}_{adv}) \cdot \boldsymbol{\phi}_j, \tag{2.54}$$

$$\mathrm{Cor}_{ij} = \int_\Omega \boldsymbol{\phi}_i \cdot f\mathbf{k} \times \boldsymbol{\phi}_j, \tag{2.55}$$

$$\mathrm{K}_{ij} = \int_\Omega \nabla\boldsymbol{\phi}_i \cdot (\nabla\boldsymbol{\phi}_j + (\nabla\boldsymbol{\phi}_j)^T), \tag{2.56}$$

$$\mathrm{G}_{ij} = \int_\Omega \boldsymbol{\phi}_i \cdot \nabla\psi_j, \tag{2.57}$$

$$\mathbf{B}_i = \int_\Omega -\boldsymbol{\phi}_i \cdot \frac{\rho'}{\rho_0} g\mathbf{k}, \tag{2.58}$$

$$\mathbf{F}_i = \int_\Omega \boldsymbol{\phi}_i \cdot \mathbf{F}, \tag{2.59}$$

and likewise, restricting the pressure trial basis functions and the test functions for the discretised continuity equation to a finite dimensional subspace $\mathcal{W}$ of size M gives

$$\mathrm{G}^T\underline{\mathbf{u}} = \mathbf{0} \qquad \forall \psi \in \mathcal{W}^M, \tag{2.60}$$

where

$$\mathrm{G}_{ij}^{T} = \int_{\Omega} \nabla\psi_i \cdot \boldsymbol{\phi}_j. \tag{2.61}$$

$\underline{\mathbf{u}}$ is a vector which stores the solution (degrees of freedom) for velocity associated with each trial basis function

$$\mathbf{u} = (u, v, w)^T, \tag{2.62}$$

$$u(\mathbf{x}) = \sum_j u_j \phi_j(\mathbf{x}), \tag{2.63}$$

$$v(\mathbf{x}) = \sum_j v_j \phi_j(\mathbf{x}), \tag{2.64}$$

$$w(\mathbf{x}) = \sum_j w_j \phi_j(\mathbf{x}), \tag{2.65}$$

where the summation is over the number of basis functions, N, and the coefficients u_j, v_j and w_j represent the degrees of freedom associated with each velocity component for basis function ϕ_j. Similarly $\underline{p}$ is the vector containing the pressure degrees of freedom associated with the each trial basis function

$$p(\mathbf{x}) = \sum_j p_j \psi_j(\mathbf{x}). \tag{2.66}$$

Note that boundary conditions and the interior penalty terms have been omitted from the discretised form of equations for brevity. It is worth noting that the discrete pressure gradient operator G is the transpose of the divergence operator in the continuity equation due to the choice of surface integrals for the continuous weak form.

We have implemented a θ-timestepping scheme for the momentum equations. θ is a parameter which can take a value between zero and one. A choice of $\theta = 1$ leads to a fully implicit scheme, i.e. Backward Euler, $\theta = 0$ leads to fully explicit scheme, i.e., Forward Euler and $\theta = 1/2$ leads to the implicit midpoint scheme, i.e., Crank-Nicholson. For Backward Euler the system of equations written in block form is

$$\begin{pmatrix} \mathrm{M} + \Delta t\mathrm{A} + \Delta t\mathrm{Cor} + \Delta t\mathrm{K} & \Delta t\mathrm{G} \\ \mathrm{G}^T & 0 \end{pmatrix} \begin{pmatrix} \underline{\mathbf{u}}^{n+1} \\ \underline{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathrm{M}\underline{\mathbf{u}}^{n} + \Delta t\underline{\mathbf{B}} + \Delta t\underline{\mathbf{F}} \\ 0 \end{pmatrix}, \tag{2.67}$$

where Δt is the timestep, the superscript n refers to the current timestep and $n+1$ to the unknown value at the new timestep.

Time dependent finite element models always require an inversion of a mass matrix, M, so it is common to use implicit timestepping techniques. Many ocean models use explicit timestepping, which do not require solving a linear system at each timestep. This means that the each timestep is quick to compute. For stiff problems, which are typical of ocean dynamics, explicit solvers may be forced to use very short time steps for reasons of stability, rather than accuracy. Finite element models which use implicit time stepping can be competitive by taking longer timesteps, without compromising on accuracy.

For small 2D problems a direct solver can be used on the full coupled system, i.e., to solve for velocity and pressure at the same time. However, direct methods do not scale well as the problem size increases, particularly for 3D problems, in terms of both computational cost and memory requirements [Trefethen and III, 1997]. Instead, iterative methods must be used. Choosing effective preconditioners is crucial to ensure that iterative solvers lead to optimal performance. It is possible to apply iterative methods to the full coupled system but choosing appropriate preconditioners is challenging [Elman et al., 2014]. We solve the equations by the established pressure projection approach [Chorin, 1967], which splits the solve into two parts. There is an initial prediction step to find an intermediate velocity, $\underline{\mathbf{u}}^*$, where the momentum equation is solved using an implicit Backward Euler timestep for all the terms in the momentum equation, except the pressure, p^n, which uses the value from the previous timestep. The nonlinear advection and drag terms are linearised by setting the advecting velocity to the velocity value from the previous time step. The prediction solve is given by

$$\left(\mathrm{M} + \Delta t\mathrm{A} + \Delta t\mathrm{Cor} + \Delta t\mathrm{K}\right)\left(\underline{\mathbf{u}}^*\right) = \left(\mathrm{M}\underline{\mathbf{u}}^n + \Delta t\underline{\mathbf{B}} + \Delta t\underline{\mathbf{F}} - \Delta t\mathrm{G}\underline{p}^n\right), \tag{2.68}$$

This intermediate velocity is not divergence-free, however, so a correction step enforces the incompressibility constraint. This finds the updated pressure, $\underline{p}^{n+1}$, needed to obtain a divergence-

free velocity field $\underline{\mathbf{u}}^{n+1}$. This correction step takes the form of a time-splitting step

$$\underline{\mathbf{u}}^{n+1} = \underline{\mathbf{u}}^{*} + \Delta t \nabla \left(\underline{p}^{n+1} - \underline{p}^{n} \right), \tag{2.69}$$

that now has a simpler block discretisation than Equation 2.67, given by

$$\begin{pmatrix} \mathrm{M} & \Delta t \mathrm{G} \\ \mathrm{G}^{T} & 0 \end{pmatrix} \begin{pmatrix} \underline{\mathbf{u}}^{n+1} \\ \underline{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathrm{M}\underline{\mathbf{u}}^{*} + \Delta t \mathrm{G}\underline{p}^{n} \\ 0 \end{pmatrix}. \tag{2.70}$$

Firedrake provides an interface to choose solver options to pass to PETSc, the package which implements the solver routines [Balay et al., 1997]. Typically, we choose the GMRES Krylov subspace method to solve the momentum equation during the prediction step, because the matrix is not symmetric. After initial testing we found BoomerAMG, an algebraic multigrid preconditioner from the HYPRE suite [Henson and Yang, 2002], to be more robust for our applications than PETSc's default Geometric Algebraic Multigrid (GAMG) preconditioner. The block structure of the correction solve is simpler than the full coupled system because the matrix M in the top left-hand block is only a mass matrix (resulting from the time derivative term in (2.53)), compared with the equivalent matrix in the full system which would contain contributions from all the other terms in the momentum equation.

Although it is possible to apply preconditioners to this simplified block system and solve for velocity and pressure together, our solution strategy relies on solving for the pressure first. We carry out a Schur complement of factorisation Equation 2.70, which is an elimination procedure so that pressure can be solved for independently of velocity. This gives

$$\begin{pmatrix} \mathrm{M} & \Delta t \mathrm{G} \\ 0 & -\mathrm{G}^{T}\mathrm{M}^{-1}\mathrm{G} \end{pmatrix} \begin{pmatrix} \underline{\mathbf{u}}^{n+1} \\ \underline{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathrm{M}\underline{\mathbf{u}}^{*} + \Delta t \mathrm{G}\underline{p}^{n} \\ -\mathrm{G}^{T}\mathrm{M}^{-1}(\mathrm{M}\underline{\mathbf{u}}^{*} + \Delta t \mathrm{G}\underline{p}^{n}) \end{pmatrix}. \tag{2.71}$$

As an aside the structure of the system clearly shows that it is a saddle point matrix. M, the mass matrix, is symmetric positive definite (SPD), hence it has all positive eigenvalues. Whereas $-\mathrm{G}^{T}\mathrm{M}^{-1}\mathrm{G}$ is negative definite (because $\mathrm{G}^{T}\mathrm{M}^{-1}\mathrm{G}$ is SPD) and hence has all nega-

tive eigenvalues. The full matrix therefore has both positive and negative eigenvalues and is indefinite. This is the definition of a saddle point structure [Elman et al., 2014].

The second row of Equation 2.71 can be solved independently to find the pressure, $\underline{p}^{n+1}$, by inverting the matrix, $-\mathrm{G}^T\mathrm{M}^{-1}\mathrm{G}$. Since $-\mathrm{G}^T\mathrm{M}^{-1}\mathrm{G}$ is symmetric and can be made to be positive definite, the Conjugate Gradient (CG) algorithm would be the optimal Krylov subspace method [Trefethen and III, 1997]. However, the M^{-1} matrix is problematic because for most sparse matrices M, explicitly constructing the inverse is out of the question, as the inverse would be a dense matrix. The alternative is to invert M (i.e. solve a $\mathrm{M}\underline{y} = \underline{b}$ system) each time the action of $-\mathrm{G}^T\mathrm{M}^{-1}\mathrm{G}$ is required for each iteration of the CG algorithm. This is also not ideal, however, because even inverting a simple mass matrix at each iteration of the CG algorithm can become expensive.

However, because of the careful choice of velocity-pressure finite element function spaces (see section 2.3.2), where the gradient of pressure maps pointwise exactly into the velocity function space, it can be shown that the Schur complement of this system is exactly equivalent to a standard Continuous Galerkin discretisation of the pressure Poisson equation [Cotter and Ham, 2011], i.e. we can neglect the inverse of the mass matrix, instead using $-\mathrm{G}^T\mathrm{G}$.

Note for the P1DG-P2 discretisation, because the velocity is discretised with discontinuous elements, each term of M is only associated with its own cell so the mass matrix forms a block diagonal structure. It is therefore possible, and fast, to find the inverse of M. We make use of the Slate package [Gibson et al., 2020], which implements local tensor operations in Firedrake to carry out this inversion locally on each cell. This means we explicitly assemble $-\mathrm{G}^T\mathrm{M}^{-1}\mathrm{G}$ for the P1DG-P2 discretisation, but not for the tensor product discretisation on 'extruded' meshes, instead relying on the Continuous Galerkin discretisation of the pressure Poisson equation.

This Poisson equation is solved using the Conjugate Gradient (CG) algorithm, preconditioned with an algebraic multigrid. The efficiency of this solution strategy is highly dependent on the condition number. For an anisotropic domain, this is proportional to the square of the aspect ratio of the grid cells [Kramer et al., 2010]. Thus, as the aspect ratio goes to infinity, i.e. the horizontal length scales are much longer than the vertical, the condition number can

become unbounded, with negative implications for the convergence of the solve. However, by preconditioning the CG algorithm with a preconditioner that approximates the solution to the depth averaged equations, convergence can be made independent of the aspect ratio. This so called vertical lumping approach [Kramer et al., 2010] is similar to the idea first used in MITgcm and ensures that when the dynamics are close to a hydrostatic state, the solve is as fast as when only solving the hydrostatic equations [Marshall et al., 1997].

For parallel efficiency, the method does require cells to be aligned in columns in the vertical direction. This can be achieved using Firedrake's in-built extruded mesh feature in combination with the tensor product discretisation discussed in Section 2.3.1. We have found that when running 3D simulations on structured tetrahedral meshes, on aspect ratios larger than 10^2 (e.g., $\Delta x = \Delta y = 2\text{km}$, $\Delta z < 20\text{m}$), using the generic algebraic multigrid preconditioner GAMG, the wall clock time needed to solve one timestep of the model was dominated by the pressure correction solve. However, on extruded meshes with the vertical lumping preconditioner, the pressure correction solve is faster than the momentum solve for the intermediate velocity. Tables 2.1-2.4 also show that in 2D the pressure solve iterations was constant of about 30 across the σ and z-layer meshes, even with the squashing near the grounding zone for the σ where the aspect ratio approaches 10^3. Before in 2D without the vertical lumping the pressure correction solve required $\sim$600 iterations, so that led to a factor of 20 speedup. After solving for velocity and pressure we solve for temperature and salinity individually. We use a 3-stage Diagonally Implicit Runge-Kutta (DIRK) timestepping method [Ascher et al., 1995] for the temperature and salinity advection diffusion equations (2.20). The method is third order accurate. It is also an L-stable scheme, so it is suitable for stiff problems. We apply a vertex-based slope limiter after each timestep to prevent the solution from becoming unbounded [Kuzmin, 2010]. We have found that using a slope limiter is only necessary in the region of the ice front transition, which tends to have the fastest flow as the buoyant melt water plume accelerates up the ice front. Without the limiter a spurious freezing signal can occur, confined to the final grid cell before the ice front transition. With the limiter, however, freezing does not take place. Section 2.7 explores this in more detail.

We expect our P1DG finite element based method with implicit time stepping to be about an

order of magnitude slower than a conventional low order (possibly explicit) finite volume model for the same model domain and grid resolution. The ability to take longer timesteps can offset this. More importantly, by using a fully flexible meshing strategy we are able to represent small scale features in a large domain that would be impossible to represent with structured grids, typical of other ocean models. We have run Firedrake simulations of ocean circulation beneath ice shelves in parallel with up to 128 cores. Section 3.7 at the end of Chapter 3 gives a timing comparison between Firedrake and MITgcm for the runs in that Chapter. The reader is referred to the main Firedrake paper for a more in depth study of performance [Rathgeber et al., 2016]. Future work will investigate the performance of this ocean model when it is applied to more demanding, geometrically complex 3D domains that require a fully flexible unstructured mesh.

2.6 Melt Parameterisation

Melt rates are calculated based on conservation of heat and salt at the ice-ocean boundary and under the assumption that the boundary is at the freezing point, commonly referred to as the 'three-equation' melt parameterisation [Hellmer and Olbers, 1989, Holland and Jenkins, 1999, Dinniman et al., 2016]. Conservation of heat is expressed as

$$Q_i^T - Q_w^T = Q_{\text{latent}}^T, \tag{2.72}$$

where the latent heat term, Q_{latent}^T is given by

$$Q_{\text{latent}}^T = -\rho_0 w_b L_f, \tag{2.73}$$

with w_b the unknown melt rate (melting: $w_b > 0$, freezing: $w_b < 0$) and $L_f = 3.34 \times 10^5$ J kg^{-1} is the latent heat of fusion. The conductive heat flux into the ice, Q_i^T is

$$Q_i^T = -\rho_i c_{pi} w_b (T_i - T_b), \tag{2.74}$$

where $\rho_i = 920\,\mathrm{kg\ m^{-3}}$ is the density of ice, $c_{pi} = 2000\,\mathrm{J\ kg^{-1}\,K^{-1}}$ is the specific heat capacity of ice, T_i is the far field ice temperature, taken to be $-20\,^\circ\mathrm{C}$ and T_b is the unknown temperature at the ice-ocean boundary. The turbulent heat flux from the ocean to the ice, Q_w^T is given by

$$Q_w^T = -\rho_0 c_p \gamma_T (T_b - T_w), \tag{2.75}$$

where $c_p = 3974\,\mathrm{J\ kg^{-1}\,K^{-1}}$ is the specific heat capacity of water and γ_T the turbulent thermal exchange velocity. Commonly γ_T is expressed as $\Gamma_T u^\star$, where Γ_T is a dimensionless constant and $u^\star = \sqrt{C_D}|u|$ is the friction velocity [Asay-Davis et al., 2016]. We use the velocity at the top surface of the domain to calculate $u^\star$, in effect making the assumption that our computational boundary is in the surface layer (where rotation effects are unimportant) which may not be valid in the presence of strong stratification [McPhee, 2008]. T_w is defined by Holland and Jenkins [1999] as the temperature in the mixed layer at the edge of the turbulent ice-ocean boundary layer. In most ocean models this is either taken as the value of the temperature in the top model cell adjacent to the boundary or temperature value representative of some distance from the ice-ocean interface [Gwyther et al., 2020]. We take T_w as the model temperature value at the computational boundary. This is perhaps an uncommon choice compared with other ocean models [Dinniman et al., 2016]. Unless doing Direct Numerical Simulation (DNS), which is prohibitively expensive for ice-ocean simulations in more than metre scale domains [Couston et al., 2021], the scalar molecular and turbulence boundary layer will not be modelled explicitly; it must be accounted for with a parameterisation. Since we are unable to model these effects explicitly we choose to imagine that the actual ice-ocean interface is offset slightly beyond the computational domain, which is similar to the approach taken for the parameterisation of wall regions in Reynolds Averaged Navier-Stokes (RANS) turbulence models [Yeager, 2018]. Figure 2.5 shows a schematic of the melt parameterisation. While the finite element solution is valid at any point within each element, and this means it is possible to offset where the tracer is sampled towards the interior of the domain as shown by Kimura et al. [2013], this could cause performance issues when running the model in parallel because the sampled tracer value may not be stored on the same processor.

The conservation of salt is similar to heat and is given by

$$Q_i^S - Q_w^S = Q_{\text{brine}}^S, \tag{2.76}$$

where the fresh water melt flux Q_{brine}^S is given by

$$Q_{brine}^S = -\rho_0 w_b (S_i - S_b), \tag{2.77}$$

and where S_i is the salinity of the ice, which is taken to be zero, and S_b is the unknown salinity at the ice base. The turbulent salt flux across the ocean boundary layer Q_w^S can be written as

$$Q_w^S = -\rho_0 \gamma_S (S_b - S_w), \tag{2.78}$$

where γ_S is the turbulent salinity exchange velocity, analogous to γ_T and as before we take S_w as the value of salinity at the edge of the computational domain offset from the ice-ocean boundary. The diffusive flux of salt into the ice, Q_i^S is assumed to be zero. The ice-ocean boundary temperature, T_b is assumed to be at the freezing point

$$T_b = aS_b + b + cP_b, \tag{2.79}$$

where $a = -5.73 \times 10^{-2}\,^\circ$C, $b = 8.32 \times 10^{-2}\,^\circ$C and $c = -7.53 \times 10^{-8}\,^\circ$C Pa^{-1} are the coefficients obtained from linearisation of the nonlinear freezing point equation [Millero, 1978] using the values from Asay-Davis et al. [2016] and P_b is the pressure at the boundary. This means there are now three equations ((2.72) (2.76) and (2.79)) to solve for the three unknowns, w_b, S_b, and T_b; the melt rate and the salinity and temperature values at the boundary, respectively. Rearranging the equations gives a quadratic equation for S_b, and the positive value is taken as the solution. By using the form of the heat and salt flux from (2.75) and (2.78), Robin boundary conditions for temperature and salinity can be imposed as shown in Figure 2.5 according to

$$\Phi_T = \kappa \mathbf{n} \cdot \nabla T = (\gamma_T + w_b)(T_b - T_w), \tag{2.80}$$

Parameter	Value	Description
f	-1.409 $\times 10^{-4}$ s^{-1}	Coriolis parameter
ρ_0	1027.51 kg m^{-3}	Reference density of sea water
g	9.81 m s^{-2}	Gravitational acceleration
α_T	-3.733 $\times 10^{-5}$ °C^{-1}	Temperature expansion coefficient
β_S	7.843 $\times 10^{-4}$	Salinity contraction coefficient
T_0	1.0 °C	Reference temperature in EOS
S_0	34.2	Reference salinity in EOS
L_f	3.34 $\times 10^5$ J kg^{-1}	Latent heat of fusion of ice
ρ_i	920 kg m^{-3}	Density of ice
c_{pi}	2000 J kg^{-1} K^{-1}	Specific heat capacity of ice
c_p	3974 J kg^{-1} K^{-1}	Specific heat capacity of water
T_i	-20 °C	Far field ice temperature
S_i	0	Far field ice salinity
a	-5.73 $\times 10^{-2}$ °C	Coefficient freezing point equation
b	8.32 $\times 10^{-2}$ °C	Coefficient freezing point equation
c	-7.53 $\times 10^{-8}$ °C Pa^{-1}	Coefficient freezing point equation
Γ_T	0.011	Turbulent heat exchange coefficient
Γ_S	Γ_T / 35	Turbulent salt exchange coefficient
C_D	2.5$\times 10^{-3}$	Drag coefficient

Table 2.5: General constants for model equations and melt parameterisation. Note in this work Salinity is treated as unitless, except for thermodynamic relations or equations of state, where it is assumed to be g kg^{-1}.

and

$$\Phi_S = \kappa \mathbf{n} \cdot \nabla S = (\gamma_S + w_b)(S_b - S_w), \tag{2.81}$$

where Φ_T and Φ_S are substituted into the third term of (2.23). Since the ice-ocean boundary is not a material interface, we include the melt water correction term accounting for the advection of melt water (with water mass properties corresponding to values at the ice-ocean interface) into the ocean domain as part of the temperature and salt flux boundary conditions following Jenkins et al. [2001].

We verify the melt parameterisation by substituting the solution to an example problem to check both sides of the simultaneous equations are consistent with each other. These tests are carried out automatically with GitHub Actions.

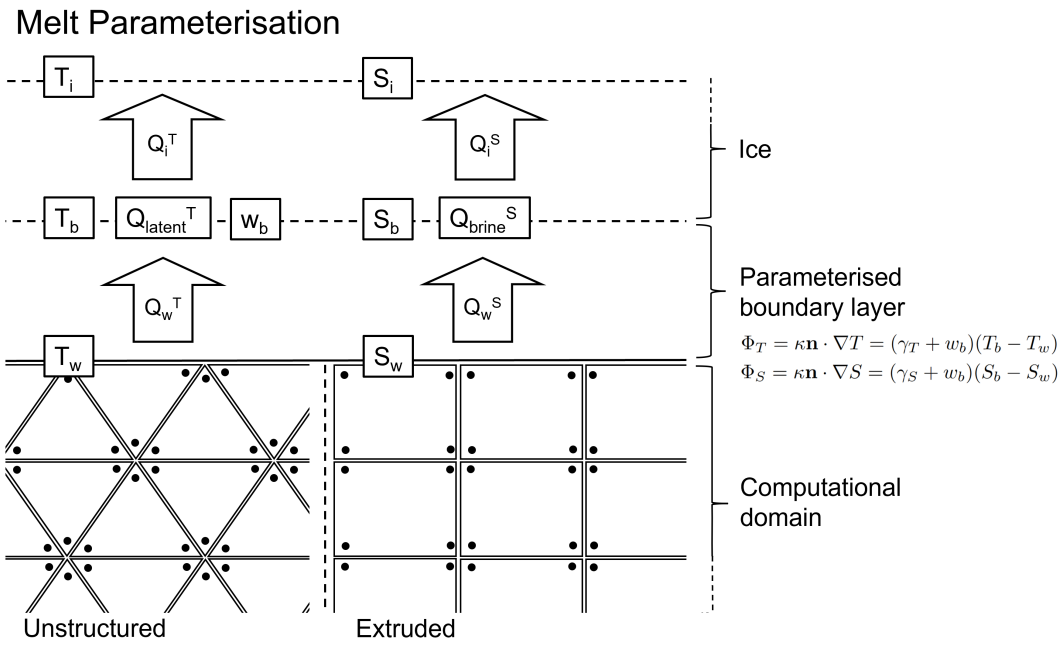


Figure 2.5: 2D vertical schematic of melt parameterisation and implementation of tracer boundary conditions. Black dots on the left hand side of the computational domain represent locations of Degrees of Freedom (DOFs) for P1DG finite element, which is used for the tracers and velocity components on vertically unstructured meshes. The right hand side of the figure represents the computational domain for the vertically structured (but horizontally unstructured) 'extruded' mesh. Black dots are DOFs for the element used for scalar quantities formed by taking the tensor product of P1DG on a horizontal interval with P1DG defined on a vertical interval. The velocity element (not shown) is also defined using a tensor product (as described in Section 2.3.2) and is discontinuous in the normal direction, but continuous in the tangential direction. Note the offset computational domain so in our modelling we assume that the ice-ocean interface is slightly beyond the computational domain following Yeager [2018]. T_w and S_w are the finite element solutions for temperature and salinity at the edge of the computational domain. Other terms are as defined in Section 2.6. After solving for T_b, S_b and w_b, Robin boundary conditions for temperature and salinity are imposed weakly (third term of (2.23)) using the definition of the heat, Q_w^T, and salt Q_w^S, fluxes from the melt parameterisation.

2.7 Slope Limiters

Slope limiters are commonly used in finite element and finite volume models. Higher order linear schemes necessarily lead to oscillatory behaviour. This is described by the well known Godunov theorem [Strang, 2007]. For nonlinear problems, these oscillations can grow to become unstable, leading to a breakdown in the numerical simulation. The motivation behind using limiters is to suppress the numerical artefacts with as little numerical diffusion as possible, so that realistic fronts can be maintained in the solution and at the same time ensure that the simulation progresses stably. Limiters are especially useful in the context of turbulence modelling, where certain physically meaningful quantities, such as the turbulent kinetic energy, must remain bounded above zero.

For linear DG finite elements, the maximum and minimum values of the solution within the element are defined at the vertices. As a result, adjusting the slope of the linear function is sufficient to maintain the solution within bounds. We use the vertex-based limiter approach from Kuzmin [2010]. The advantage of this scheme is that there are no tuneable coefficients. Often slope limiter methods rely on multiple user defined parameters which can limit their applicability in different settings. This technique adjusts the slope by ensuring that degree of freedom is within the bounds of the maximum and minimum degrees of freedom for that particular vertex.

We have found that we require a limiter on tracers to prevent a spurious freezing signal at the ice front transition, as seen in Figure 2.6. The flow accelerates at the ice front transition since the ice shelf no longer confines the buoyant water from rising upwards. Melt rates are strongly linked to flow velocity by the velocity dependent exchange coefficients, and the positive feedback mechanism between melt rate and flow velocity, where freshening leads to the injection of more buoyant water.

As a result, high flow rates at the ice front transition produce high melt rates which leads to a sudden drop in the temperature and salinity due to the Neumann boundary conditions. This rapid freshening (and cooling) changes the thermodynamic regime from melting to freezing.

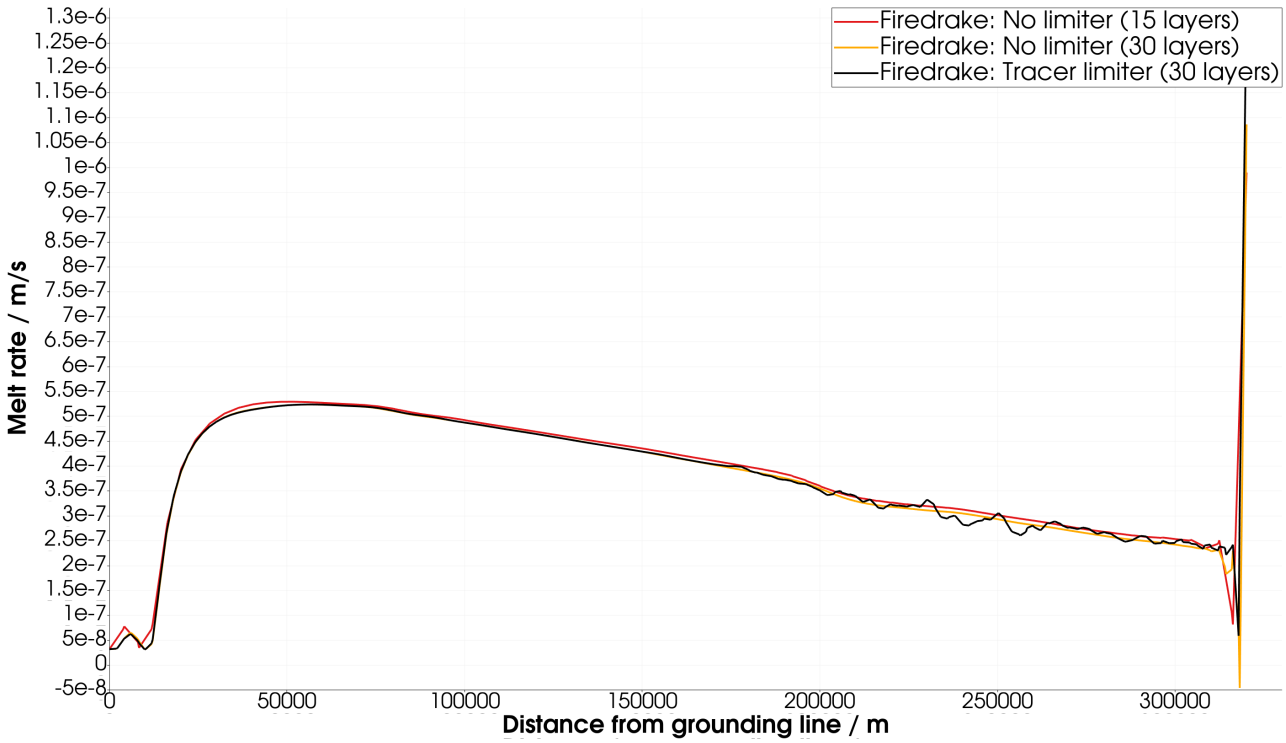


Figure 2.6: Melt rate profile at 50 days for a 2D early, pre-ISOMIP+ (see Section 3.6) simulation showing the effects of limiters on a spurious freezing signal at the ice front. The red line, without limiters, has horizontal grid resolution of 4 km and vertical resolution of 15 layers. The yellow line, without limiters has finer resolution of 2 km and 30 layers, shows a spurious freezing signal in the last cell before the ice front. The black line, with limiters applied on the tracers and the same finer resolution removes the freezing signal.

Note that the drop in salinity is more important than the cooling effect. We believe this is a spurious signal because the freezing only occurs in the final cell before the ice front transition, it is a very isolated feature. We do not fully resolve the flow at the ice front, which would require much increased vertical resolution. Enhanced mixing would occur as the buoyant melt water entrained fluid on its ascent, which would prevent the build-up of fresh melt water. Although further investigation is required, this freezing signal maybe due to the imperfect representation of a vertical step by our 'extruded' mesh. With our current meshing strategy (see Chapter 3 for more details) as the number of cells in the vertical is increased the mesh quality in the squashed ice front transition will become worse. This may explain why the freezing signal occurs when refining the mesh from 15 to 30 vertical levels.

It is also likely this signal is primarily caused by spinning the model up from unrealistic initial conditions. Freezing only begins once a noticeable 'pulse' of fresh melt water reaches the icefront, having started from the grounding line.

Using a slope limiter on tracer components prevents the freezing from occurring. This happens automatically and is likely caused by additional numerical diffusion which prevents the salinity fields from becoming too fresh at the boundary, preventing freezing. Incidentally a similar problem occurred for the initial 3D ISOMIP+ simulations. The freezing signal clearly occurs in the final grid cell on the 'western' edge before the ice front transition, where there was the fastest flow. Again the problem was solved by using a limiter on the tracer components.

Away from the ice front transition the addition of a limiter has no noticeable effect on the solution. In particular, the calculated melt rate away from the ice front transition is not affected by using a limiter on the tracer components.

We also experimented with applying a limiter to the velocity components. When not using P1DG, the velocity field was projected into P1DG first so that each component of the velocity could use the vertex-based limiter approach. We found that using a limiter on the velocity component could significantly affect melt rates. As shown in Figure 2.7, 50 km from the grounding line the melt rate for the z-layer (fixed vertical resolution – see Section 3.5) is nearly doubled by using limiters on the velocity components.

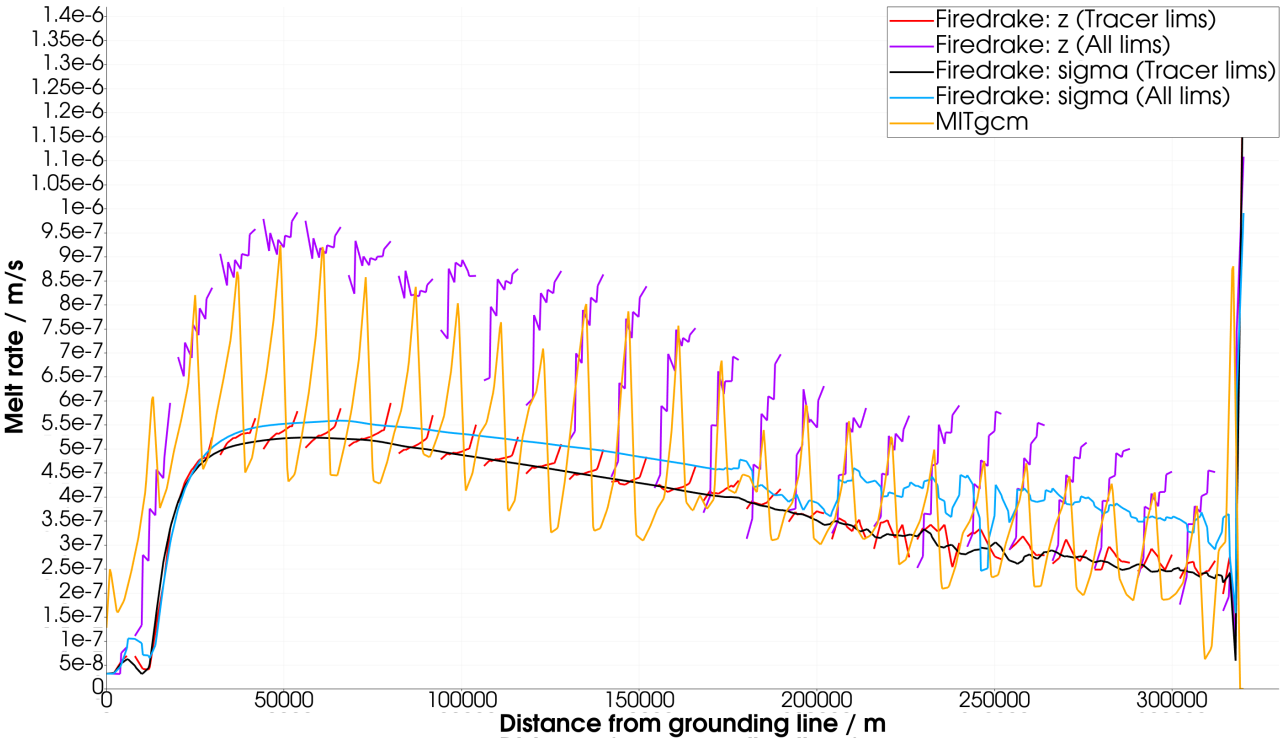


Figure 2.7: Melt rate profile for Firedrake and MITgcm showing that applying limiters on velocity components significantly affects melt rate. The red line shows Firedrake only with limiters applied to the tracer fields with a z-layer mesh: $\Delta x = 2$ km, $\Delta z = 20$ m. The purple line shows Firedrake with limiters applied to the velocity components as well as the tracer fields with a z-layer mesh: $\Delta x = 2$ km, $\Delta z = 20$ m. The yellow line shows a default MITgcm simulation with a z-layer mesh: $\Delta x = 2$ km, $\Delta z = 20$ m. The black line shows Firedrake with limiters applied only to the tracer fields with a σ-layer mesh: $\Delta x = 2$ km, 30 layers. The light blue line shows Firedrake with limiters applied to the velocity components as well as the tracer fields with a σ-layer mesh: $\Delta x = 2$ km, 30 layers. For a discussion of z vs σ-layer meshes and a detailed description of the experimental setup for the 2D extended ISOMIP-like domain please see Sections 3.4 and 3.5.

Lowest flow speeds are found at the boundary, so the limiter generally increases the flow velocity along the ice ocean boundary. This feeds directly into our melt calculation because the melt scheme we have implemented uses the velocity defined along the boundary.

This is a resolution dependent effect. Refining the vertical resolution decreases the magnitude of the effect. Moreover, the terrain following mesh (fixed number of layers – see Section 3.5) with higher resolution at the ice-ocean boundary shows less of an effect than the z-layer mesh. The fact that the limiter is having such an effect, consistently in cells right along the ice ocean boundary, probably should have been an early indication for the 3D ISOMIP+ test case in Section 3.6 that the flow was not being resolved sufficiently with 20 m resolution. If the flow were being resolved the limiter would not turn on.

It should be pointed out that we use the tracer limiters with an implicit time integration scheme. This is not guaranteed to prevent overshoots, but so far, in practice, this has not been a problem.

2.8 Model Implementation

We have built a Python module on top of Firedrake to facilitate simulations of ice shelf cavities. The core part of our Python package is made up of 11 modules. We make use of Python classes to build up the complexity of the code in blocks. Each equation inherits useful attributes from a BaseEquation, i.e. a method for summing the residuals from each term. And each equation is made up of terms, effectively each term in (2.26) and (2.20), themselves based on a BaseTerm, which contains the required features of a term, i.e. a method to evaluate the residual form for the term. This approach leads to a compact codebase, aids with debugging, as individual pieces of the code can be isolated and tested, but retains the flexible nature of Firedrake, which is one of the main reasons for using the software. The code is available at https://github.com/thwaitesproject/thwaites.git. It has also been stored in this Zenodo repository: https://doi.org/10.5281/zenodo.7584700, along with instructions for running the examples from Sections 2.9, 3.6 and 4.3 that appear in Scott et al. [2023].

The next section presents a simplified verification test applicable to ice-ocean models, which will hopefully prove of value for other models, to ensure that the code is solving the model equations correctly and consistently.

2.9 Method of Manufactured Solution (MMS) tests

The Method of Manufactured Solutions (MMS) is a rigorous method for verifying that the numerical scheme as implemented is solving the model equations accurately and consistently [Farrell et al., 2011, Roache, 1998]. This is a separate problem to that of validating whether the model, including the choice of underlying equations, yields a good approximation to the real system. Such a validation requires comparison with observations, which is challenging for simulations of ice shelf cavity circulation. The most reliable way to test whether a numerical approximation is consistent is to ensure that the solution error relative to a known solution decreases at the correct order as the grid is refined [Salari and Knupp, 2000]. However, finding analytical solutions to systems of coupled Partial Differential Equations (PDEs) that arise in fluid dynamics is notoriously difficult, and only possible for simple geometrical configurations and often simplified Boundary Conditions (BCs). MMS removes much of this difficulty. Instead of solving the PDE system to find the solution, the analytical form of the solution is chosen to begin with. Solution fields are constructed from smoothly varying differentiable functions, often trigonometric and polynomial functions. These functions are substituted into the strong form of the equations and the resulting residuals (which are expected to be non-zero as these functions could not be expected to represent an exact solution) provide additional source terms that can be used to force a version of the simulation for which the chosen functions are exact solutions. The chosen functions can be manufactured to meet specific constraints (for example incompressibility of the flow field) or specific BCs. Alternatively, consistent BCs can be found directly from the chosen solutions simply by evaluating the solution at the boundary for the type of BC required for testing.

Following the notation from Farrell et al. [2011], the error on a given mesh is

$$E_{h_1} \approx C h_1^{c_p}, \tag{2.82}$$

where C is a constant, h_1 is the characteristic mesh size, and c_p is the order of accuracy of the numerical approximation. The error on a finer mesh is

$$E_{h_2} \approx C \left(\frac{h_1}{r} \right)^{c_p}, \tag{2.83}$$

where the new characteristic mesh size $h_2 = h_1/r$. By taking the ratio of the two errors the convergence rate can be found using

$$c_p \approx \log_r \left(\frac{E_{h_1}}{E_{h_2}} \right). \tag{2.84}$$

The MMS solutions should be constructed to be as general as possible to ensure good coverage of the terms in the equations [Salari and Knupp, 2000]. Although this does not require the chosen solution fields to be realistic, the solution we have chosen for this MMS test is an overturning flow that is a crude representation of an ice-pump mechanism.

For simplicity we have limited ourselves here to steady flow within a 2D vertical slice. The domain is a square box with length $L = 100$ m, height $H = 100$ m, and with the bottom boundary at a depth $D = 1000$ m. The chosen solution for the horizontal velocity is

$$u = u_0 \frac{x}{L} \cos \left(-\frac{\pi(z + D - H)}{H} \right), \tag{2.85}$$

where u_0 is a characteristic velocity of 1 m/s, x and z are the horizontal and vertical coordinates. As an example of the MMS procedure, Figure 2.8 shows the finite element solution fields for the horizontal velocity component on the two coarsest grids alongside the corresponding error relative to the chosen solution. The error on the finer mesh is noticeably reduced compared to the error on the coarser mesh despite the two solution fields being qualitatively similar.

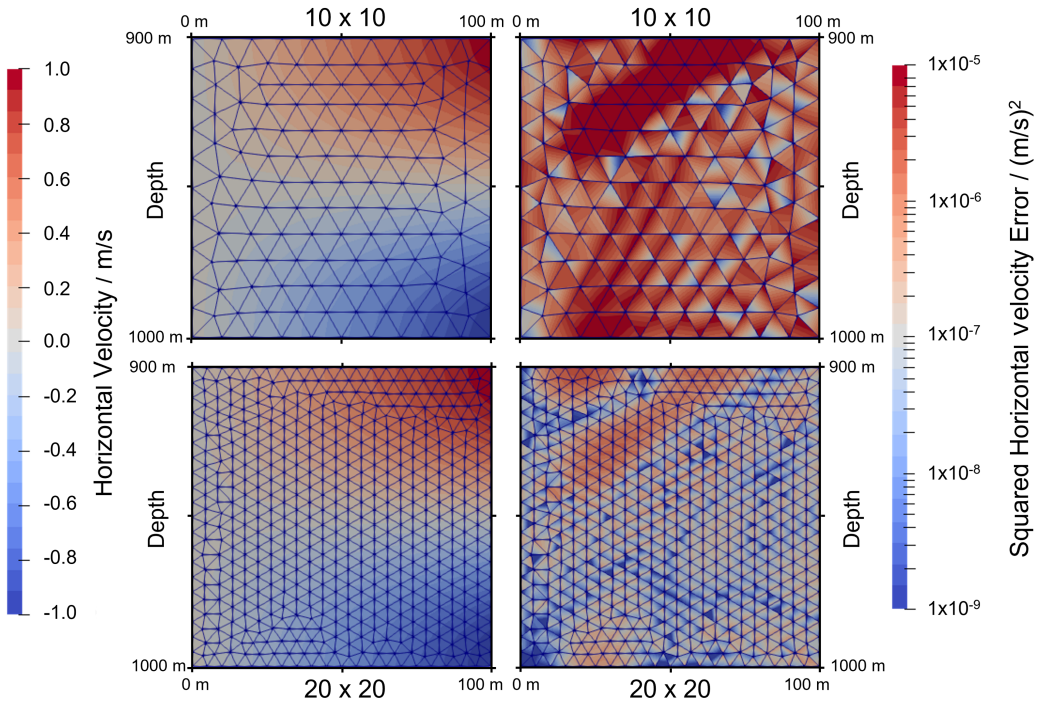


Figure 2.8: Example MMS solution and error fields. Left panel shows example solution field for horizontal velocity components on the two coarsest grids (10x10 cells and 20x20 cells). Red indicates flow to the right and blue indicates flow to the left. The right panel shows the squared error of the finite element solution relative to the chosen manufactured solution on the two meshes. Note the log scale on the right hand plot. Although the solution fields in the left hand panels at the two resolutions are visually similar, the error is clearly lower on the finer grid in the right hand panel. This plot is intended as a qualitative example of MMS, Figure 2.9 demonstrates that the solution fields converge at the correct order.

The chosen solution for vertical velocity is

$$w = u_0 \frac{H}{\pi L} \sin\left(-\frac{\pi(z + D - H)}{H}\right). \tag{2.86}$$

This has been chosen to ensure that the flow is incompressible so that the continuity equation, $\nabla \cdot \mathbf{u} = 0$, is satisfied. For this simple flow configuration, the vertical velocity does not depend on x. The right hand side is specified as an open boundary and all other boundaries employ free-slip conditions, with no normal flow imposed. The pressure field has been chosen to have homogeneous Neumann boundaries ($\nabla p \cdot \mathbf{n} = 0$)

$$p = p_0 \cos\left(\frac{\pi x}{L}\right) \cos\left(-\frac{\pi(z + D - H)}{H}\right), \tag{2.87}$$

where p_0 is a constant set to 1 Pa to ensure the dimensions remain consistent and the reference density, ρ_0, is set to 1 kg m^{-3} for this test. The temperature field is

$$T = T_c \sin\left(\frac{4\pi x}{L}\right) + a_T z^2 + b_T z + c_T, \tag{2.88}$$

where T_c is 0.1 °C, a_T is -3.89×10^{-4} °C m^{-2}, b_T is -7.54×10^{-1} °C m^{-1}, and c_T is -3.64×10^2 °C. The salinity field is

$$S = S_c \cos\left(\frac{4\pi x}{L}\right) + a_S z^2 + b_S z + c_S, \tag{2.89}$$

where S_c is 0.345, a_S is -1.44×10^{-4} m^{-2}, b_S is -2.81×10^{-1} m^{-1}, and c_S is -1.03×10^2. The viscosity and diffusivity are 1 m^2/s.

As already described, the source terms are derived by substituting the solutions into the strong form of the equations. The source term for velocity is thus given by

$$\mathbf{S_u} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho_0} \nabla p - \nabla \cdot \boldsymbol{\tau} + \frac{\rho'}{\rho_0} g \mathbf{k}. \tag{2.90}$$

Note the coriolis term is not included because the domain is a 2D vertical slice. In general the source term can vary with time, but since we have chosen the velocity solution to be steady the time derivative will not contribute to the source term. The implementation of rotation has

been verified against analytical solutions of Ekman spirals in Section 2.10. The source term for temperature is given by

$$\mathbf{S}_T = \frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T - \nabla \cdot \kappa_T \nabla T. \tag{2.91}$$

The source term for salinity is given by

$$\mathbf{S}_S = \frac{\partial S}{\partial t} + \mathbf{u} \cdot \nabla S - \nabla \cdot \kappa_S \nabla S. \tag{2.92}$$

As with velocity, the salinity and temperature source terms will be time independent due to the steady chosen solutions for tracers. Use of symbolic computation tools, such as SymPy [Meurer et al., 2017], make this process relatively straightforward to implement. For example, the source term for the horizontal velocity component in this case is given by

$$\begin{aligned}\mathbf{S}_u = \frac{-p_0 \pi \sin(\pi x/L)\cos(\pi(-H+D+z)/H)}{L\rho_0} + \frac{u_0^2 x \sin(\pi(-H+D+z)/H)^2}{L^2} + \\ \frac{u_0^2 x \cos(\pi(-H+D+z)/H)^2}{L^2} + \frac{\pi^2 \nu u_0 \, x \cos(\pi(-H+D+z)/H)}{H^2 L}.\end{aligned} \tag{2.93}$$

Note we chose this term for its compactness because it does not include buoyancy terms.

Dirichlet BCs for temperature and salinity are applied on the left and right boundaries, with Neumann conditions on the bottom boundary. Values for these boundary conditions are found in a similar manner to defining source terms, the solution itself is substituted to give the exact form of the (generally inhomogeneous) boundary conditions [Salari and Knupp, 2000]. Particular attention is given to the boundary conditions at the ice-ocean interface situated at the top of the domain, as verifying the accuracy of the model implementation at this location is a key reason for carrying out the MMS test. The boundary conditions at the ice-ocean interface are represented by Robin boundary conditions, which define the gradient of the temperature and salinity fields at the boundary as a function of the temperature and salinity at the edge of the computational domain (see Section 2.6 for more information). The default boundary condition for the melt boundary is given by (2.80) and (2.81), which are imposed weakly in the third term of (2.23). For an MMS test the temperature boundary condition applied at the top

is given by

$$\Phi_T = \kappa \mathbf{n} \cdot \nabla T^a + \Phi_T^h - \Phi_T^a. \tag{2.94}$$

The first term in (2.94) represents the temperature gradient of the analytical temperature field, T^a, (found from (2.88)), at the boundary with the eddy diffusivity as a multiplication factor as discussed previously. This is the only term needed to implement Neumann boundary conditions for conventional MMS tests. The second term Φ_T^h is the flux boundary condition calculated by the melt parameterisation using the modelled temperature (and salinity) values, equivalent to (2.80) (and (2.81)). The final term, Φ_T^a, is the flux calculated by the melt parameterisation using the analytical temperature field. Provided the temperature, salinity and velocity (through $u^\star$) fields converge at the correct order, then the melt rate error (calculated using the exact temperature, salinity and velocity fields) should also converge at the correct order because the contribution from Φ_T^h and Φ_T^a cancel out.

This is a spatial convergence test so the solution fields have been chosen to be stationary [Farrell et al., 2011]. The simulation is spun up from rest. Backward Euler is used for timestepping the momentum equations and the scalar equations. The timestep is initially set to 4 m/$(u_0 n_x)$, chosen for robustness in the initial spin-up, where n_x is the number of cells in the x direction. The timestep is increased to L/$(u_0 n_x)$ after 100 time steps.

Velocity and tracer fields are discretised with P1DG elements and so second order convergence is expected, which means that halving the grid size should reduce the error by a factor of four. Pressure is discretised with P2 elements so third order might be expected; however, coupling errors associated with the velocity mean that in practice the convergence is about second order [Cotter and Ham, 2011].

Figure 2.9 plots the errors for each solution field, with logarithmic axes. Velocity, temperature and melt rate are all very close to the expected second order convergence or higher. Salinity and pressure are slightly lower, though never below 1.8 and the convergence rate is increasing towards second order as the grid is refined. The integrated melt is also initially above second order but drops to 1.82 convergence. The errors and convergence rates are also given in Table 2.6. Overall these convergence results give confidence in the accuracy and consistency of the

	10x10	20x20	40x40	80x80
temperature	6.60e-1	1.52e-1	3.71e-2	9.43e-3
	-	2.12	2.03	1.98
salinity	2.19	5.95e-1	1.57e-1	4.10e-2
	-	1.88	1.92	1.94
melt	5.31e-7	1.21e-7	2.93e-8	7.39e-9
	-	2.14	2.04	1.99
integrated melt	3.71e-7	6.23e-8	1.40e-8	3.95e-9
	-	2.57	2.16	1.82
u velocity	2.34e-1	5.60e-2	1.40e-2	3.52e-3
	-	2.06	2.00	1.99
v velocity	2.21e-1	5.53e-2	1.41e-2	3.54e-3
	-	2.00	1.97	1.99
pressure	3.96e-1	1.13e-1	3.10e-2	8.36e-3
	-	1.80	1.87	1.89

Table 2.6: MMS convergence tests for temperature, salinity, melt, integrated melt, velocity and pressure. MMS convergence tests for temperature, salinity, melt, integrated melt, velocity and pressure. The test was carried out on four grids with grid resolution of 10m (10x10 cells), 5m (20x20 cells), 2.5m (40x40 cells), 1.25m (80x80). The error is given in the L_2 norm above the order of convergence from the previous grid.

numerical discretisation and hopefully will prove to be a useful tool as the model continues to be developed. In the following section we demonstrate our ability to simulate more complex ice shelf cavity geometries in 3D using the ISOMIP+ *Ocean0* test case.

2.10 Ekman Spiral

In order to test the implementation of the Coriolis term, simulations were run and compared against an analytic solution for an Ekman spiral. It is interesting to note that, as McPhee [2008] states in the introduction of his book, the motivation for Ekman's work came from observations in the Arctic when Nansen on board the Fram was stuck in the sea ice floe and noticed that they did not drift directly downwind. As McPhee [2008] also notes, Ekman spirals were first observed unambiguously in an ice-ocean setting since sea-ice acts to damp turbulence due to wave forcing which makes observation in the open ocean of Ekman spirals very difficult.

Simulations were carried out both for 3D and 2.5D configurations. 2.5D refers to a method of adding rotational effects to a 2D vertical slice by assuming that there are no gradients in the

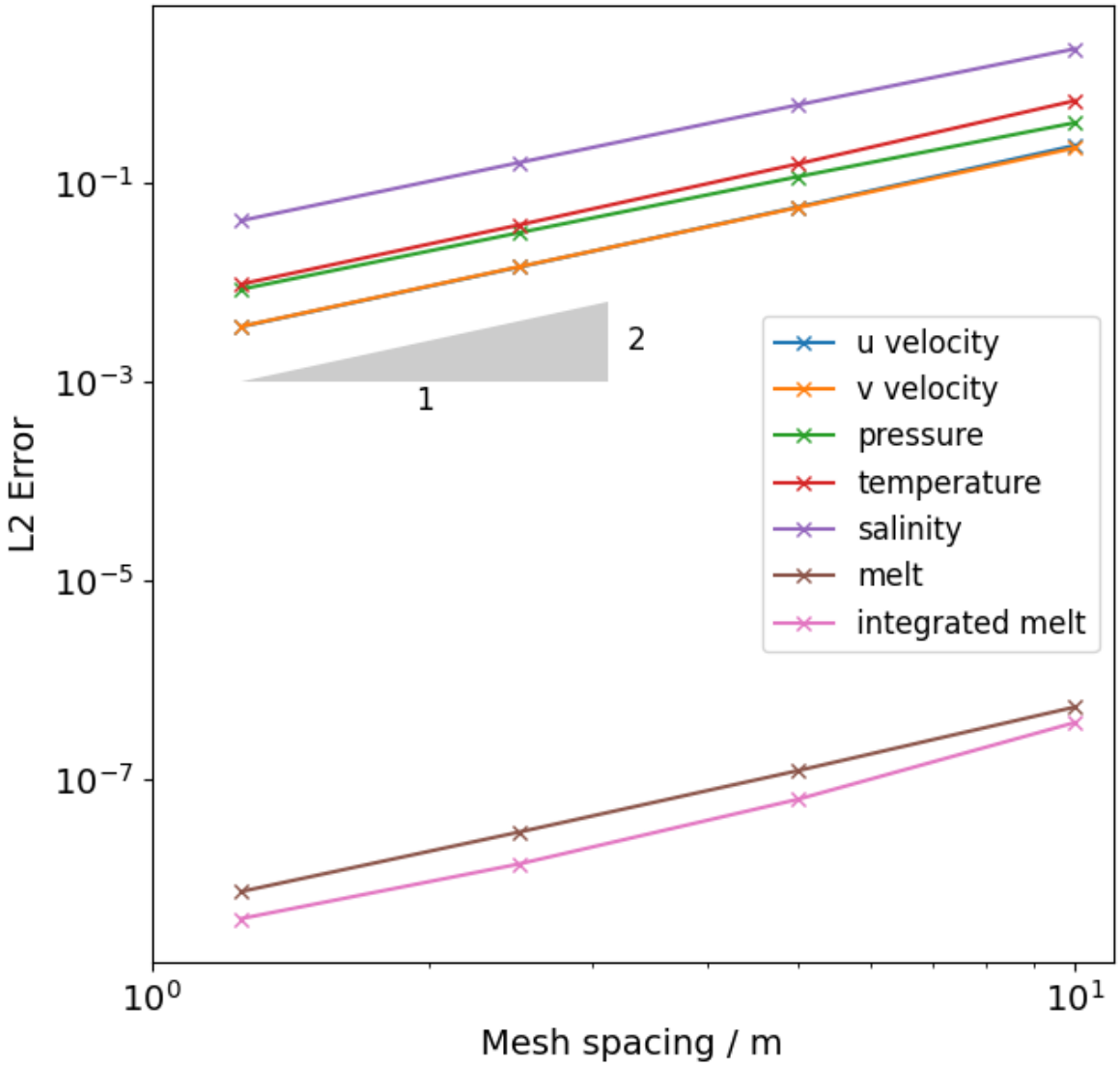


Figure 2.9: MMS convergence tests for temperature, salinity, melt, integrated melt over the top boundary, velocity and pressure. The test was carried out on four grids with grid resolution of 10 m (10×10 cells), 5 m (20×20 cells), 2.5 m (40×40 cells), 1.25 m (80×80). The error is given in the L_2 norm. Both axes use a logarithmic scale. A triangle with slope of two has been added to the plot to help show that all fields achieve the expected second order convergence.

Parameter	Value	Description
L	100 m	Domain length
H	100 m	Domain height
D	1000 m	Domain depth
u_0	1 m s^{-1}	Characteristic velocity
p_0	1 Pa	Characteristic pressure
ρ_0	1 kg m^{-3}	Reference density
T_c	0.1 °C	Coefficient in chosen temperature solution
a_T	-3.89×10^{-4} °C m^{-2}	Coefficient in chosen temperature solution
b_T	-7.54×10^{-1} °C m^{-1}	Coefficient in chosen temperature solution
c_T	-3.64×10^2 °C	Coefficient in chosen temperature solution
S_c	0.345	Coefficient in chosen salinity solution
a_S	-1.44×10^{-4} m^{-2}	Coefficient in chosen salinity solution
b_S	-2.81×10^{-1} m^{-1}	Coefficient in chosen salinity solution
c_S	-1.03×10^2	Coefficient in chosen salinity solution
μ	1 m^2 s^{-1}	Viscosity
κ	1 m^2 s^{-1}	Diffusivity
n_x	Variable	Number of grid points
Δt	4 m / ($u_0 n_x$)	Timestep (first 100 steps)
Δt	L / ($u_0 n_x$)	Timestep (after 100 steps)

Table 2.7: Parameters and constants used for MMS convergence test in Section 2.9.

third 'into-the-page' direction. This assumption of having infinitely long domains is effectively the same as when using periodic boundary conditions. Equation 2.95 shows the equation for the additional u velocity component

$$\frac{\partial u}{\partial t} + \mathbf{u} \cdot \nabla u - fv = \nabla \cdot (\boldsymbol{\nu} \cdot \nabla u). \tag{2.95}$$

Note in this formulation the vertical slice is assumed to be in the y-z plane, so the velocity vector, $\mathbf{u}$, corresponds to $(v, w)^T$. This is effectively a scalar advection diffusion equation similar to Equation 2.14, but with an additional forcing term coming from the momentum equation. A corresponding source term, fu, is added to the momentum equations to provide coupling between the u and v components. Although the equations and the Coriolis term are not as tightly coupled as the 3D equations, provided the dynamics are slow and the timesteps is sufficiently small this type of coupling is robust and provides a useful modelling strategy either as a stepping stone to 3D or when the geometry or flow in the third 'into-the-page' direction is not well constrained.

In this work Coriolis is implemented using the f-plane approximation, where the Coriolis parameter is constant, here with $f = 1.023 \times 10^{-4}$ s^{-1}. For process based studies on a small domain, which are the object of this thesis, this is a valid approximation but for larger scale ocean circulation problems it is important that the strength of the Coriolis force changes with longitude.

The domain has a length of 100 m (width of 100 m for 3D) and a depth of 100 m. The horizontal grid resolution is 25 m and the vertical grid resolution is 2 m. The simulation is forced with a normalised wind stress, $|\boldsymbol{\tau}|$, on the top surface with a magnitude of 1.775×10^{-4} m^2/s^2 in the y-direction. The top surface is treated as a rigid lid, with no normal flow. Side walls are periodic to allow the Ekman spiral to develop unimpeded. Periodicity is enabled by using Firedrake's inbuilt utility meshes. The bottom boundary is set to no normal, no slip flow. Horizontal viscosity is set to 100 m^2/s and vertical viscosity, ν_v, is set to 1.4×10^{-2} m^2/s. Note that tracers are not included in this setup. The velocity is initialised from rest and the solution is run for 10 days until a steady state is reached. The time step is 60 s.

The analytic solution for the Ekman spiral as a function of vertical coordinate, z (positive upwards) is given by

$$u = u_0 \mathrm{e}^{\pi z/D} \cos(4\pi + \pi z/D), \tag{2.96}$$

$$v = u_0 \mathrm{e}^{\pi z/D} \sin(4\pi + \pi z/D), \tag{2.97}$$

where the characteristic velocity, u_0 is given by

$$u_0 = \frac{|\boldsymbol{\tau}| D}{\sqrt{2}\pi\nu_v}, \tag{2.98}$$

the Ekman depth, D, is given by

$$D = \pi\sqrt{2\nu_v/f}, \tag{2.99}$$

and the surface normalised wind stress is

$$|\boldsymbol{\tau}| = \frac{\rho_{\text{air}}}{\rho_{\text{water}}} C_s |\mathbf{u}_{10}|. \tag{2.100}$$

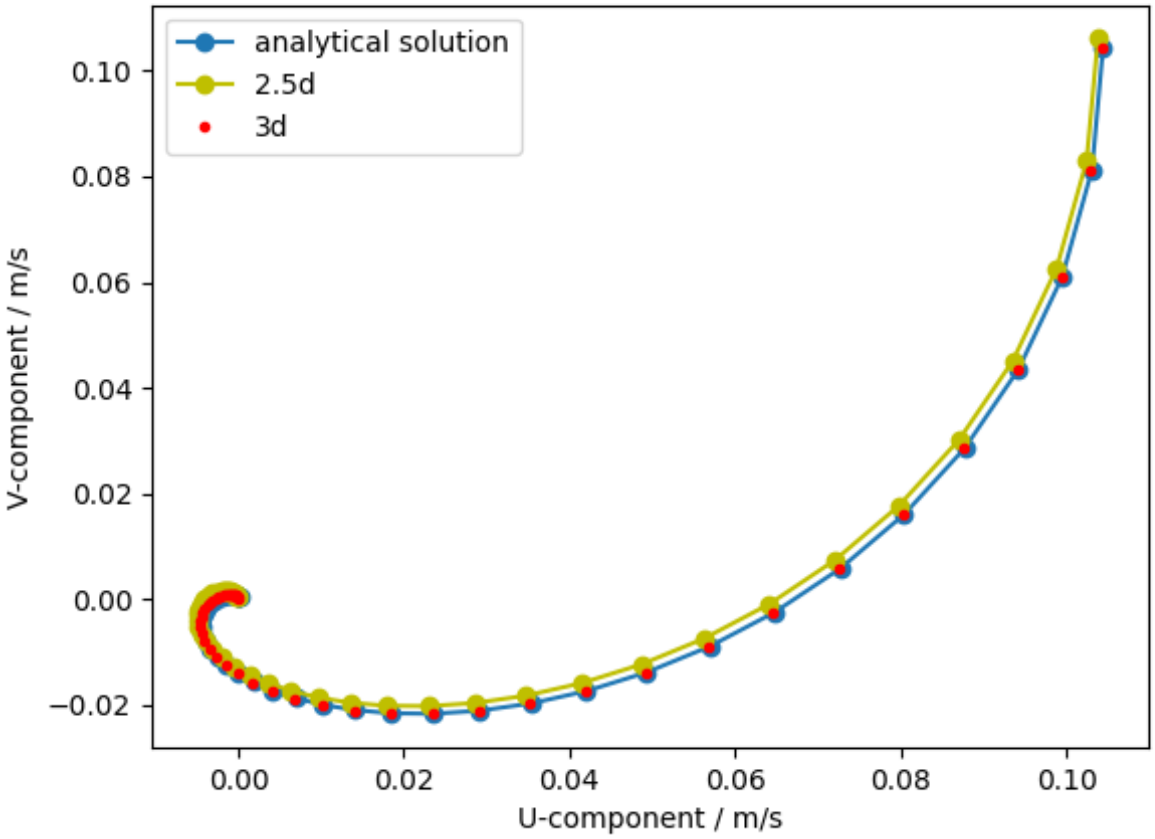


Figure 2.10: Ekman spiral results for 2.5D and 3D simulations compared with analytical solution.

With the drag coefficient, $C_s = 1.4{\times}10^{-3}$, $\rho_{\text{air}} = 1.3$ kg m^{-3}, $\rho_{\text{water}} = 1025.34$ kg m^{-3} and the horizontal wind speed at 10 m height, $\mathbf{u}_{10} = (0,\ 10\ \text{m/s})^T$, this gives $|\boldsymbol{\tau}| = 1.775{\times}10^{-4}$ m^2/s^2 as noted above.

As can be seen from Figure 2.10 both the 2.5D and 3D solutions fit the Ekman spiral very well.

2.11 Code Availability

The code is available at https://github.com/thwaitesproject/thwaites.git.

Run scripts for MMS verification test with melt rate (Section 2.9):

- thwaites/tests/mms_groundingzone_TS/mms_groundingzone.py

Run scripts for Ekman layer simulations (Section 2.10):

- thwaites/demos/ekman/ekman2.5D.py
- thwaites/demos/ekman/ekman3D.py

2.12 Conclusion

In this chapter we described the model equations and our flexible discretisation approach. We used a new velocity-pressure element pair for large scale ocean applications defined using tensor products on Firedrake's 'extruded' mesh functionality. As part of the description of our Discontinuous Galerkin discretisation we highlighted that the iterative solver performance was sensitive tothe interior penalty scheme. We also showed how slope limiters can have a non-negligble impact on melt rates, especially when applied to the velocity components, so care needs to be taken when using them for ice-ocean applications. Limiters proved crucial, however, in eliminating a spurious freezing signal at the ice front region.

The second part of the chapter demonstrated that the model is implemented successfully. In particular, we developed a novel Method of Manufactured Solutions (MMS) verification test that includes melting, thus rigorously verifying that the spatial discretisation of the model is second order accurate as expected. We also compared our model with analytical results for an Ekman spiral, confirming the accurate implementation of rotation for both the 3D and 2.5D formulations. From this solid base, the next chapter applies the model to more complicated simulations of ice shelf ocean cavities carrying out a benchmark comparison against MITgcm, an ocean model commonly used for ice ocean research.

Chapter 3

Model Benchmarking vs MITgcm

Highlights

- We show good agreement with MITgcm for a range of test cases, including the ISOMIP+ *Ocean0* test case simulating a full 3D ice shelf ocean cavity.
- We demonstrate a range of meshing techniques suitable for ice shelf cavities.
- Our model is shown to have robustness and accuracy benefits over MITgcm for a given mesh resolution, owing to the higher order discretisation.
- We investigate trade-offs between grid resolution, spurious numerical mixing and melt rate convergence.

Summary

The MMS test presented in Chapter 2 is a rigorous verification of the model discretisation and melt parameterisation, but it does require simplifying the test problems. Here we present a suite of simulations comparing our model against MITgcm, a popular ocean model used for ice shelf cavity modelling. The tests become progressively more complex in geometry and physics,

culminating in the community standard 3D ISOMIP+ *Ocean0* test case. For all the experiments we see very good agreement between Firedrake and MITgcm, giving good confidence in our model implementation. We find that our model discretisation tends to be more robust and accurate than MITgcm for a given mesh resolution, again showing the potential of the model. We demonstrate a range of meshing techniques including a direct comparison of z versus σ-layer that is uncommon within the same model. The penultimate section on the 3D ISOMIP+ test case has a particular emphasis on the implications of numerical, spurious mixing that is inherent to virtually all models to varying degrees and the impact that this has on obtaining melt rates independent of the model resolution, which is a pressing area of ice-ocean modelling research. The final section presents run times for the simulations in this Chapter and shows that given the tradeoff between accuracy and computational expense our Firedrake model is competitive with MITgcm.

Publications and Disclaimer

Some results from Section 3.6, demonstrating the 3D ISOMIP+ comparison between Firedrake and MITgcm are shown in Scott et al. [2023], but have been expanded for this work. Ben Yeager kindly provided an initial MITgcm setup similar to the domain from Section 3.2. The discussions of numerical mixing and melt rate convergence in Section 3.6 were helped by a succession of Masters students supervised by Professor Matthew Piggott using Background Potential Energy as a measure of spurious mixing for a Rayleigh Taylor instability problem using our Firedrake model.

3.1 Introduction

Our motivation for the validation exercise was to test the accuracy and stability of our implementation. Although verification tests like the MMS test in Section 2.9 are rigorous, these results often require the physics of the problem being solved to be simplified. In this Chapter

we carry out tests of increasing complexity, using a range of meshing techniques, for simulations of ice shelf cavities. The simulations are idealised but by increasing the complexity iteratively we build good confidence in the model. Crucially, we compare our model with MITgcm for each simulation. We chose to compare Firedrake against MITgcm, as it is an established ocean model used to simulate circulation beneath ice shelves. MITgcm is a finite volume, z-layer (fixed vertical resolution), ocean model widely used in a number of applications including investigations of ice shelf-ocean cavities [Losch, 2008, Dansereau et al., 2014, Seroussi et al., 2017, Nakayama et al., 2017, Kimura et al., 2017, Nakayama et al., 2021a, Naughten et al., 2021]. It also has a nonhydrostatic option, selected for these comparisons, so that both models solve the same underlying set of equations, albeit with different discretisations.

3.2 2D Benchmark vs MITgcm for an Idealised Grounding Zone Domain

Our initial validation exercise compares a simulation of an idealised 2d ice shelf cavity with the same simulation by MITgcm. The two models are in very good agreement. A particularly positive result is that Firedrake seems to be numerically more stable than MITgcm for the same grid resolution, timestep, viscosity and diffusivity values.

The setup was kept purposefully simple. The model domain is a 2D vertical slice through an idealised ice shelf cavity. The domain is 10 km long, with a horizontal grid resolution of 50 m. The ice shelf draft linearly increases from a depth of 598 m at the grounding line to a depth of 498 m at the right-hand boundary. Bathymetry is constant throughout the domain at 600 m depth. Vertical grid resolution is 1 m, except along the top boundary where grid resolution decreases to 0.5 m in an alternating pattern designed to match the slope of the ice shelf. This choice of mesh is motivated by MITgcm's z-coordinate grid. Both models solve the incompressible, nonhydrostatic Navier-Stokes equations under the Boussinesq buoyancy approximation. Melt rates are calculated using the three equation parameterisation with velocity dependent turbulent exchange coefficients as described in Holland and Jenkins [1999].

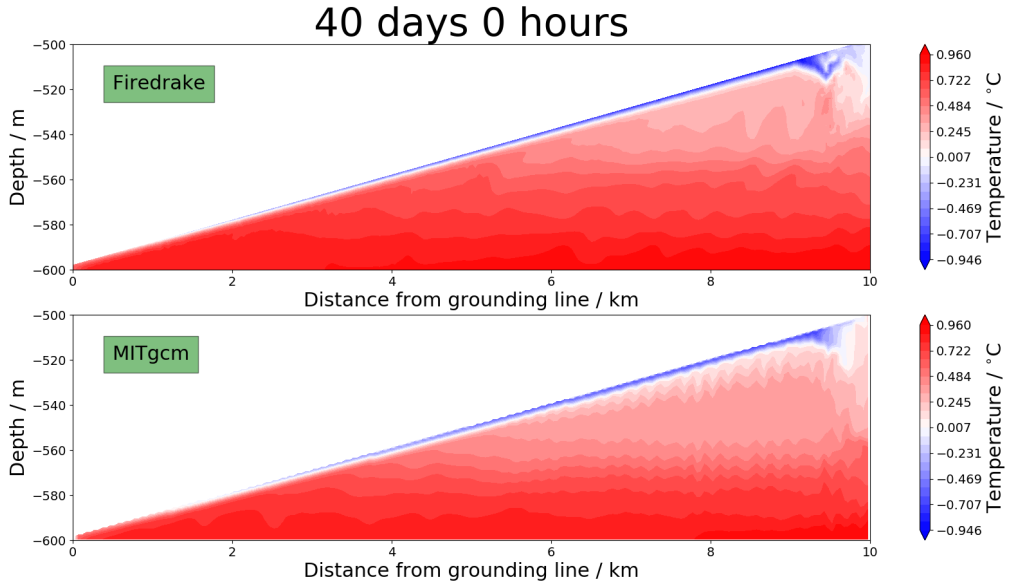


Figure 3.1: Plots of temperature for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days.

Temperature was initialised with constant values of 1 °C and salinity with 34.4. These values were used to restore the region over the last 600 m, with a restoring period of 1 day. Viscosity was set equal to diffusivity for both scalars. Horizontal viscosity was set to 1×10^{-3} m^2/s, and vertical viscosity was set to 1×10^{-4} m^2/s.

The agreement between MITgcm and Firedrake is very good. Both models predict the same density stratification after 40 days as shown by the temperature and salinity plots in Figures 3.1 and 3.2, respectively. Note these are snapshots not time averages. Figures 3.3 and 3.4 show the dominant flow in both models is driven by the buoyant melt water that rises up the underside of the ice shelf in a concentrated plume. The structure and magnitude of this clockwise overturning circulation is very similar in both models.

Both models predict undulations in density contours though the structures are not identical. The 'wiggles' seem to be generated at the right-hand side of the domain in both models. Most activity is seen near the edge of the restoring region in the top right corner. Even if the cause of the flow is not physically meaningful, both models are demonstrating the same numerical behaviour. Assuming the MITgcm simulation is a measure of 'truth' for this setup, this implies that we are solving the system accurately.

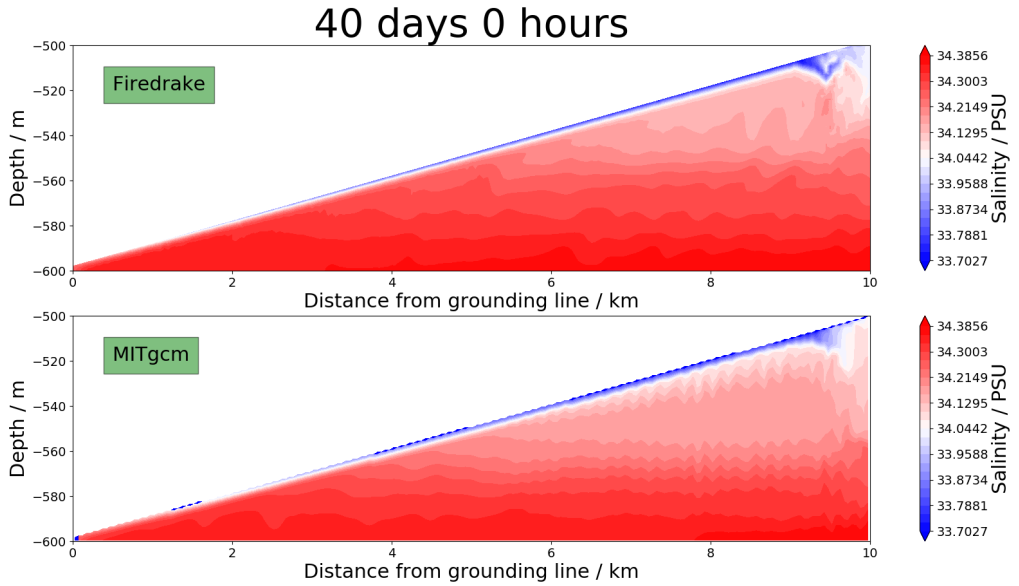


Figure 3.2: Plots of salinity for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days.

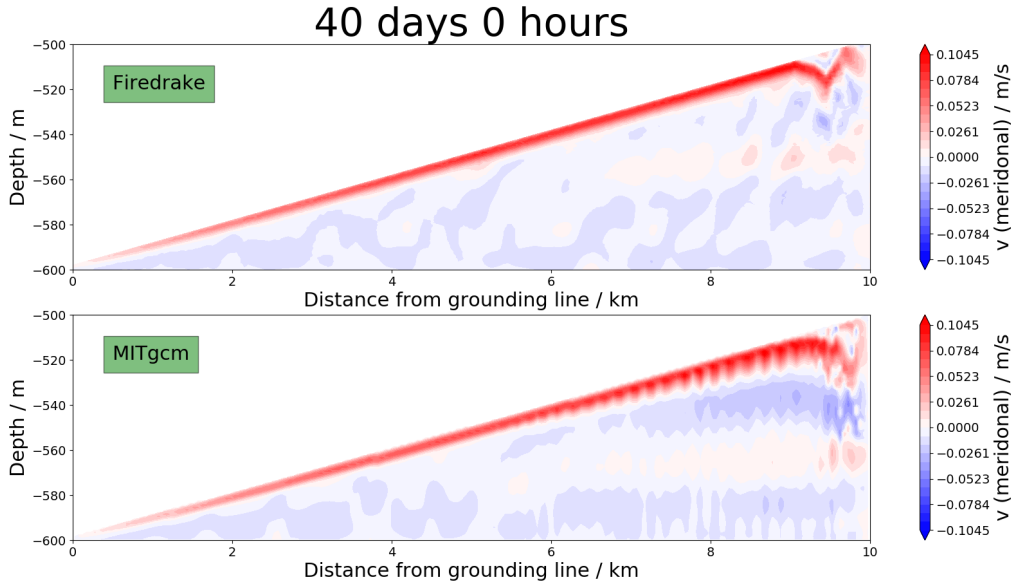


Figure 3.3: Plots of horizontal velocity for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days.

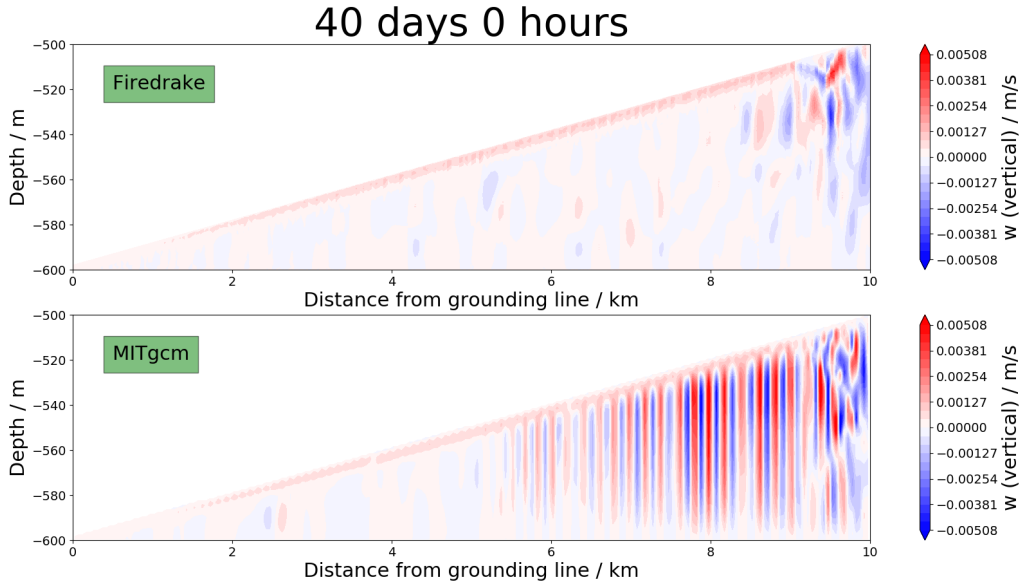


Figure 3.4: Plots of vertical velocity for the 2D idealised grounding zone comparison between MITgcm and Firedrake after 40 days.

There are less 'wiggles' in the Firedrake simulation. In MITgcm these 'wiggles' seem to match up with the grid. This can be seen in plots of MITgcm's temperature and salinity fields, but it is particularly noticeable in plots of vertical velocity in Figure 3.4. These vertical stripes are characteristic of grid scale numerical 'noise', which is dominating the small vertical velocities associated with the ice pump mechanism. In contrast, in the Firedrake simulation these 'wiggles' appear more like resolved eddies and do not seem to be directly related to the grid resolution. It is important to emphasise that both models have the same grid resolution, timestep, viscosity and diffusivity values. Clearly, for this simulation our underlying numerical discretisation is less sensitive to grid scale instabilities, which is a very positive result. This is not entirely unexpected because we are using piecewise linear discontinuous elements (P1DG) for tracers and velocity (see Sections 2.3.1 and 2.3.2), whereas MITgcm's low order finite volume scheme is effectively equivalent to a piecewise constant (P0DG) finite element scheme.

One other difference, is a time lag as the initial density front propagates after 1 day of spin up (not shown). In MITgcm the front lags behind the Firedrake simulation by 500 m. By two days the noise in the vertical velocity component of MITgcm obscures this time lag. Similar behaviour to this occurred when experimenting with timestep values to assess Firedrake's sta-

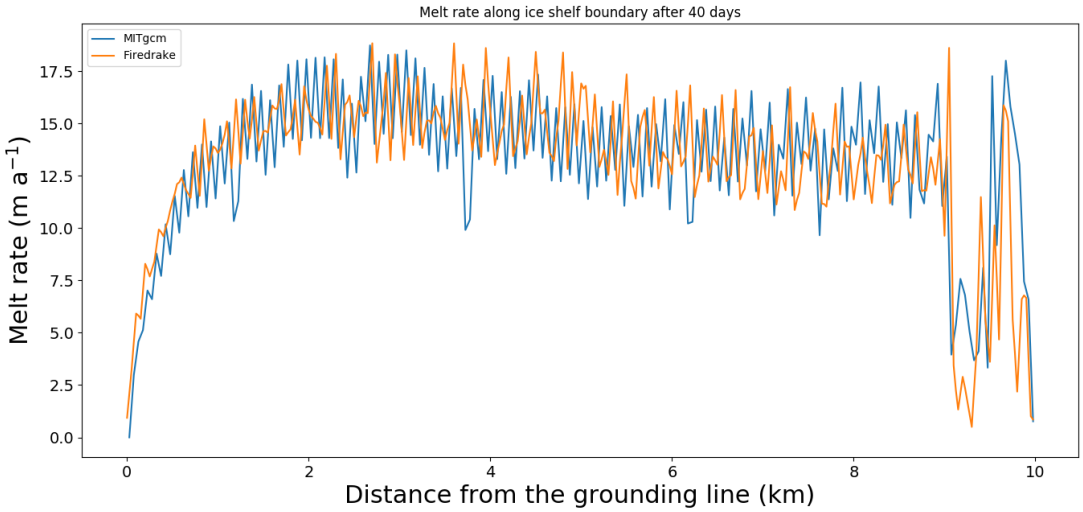


Figure 3.5: Melt rate profile along ice shelf base for the 2D idealised grounding zone comparison after 40 days.

bility. Simulations with longer timesteps (which tended to be less stable) demonstrated a lag in the front propagation. It is worth noting however, that even though these simulations showed different behaviour during spin up they reached a similar 'steady' state after 20 days provided the time step was not too large.

The melt rate profiles of both models agree very well as shown in Figure 3.5. This is probably the best quantitative test that the two models are producing similar results. Both models show melt rate oscillations. This is due to variable cell size used by both models (smaller triangles in Firedrake and partial cells in MITgcm) to represent the slope of the ice shelf base. This change in grid resolution impacts rates of diffusion of temperature and salinity near the ice base which in turn affect melt rates. This 'stripy' melt rate pattern has been well documented by MITgcm modellers and there are numerous boundary layer scheme to lessen the effect [Losch, 2008, Jordan et al., 2018]. These schemes were turned off in this comparison. Firedrake is not limited to using a z-coordinate mesh so does not inherently have these problems. Future work should investigate the effects of boundary layer averaging schemes and similar implementations in Firedrake.

Overall, we are very pleased with the comparison exercise. Not only are the results between the models very similar, implying we are solving the physics accurately, but our numerical

discretisation is less sensitive to an underlying instability in the setup than MITgcm. We would like to emphasise that this simple idealised setup does not demonstrate the full potential of Firedrake though. It proved to be a very beneficial process, because this helped to identify an important bug in the implementation of the DG advection term during the initial development. The following sections and Chapter 5 demonstrate our flexibility in being able to run simulations on different types of unstructured meshes. Moreover, this setup does not test the nonhydrostatic capability of either model. Turning off the non-hydrostatic terms in MITgcm had very little impact on the solution, only affecting vertical velocities in a minor way. The investigation in Chapter 5, especially with frazil ice dynamics, shows simulations on isotropic meshes where retaining the full physics is important.

3.3 2.5D Benchmark vs MITgcm for an Idealised Grounding Zone Domain

The 2D model was extended to include a scalar advection diffusion equation for the 'out of the page' horizontal vertical component, to include the effect of rotation, as discussed in Section 2.10. Figure 3.6 shows a comparison of the temperature fields between MITgcm and Firedrake after 40 days. Again, the results are very similar for both models. The thickness of the boundary layer is larger than the 2D case from Section 3.2. The u component of velocity (+ve 'out of the page'), as shown by Figure 3.7, shows a similar pattern with a strong flow into the page near the ice ocean boundary, as the buoyancy driven v flow deflects to the left due to the Coriolis effect. The horizontal velocity field at 40 days is shown in Figure 3.8. The agreement between both models is very good. The velocity boundary layers is actually thinner than for the 2D case. There is a noticeable return flow at the bottom boundary of similar strength to the ice-ocean boundary layer flow. There is some overturning dynamics at the right hand side due to the restoring sponge, but the simulations are 'quieter' than the 2D case, in the sense that there are less obvious grid artefacts present in the vertical velocity of MITgcm as shown by Figure 3.9.

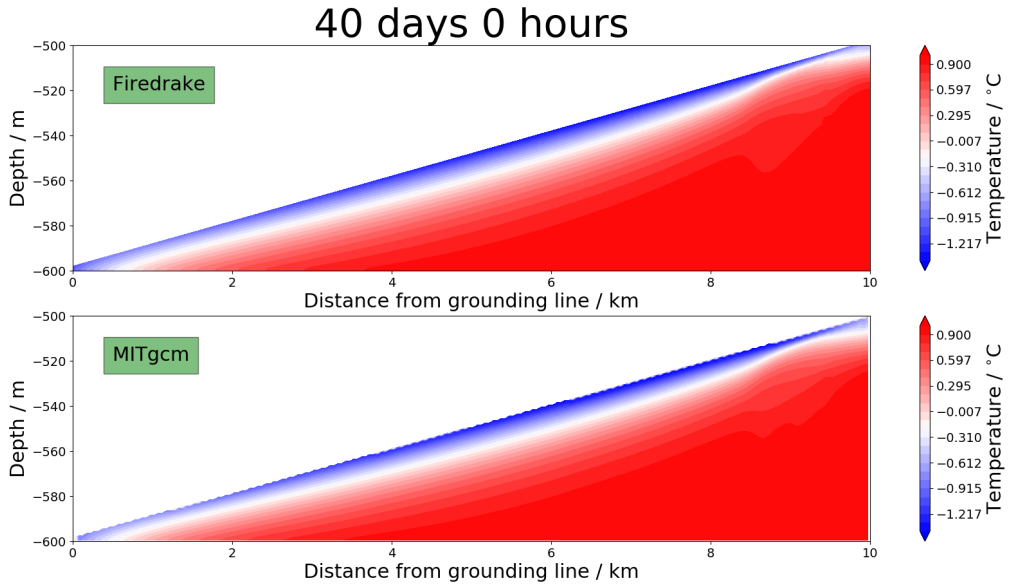


Figure 3.6: Plots of temperature for the 2.5D idealised grounding zone comparison between MITgcm and Firedrake after 40 days.

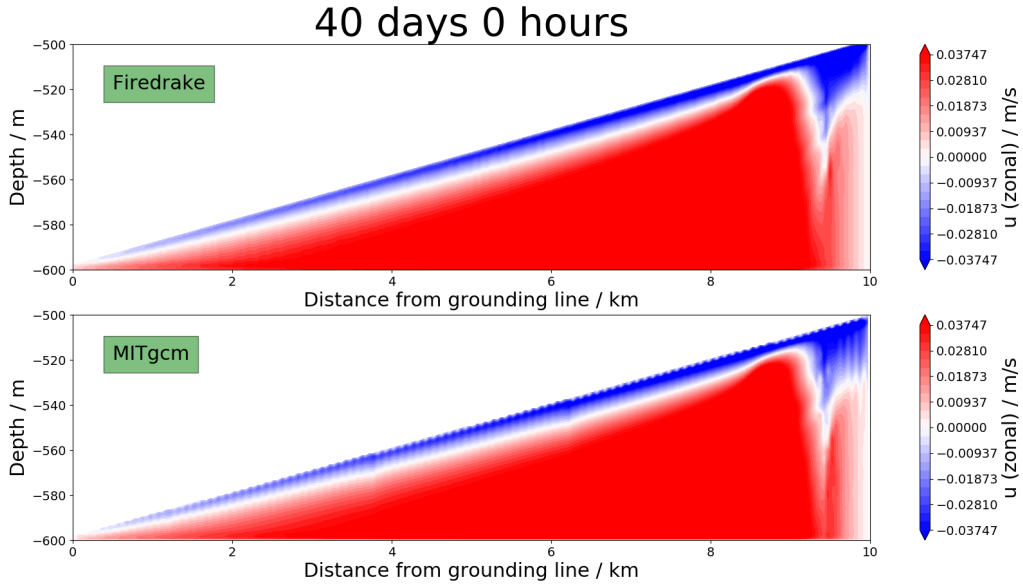


Figure 3.7: Plots of u velocity (positive 'out of the page') for the 2.5D comparison idealised grounding zone between MITgcm and Firedrake after 40 days.

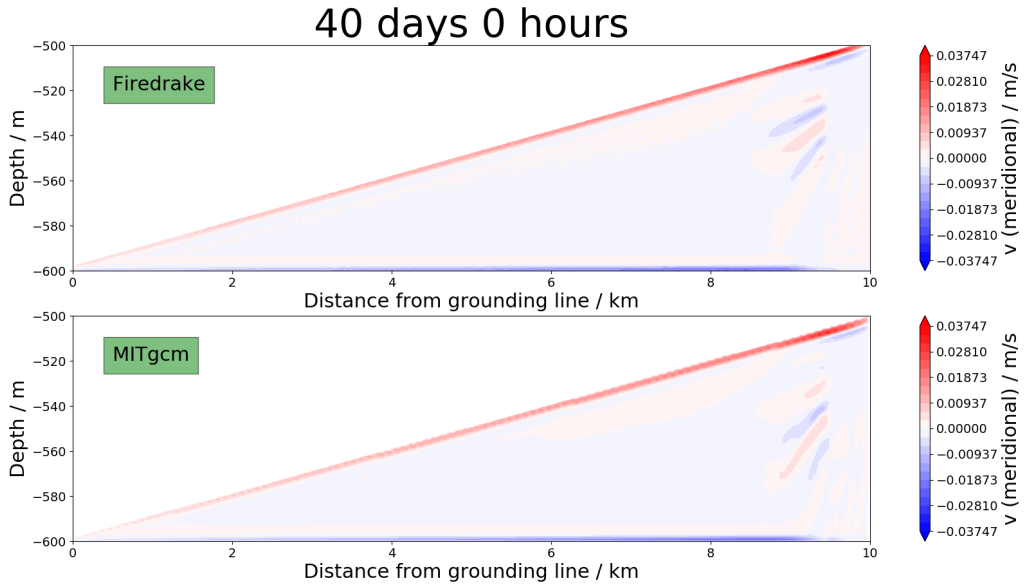


Figure 3.8: Plots of v velocity (positive to the right) for the 2.5D idealised comparison grounding zone between MITgcm and Firedrake after 40 days.

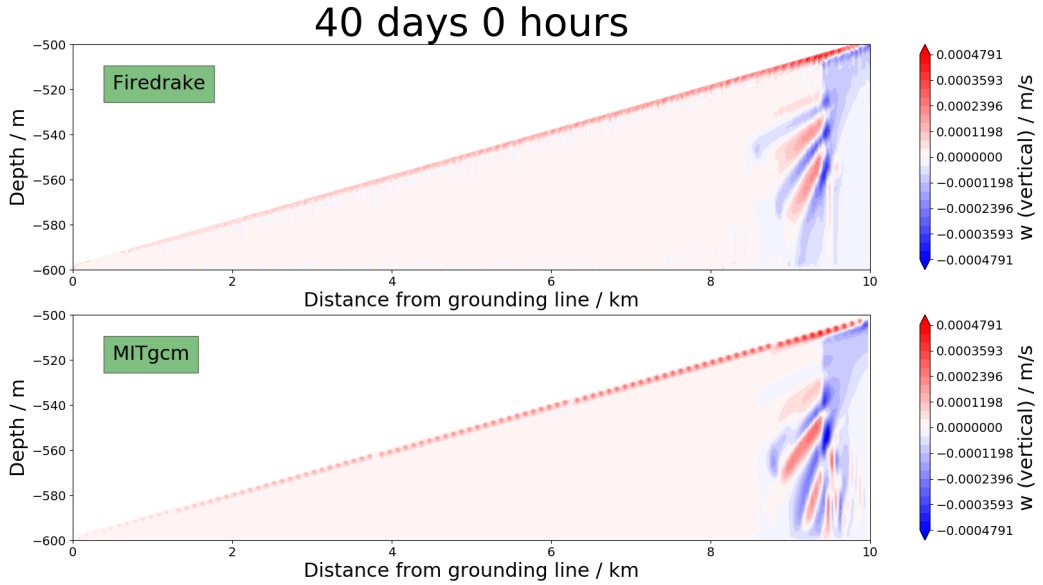


Figure 3.9: Plots of vertical velocity for the 2.5D idealised grounding zone comparison between MITgcm and Firedrake after 40 days.

3.4 2D Benchmark vs MITgcm for an Extended ISOMIP-like Domain

As a stepping stone to the 3D community standard ISOMIP+ test case (see Section 3.6) some 2D simulations were run comparing MITgcm and Firedrake for a domain of equivalent size to the full ISOMIP+ configuration. The domain length was 480 km with a constant bathymetry of 720 m. The ice shelf has a linear slope with ocean thickness of 40 m at the grounding line and 600 m at the ice front. Horizontal viscosity was set to 6 m^2/s and vertical viscosity was set to 1×10^{-3} m^2/s. Diffusivity was set equal to viscosity. The timestep for both models was 900 s. The ISOMIP+ target grid resolution is 2 km in the horizontal direction and 20 m for z-layer models (constant vertical resolution).

For this comparison we used Firedrake's inbuilt 'extruded' mesh with the tensor product element pair (RTCE2-Q2 in 2D) to mimic the z-layer representation of MITgcm. Note the implementation of the 'extruded' mesh assumes there are no interior facets that are exposed on the outside of the mesh, i.e. vertical steps in the mesh are not permitted. At the ice-ocean interface this means that, in a similar manner to partial cells employed by MITgcm, the elements change thickness when the mesh is squashed to remove the steps. When an additional cell is added to the column there are triangular elements at the top of the mesh to fit with the quadrilaterals in the other columns. Another consequence of this is that the ice front is not vertical for the Firedrake model, instead there is a transition which occurs over one grid cell, where the additional elements added to the column for the open ocean region are 'squeezed' into a stack of triangular elements to give a sloping ice front. This can be seen in Figure 3.13.

The extruded mesh is used here for two reasons. First, initial tests in 3D using a tetrahedral mesh were not running stably with the default parameters specified by ISOMIP+, notably the vertical viscosity of 1×10^{-3} m^2/s. This led to us using the 'extruded' mesh approach and trying the new tensor product elements as discussed in Section 2.3.2 to improve robustness in large horizontal aspect ratio domains. At the same time the pressure solve was very slow, which motivated the use of the 'vertical lumping' preconditioner as discussed in Section 2.5 due to

having anisotropic grid cells. As part of this investigation it turned out that having a sponge region acting over the last four grid cells imposing a linear stratification based on the 'warm' restoring profile from Asay-Davis et al. [2016] was essential in getting the 1×10^{-3} m^2/s to run stably, as referred to in Tables 2.3 and 2.4.

Figure 3.10 shows the temperature stratification for MITgcm and Firedrake is qualitatively very similar. There is an obvious colder (and fresher) boundary layer at the ice-ocean interface in both models. There is warm water close to the grounding zone owing to the linear stratification imposed on the RHS in the sponge region. Figure 3.11 clearly shows the dominant feature of the flow in both models is the buoyancy driven flow at the ice-ocean boundary, reaching speeds of 0.2 m/s. Outside the cavity there is also faster flow at the ocean surface in both models. The flow beyond the ice shelf front oscillates for MITgcm, compared with Firedrake where the flow has reached a steady state. This appears to be caused by the vertical ice front, whereas in Firedrake the ice front is tilted as referred to earlier. Perhaps the best indication is that the melt rate produced by Firedrake is within the spread of the MITgcm melt rate as shown by Figure 3.12. Similar to Section 3.2 we found that due to the higher order velocity grid discretisation, Firedrake was visually very similar to MITgcm at double the resolution. i.e. for Firedrake with grid size of $\Delta x = 4$ km, $\Delta z = 40$ m performed favourably compared with MITgcm with grid resolution of $\Delta x = 2$ km, $\Delta z = 20$ m and even $\Delta x = 1$ km, $\Delta z = 10$ m as shown by the melt rates in Figure 3.12. Both models show the obvious 'stripy' melt rate characteristic of a z-layer mesh where the top cell size changes based on grid size since they are not using boundary layer averaging schemes [Losch, 2008, Jordan et al., 2018], though it is worth emphasising that for Firedrake the effect is less pronounced for the same resolution as MITgcm. The next section will show a simulation using a corresponding σ-layer mesh, with a fixed number of cells under the ice shelf, where the melt rate has much less obvious 'stripy' patterns.

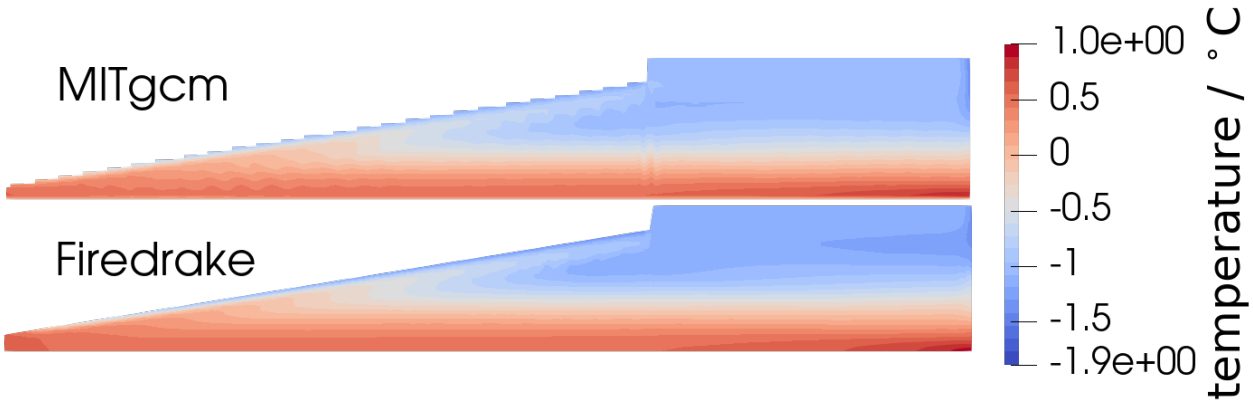


Figure 3.10: Plots of temperature for the 2D pre-ISOMIP+ comparison between MITgcm and Firedrake after 100 days. Both models have horizontal resolution of 2 km and vertical resolution of 20 m.

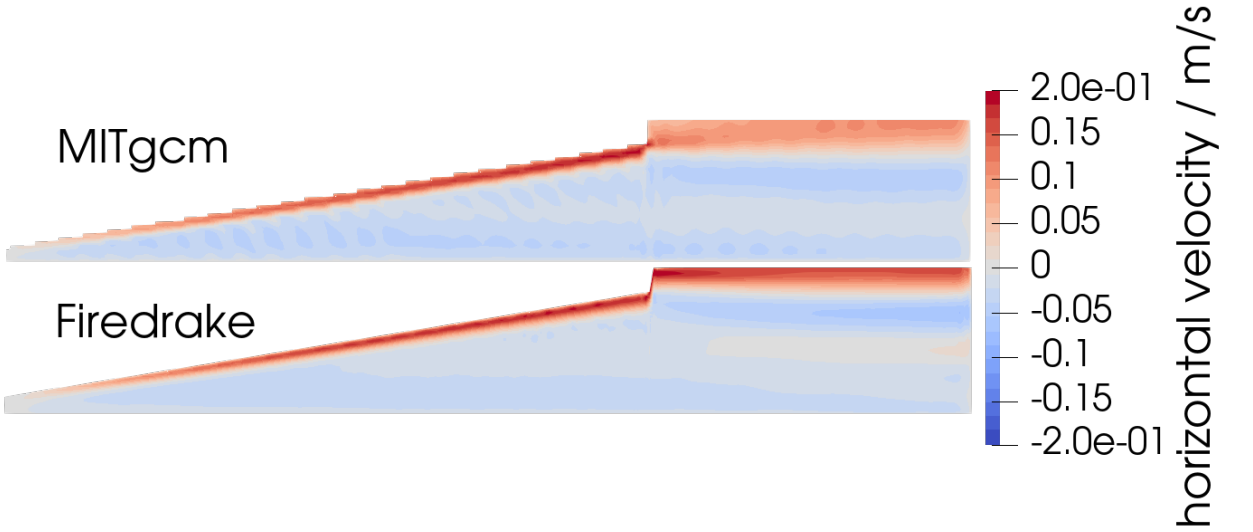


Figure 3.11: Plots of horizontal velocity for the 2D pre-ISOMIP+ comparison between MITgcm and Firedrake after 100 days. Both models have horizontal resolution of 2 km and vertical resolution of 20 m.

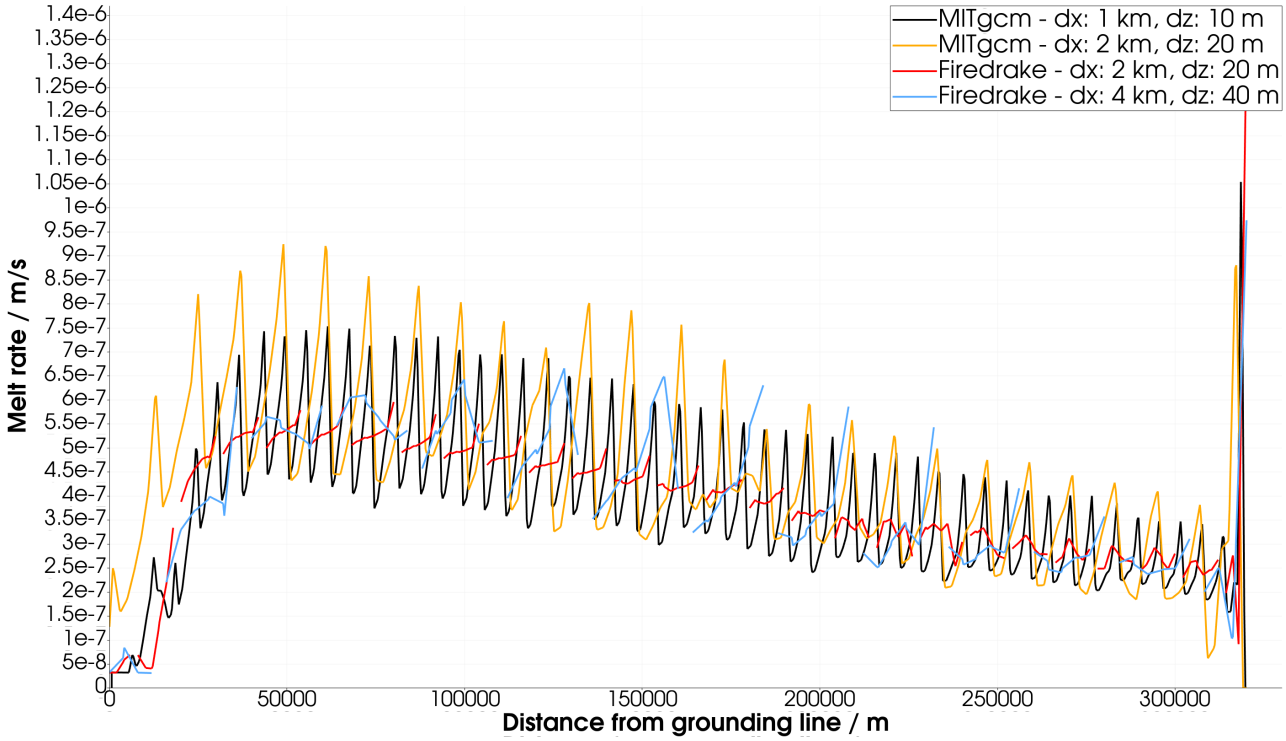


Figure 3.12: Melt rate profile for the 2D pre-ISOMIP+ comparison between MITgcm and Firedrake at 50 days.

3.5 2D z vs σ-coordinate Model Comparison for an Extended ISOMIP-like Domain

In the vertical direction, apart from z-layer models like MITgcm, which have constant vertical resolution, there are broadly two other types of structured discretisation: terrain following coordinate models (e.g. ROMS: [Shchepetkin and McWilliams, 2005]), which solve a transformed set of equations so that the domain is discretised into vertical layers, and isopycnal models (e.g MICOM: [Bleck et al., 1992]), which rescale the vertical coordinate in terms of potential density.

Isopycnal models are well suited to modelling density driven flows in the ocean, as the discretisation captures the importance of stratification in the ocean, which leads to enhanced mixing along isopycnals. MICOM was adapted by Holland and Jenkins [2001] to simulate flow under ice shelf cavities by adding a non-isopycnal boundary layer to allow buoyancy forcing associated with melting at the ice shelf ocean boundary to drive the model. There are difficulties associ-

ated with ensuring that this boundary layer does not adversely affect modelled ocean properties and dynamics outside of the cavity, however, as the layers must be matched either side of the ice front [Holland and Jenkins, 2001, Kimura et al., 2013]. The need to prescribe density levels can be also be problematic, as typical ocean properties underneath ice shelf cavities can be very different to those found in the open ocean [Losch, 2008, Kimura et al., 2013].

As seen from Figures 3.5 and 3.12 an obvious drawback of z-layer models in an ice-ocean setting is that the constant vertical resolution can only resolve the sloping cavity geometry as a series of steps. This can lead to an instability developing if the steps are too large and result in vertical flow isolated between columns. A common way to reduce this instability is to use partial cells [Adcroft et al., 1997]. The top cell of each column is artificially corrected so that it approximates the slope of the ice base better. This, however, leads to a 'stripy' melt rate pattern, since heat and salt diffuses through cells of smaller size more quickly, as seen already in the previous Sections 3.2-3.4 when comparing Firedrake with MITgcm without the mitigating effects of a boundary layer averaging scheme [Losch, 2008, Jordan et al., 2018].

In a terrain following model the vertical grid resolution changes smoothly between each column, providing a smooth representation of the ice base, hence leading to a smooth melt rate profile. The layers also converge towards the grounding line, so the grounding zone becomes more resolved. Although this can be useful, if there are too many layers terrain following models will become unstable, not only due to time step constraints, but also because pressure gradient errors will occur for lower slope angles. A dimensionless stability criterion for this instability has been referred to as the Haney number [Haney, 1991], values less than 4 are typically stable [Schnaase and Timmermann, 2019]. To resolve the pinching grounding line, z-coordinate models need to have very small cells near the grounding line. Since this resolution is constant this means that in other parts of the domain where high resolution is not needed this is a waste of computational effort. This is particularly problematic if the model is being run for a global simulation [Schnaase and Timmermann, 2019].

z-coordinates are well adapted to deal with the steep changes in water depth at the calving front, provided constant vertical resolution (along with the timestep) at the ice front can resolve

the sudden upward flow stably. Terrain following models, however, are inherently unable to model steep vertical changes in the ice shelf draft, e.g. at the calving line. This means that the ice front must be smoothed in some way [Beckmann et al., 1999]. Due to the conservation of potential vorticity, step changes in water depth act as barriers to barotropic flow [Pedlosky, 2013]. Although stratification of the water column allows baroclinic flow to penetrate into the cavity, there is a danger that if terrain following models smooth the ice front too much, then the ocean dynamics and hence the melt pattern will be inaccurate [Hattermann et al., 2014, Zhou and Hattermann, 2020].

Since our model discretisation is not tied to a particular choice of mesh it is possible to run Firedrake simulations using a σ-layer (terrain following) mesh. Indeed, it is possible to combine aspects of σ and z-layers within the same mesh. For instance, by having a fixed number of cells within the ice shelf cavity and a higher, fixed number layers outside of the cavity, with the transition mimicking a single column of a z-layer mesh. Figure 3.13 shows an example of a z-layer and a σ 'extruded' meshes with an ice front transition based on the 2D MITgcm comparison in Section 3.4 and Section 3.5. Note this is not a perfectly vertical step, as in a true z-layer model, because the implementation of 'extruded' meshes in Firedrake requires that there are no exposed interior facet boundaries, so the the mesh is squashed over the final cell as shown in Figure 3.13. As the number of cells in the vertical is increased the mesh quality in the squashed ice front transition will become worse. Possibly this is responsible for the spurious freezing signal discussed in Section 2.7 that the limiters successfully remove, though further investigation is required.

Comparisons of melt rates between z and terrain following models have been done before, but here, the comparison is carried out with the same underlying model that has only been shown by Kimura et al. [2013]. There is no difference in the physical approximations or numerical discretisation, so any differences are purely due to the choice of vertical mesh strategy. Figure 3.14 shows a comparison of the melt rate using a z-layer mesh vs a σ-layer mesh. The blue line is the melt from the z-layer mesh with a vertical grid size of 40 m (i.e. 18 layers outside the cavity). It shows the characteristic 'stripy' melt pattern which is inherent to z-layer models. The σ-layer model, with 15 layers under the ice shelf and 18 layers for the open ocean, by contrast has a

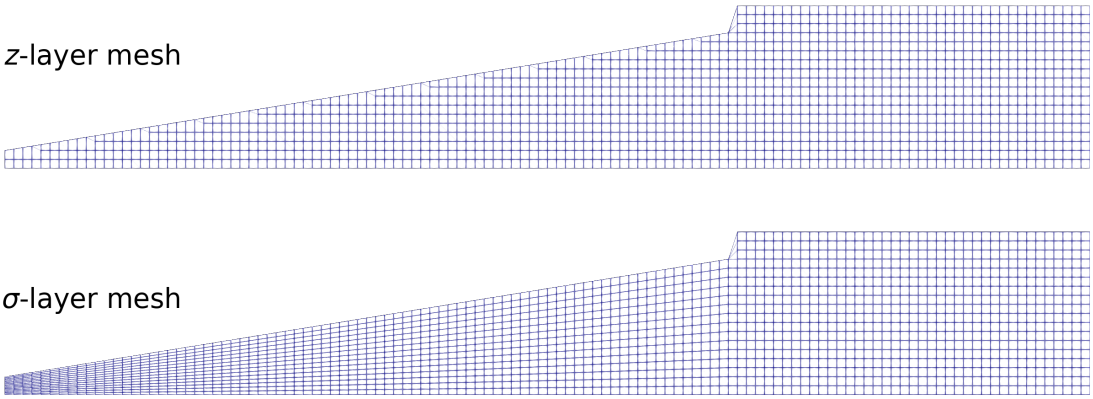


Figure 3.13: Example of z-layer (constant vertical resolution) and σ-layer (fixed number of layers) meshes built using Firedrake's 'extruded' mesh functionality.

much smoother melt profile. The profile dissects the z layer model which suggests there is good qualitative agreement between the two grid types. Also note that Figure 2.7 from Section 2.7 on using limiters has a comparison of melt rates for MITgcm and Firedrake for both z ($\Delta z =$ 20 m) and σ-layer (30 layers under the ice shelf) meshes. The agreement between the Firedrake z and σ-layer is very good and both models are well within the spread of the 'stripy' MITgcm melt rate.

It is worth pointing out that in these 2D simulations the difference in melt rates between z and σ-layer are not that obvious. This motivated moving forward with only a σ-layer meshing strategy in 3D for the ISOMIP+ domain, in order to avoid the 'stripy' melt rate issue. However, as explored further in Section 3.6 in 3D the more complicated geometry along the grounding line led to quite a significant difference between the z-layer discretisation of MITgcm and our σ-layer discretisation making it more difficult than anticipated to use MITgcm as a measure of truth.

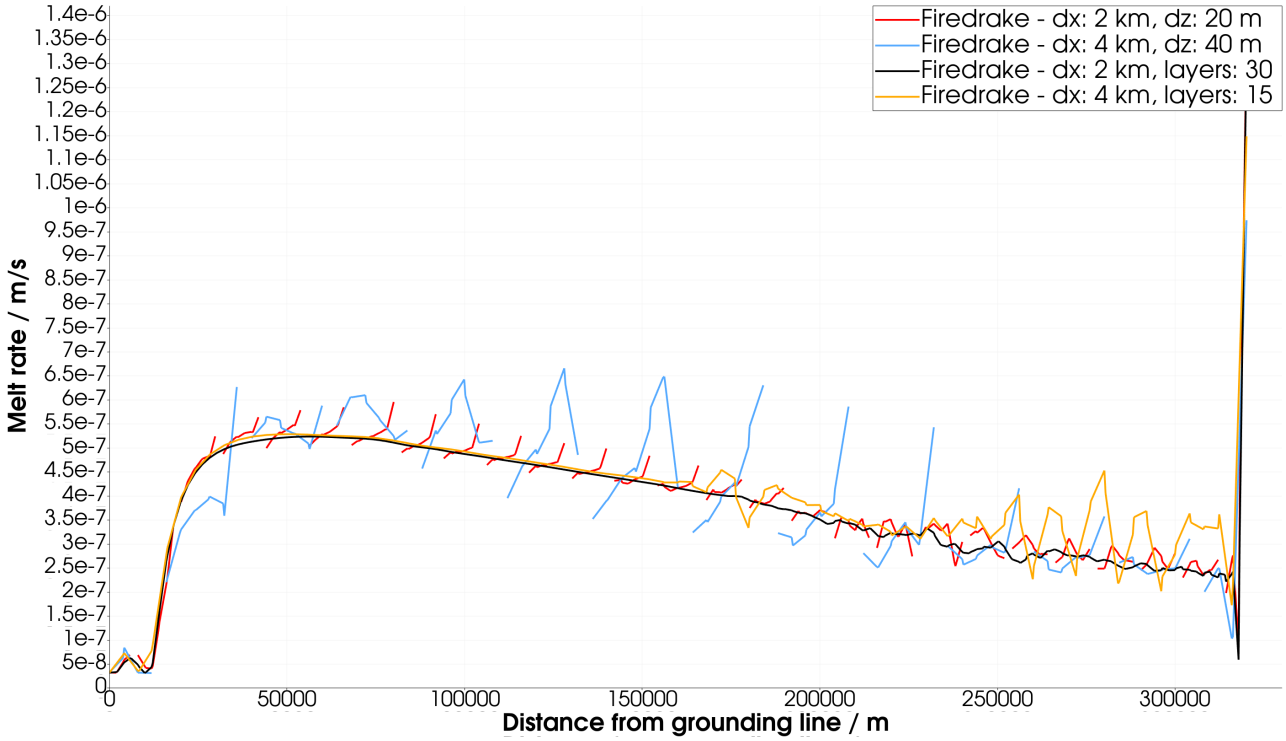


Figure 3.14: Melt rate profile for the 2D pre-ISOMIP+ comparison between z and σ-layer meshes using Firedrake after 50 days.

3.6 3D Benchmark vs MITgcm for the ISOMIP+ Ocean0 Test Case

The aim of this section is to demonstrate our ability to simulate ocean flow in a fully three-dimensional ice shelf cavity. The Ice Shelf Ocean Model Intercomparison Project+ (ISOMIP+) consists of a set of idealised ice shelf cavity geometries and parameters for a set of common experiments [Asay-Davis et al., 2016]. Here we use the *Ocean0* experiment, which has been used for a number of studies to introduce and compare different ocean models for ice shelf cavity applications [Zhou and Hattermann, 2020, Gwyther et al., 2020, Favier et al., 2019]. This experiment reaches a quasi-steady state in the cavity within a few months of simulation time, so it is a convenient test case for carrying out model comparisons and testing model parameter choices [Asay-Davis et al., 2016]. There is not an 'exact' solution that all models are aiming to replicate. For this reason, we have found it helpful to run simulations of MITgcm alongside our model.

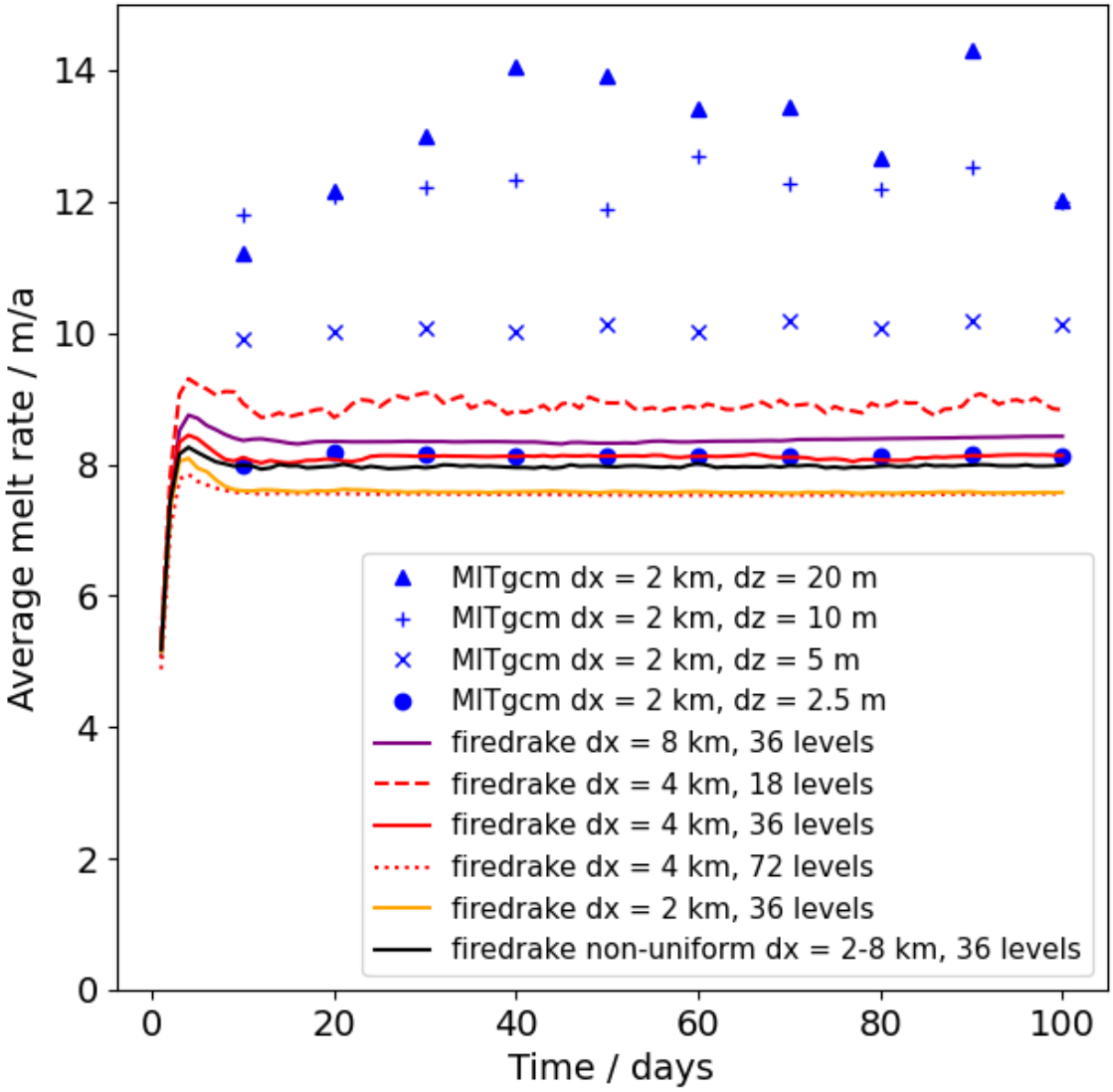


Figure 3.15: Integrated melt rate over deepest region, where ice draft is deeper than 300 m following Asay-Davis et al. [2016]. Melt rates are instantaneous values, not time averaged. MITgcm values are plotted every 10 days with blue symbols dependent on vertical resolution. Triangles - 20 m, Pluses - 10 m, Crosses - 5 m, Circles - 2.5 m. Firedrake values are plotted as lines, with colour dependent on horizontal resolution (and line style dependent on vertical resolution). Orange - 2 km, Red - 4 km (Dashed - 18 vertical levels outside the cavity, Continuous - 36 vertical levels outside the cavity, Dotted - 72 levels outside the cavity), Purple - 8 km, Black - 2 km in vicinity of grounding zone and western boundary coarsening to 8 km outside the cavity.

The *Ocean0* domain is approximately 400 km long, 80 km wide and has a maximum depth of 720 m. The bathymetry is specified by an analytical profile representing an idealised fjord, and the ice draft is from an ice flow simulation from the corresponding MISMIP+ experiment [Asay-Davis et al., 2016]. The target ISOMIP+ grid resolution is specified as 2 km in the horizontal and 36 vertical layers, with the caveat that different vertical meshing strategies will impose constraints on the resolution [Asay-Davis et al., 2016]. The ISOMIP+ protocol does not specify exactly how the cavity thickness should be defined as the 'real' cavity pinches to zero thickness at the grounding line. For MITgcm simulations we have followed the suggested 40 m minimum thickness for z-layer models, so that there are always at least two cells with the specified 20 m vertical resolution. To create the 3D Firedrake mesh we used the inbuilt 'extruded' mesh capability of Firedrake. First, we used QGIS [QGIS Development Team, 2022] to generate the horizontal extent of the mesh. We chose the grounding line by tracing the 10 m ocean thickness contour (ice draft - bathymetry). We used the *qmesh* package [Avdis et al., 2018] to generate the horizontal surface mesh that forms the base of the columns in the full 3D extruded mesh. *qmesh* is effectively a wrapper for Gmsh [Geuzaine and Remacle, 2009] to allow meshing of domains generated from GIS data. The surface mesh is then extended in the vertical direction to create columns of extruded triangular prism elements [Bercea et al., 2016, McRae et al., 2016]. We use 30 layers under the ice, with a transition to 36 layers in the open ocean region, with this transition occurring over a single horizontal grid cell. Squashing the domain under the ice gives a terrain following, quasi-sigma style mesh within the cavity, with higher vertical resolution towards the grounding line. We use the tensor product elements described in Section 2.3.2 for the discretisation of the velocity-pressure finite element pair, and for tracers the tensor product element described in Section 2.3.1.

The temperature and salinity profiles for initialisation and restoring on the northern boundary are the 'warm' profiles given in Table 6 from Asay-Davis et al. [2016]. We use a timestep of 900 s, with Backward Euler time stepping for the momentum equations and a Diagonally Implicit Runge-Kutta method, DIRK33, for the tracers. Horizontal and vertical kinematic viscosity are specified at 6 m^2/s and 1×10^{-3} m^2/s respectively, and horizontal and vertical diffusivity are specified at 1 m^2/s and 5×10^{-5} m^2/s as per the ISOMIP+ protocol [Asay-Davis et al., 2016].

We include the 'switch' to higher vertical mixing values when the stratification is unstable, however, the scheme has negligible effect on calculated melt rates due to the build up of stable stratification at the ice-ocean boundary during melting.

The melt parameterisation specified by ISOMIP+ does not include heat flux into the ice and uses constant turbulent exchange coefficients based on Jenkins et al. [2010]. For this comparison both models use the same turbulent exchange coefficients in the melt parameterisation, Γ_T: 0.011 and $\Gamma_S = \Gamma_T/35$. The vertical averaging scheme for melting is turned off in MITgcm [Losch, 2008]. Drag boundary conditions are applied on the top and bottom interfaces with a drag coefficient, $C_D = 2.5 \times 10^{-3}$.

3.6.1 Melt Sensitivity to Vertical Resolution

Figure 3.15 shows melt rates for the first 100 days of simulation integrated over the region where the ice draft is deeper than 300m, following Asay-Davis et al. [2016]. We have found that melt rates are very sensitive to the grid resolution. MITgcm melt rates are plotted in blue, with vertical resolution given by the symbol (triangles: 20 m (target ISOMIP+ resolution), pluses: 10 m, crosses: 5 m, circles: 2.5 m). Clearly, integrated melt rates calculated by MITgcm are highly dependent on vertical resolution, as refining the mesh three times leads to nearly a factor of two difference and there is no sign of convergence. Our model also shows vertical dependence of melt rate as evidenced by the red lines in Figure 3.15 (dashed: 18 levels, continuous: 36 levels, dotted: 72 levels), however, the spread is lower than with the z-layer model. These results are consistent with a recent study by Gwyther et al. [2020] using a range of models applied to the same ISOMIP+ *Ocean0* experiment and we will refer to their work later in the discussion.

Figure 3.16 shows plan view snapshots of melt rate at 50 days for MITgcm at different vertical resolutions, compared with Firedrake at the target resolution specified by ISOMIP+ for layered models (36 layers outside the cavity, 30 layers inside the cavity). As the vertical resolution of MITgcm increases the spatial melt rate pattern can be seen to approach that from Firedrake. Both models show highest melting at the grounding line, which is consistent with the depth dependent melting point, warm initial and restoring conditions at the bottom of the water

column, and steeper basal slopes close to the grounding line that enhance buoyancy driven flow. This melt pattern is also consistent with other published ISOMIP+ *Ocean0* results [Asay-Davis et al., 2016, Zhou and Hattermann, 2020, Gwyther et al., 2020].

The streaky, time evolving artefacts present in MITgcm at coarser vertical resolutions (circled in Figure 3.16) disappear as the grid is refined. Note that while time-mean fields of melt rate act to smooth out the 'streaks' there are still noisy grid scale features after time averaging (not shown). Note that reducing the vertical diffusivity and viscosity linearly at the same rate as the grid refinement does not bring the 'streaks' back. This suggests it is not a numerical instability of the type dependent on grid Reynolds number (the Reynolds number evaluated at the grid scale, $Re_{\Delta x} = U\Delta x/\nu$ where U is a characteristic velocity, Δx is the grid resolution and ν is the eddy viscosity) or grid Péclet number (similarly $Pe_{\Delta x} = U\Delta x/\kappa$ where κ is the eddy diffusivity). The streaks may therefore be a grid discretisation error caused by the step representation of the boundary that is inherent to z-layer models, even with a partial cell representation [Losch, 2008].

Within the cavity, both models show a clearly defined western (defined as positive y) boundary current, due to the effects of rotation. This can be seen in Figure 3.17, that is a cross section across the cavity at x = 480 km close to the grounding line. There is strong northward flow along the western boundary together with colder (and fresher) conditions, which results from melt water accumulating and preferentially flowing through this part of the domain. The middle panel of Figure 3.17 shows MITgcm at the higher vertical resolution. Refining MITgcm leads to better qualitative agreement with Firedrake. The return boundary flow and the stratification at the top of the domain is qualitatively much better resolved by MITgcm at the higher resolution. At this location the cavity thickness is close to 100 m, so with 30 vertical layers, Firedrake's vertical grid resolution is similar to MITgcm's vertical grid resolution of 2.5 m, so it is not unexpected that the higher resolution MITgcm simulation yields results much closer to those from Firedrake.

Another interesting feature of Figure 3.17 is the northward flow on the eastern boundary that is clearly visible in Firedrake and faintly visible in the finer resolution MITgcm simulation. This

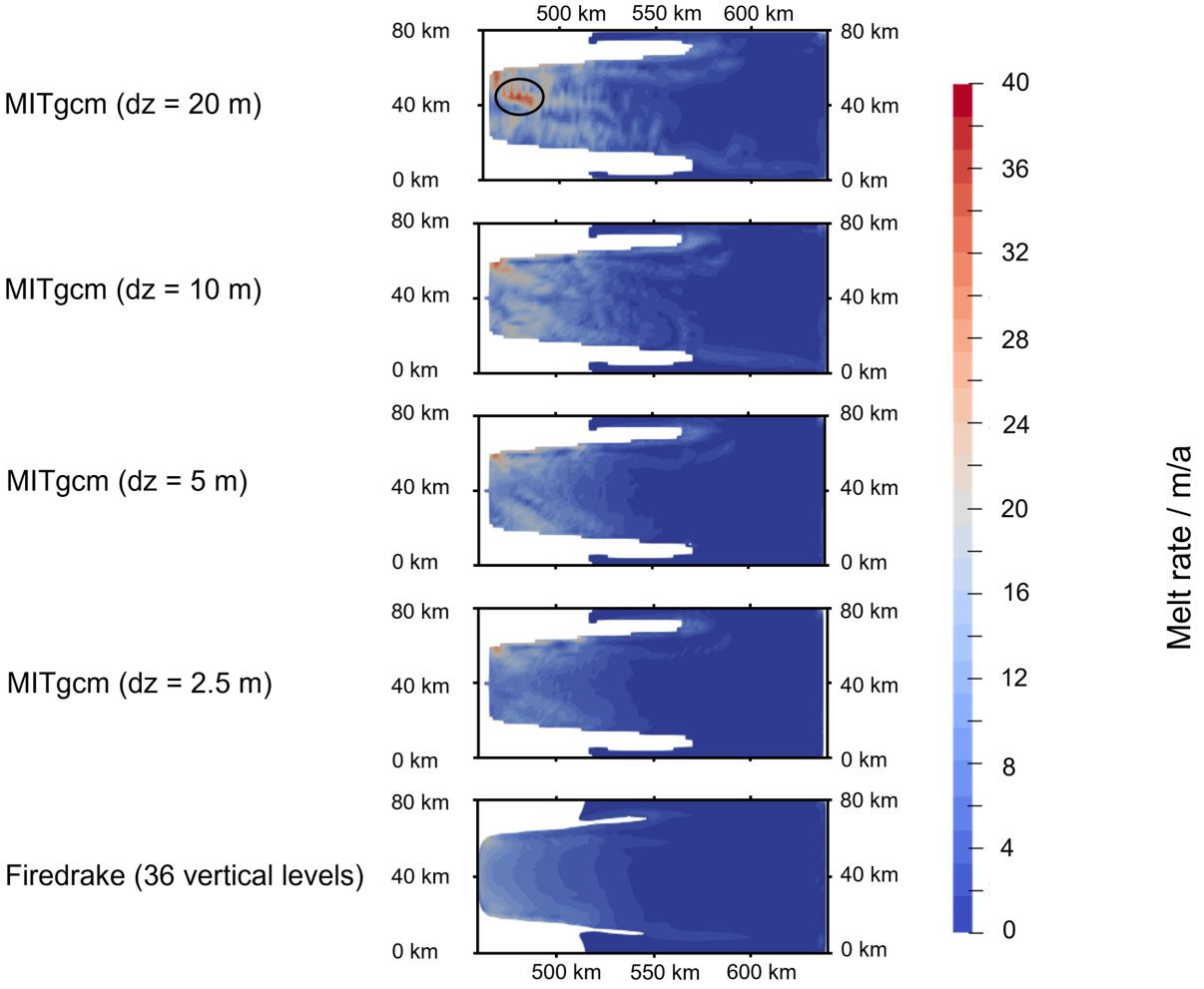


Figure 3.16: Plan view of basal melt at 50 days for MITgcm at target ISOMIP+ resolution for z-layer models (Δx: 2 km, Δz: 20 m), along with finer vertical resolution (Δz: 10 m, 5 m, 2.5 m) and Firedrake at target ISOMIP+ resolution for terrain following model (Δx: 2 km, 36 levels with 30 levels inside the ice shelf cavity). The circle in the top panel highlights a time varying, grid resolution dependent feature that is referred to in the text.

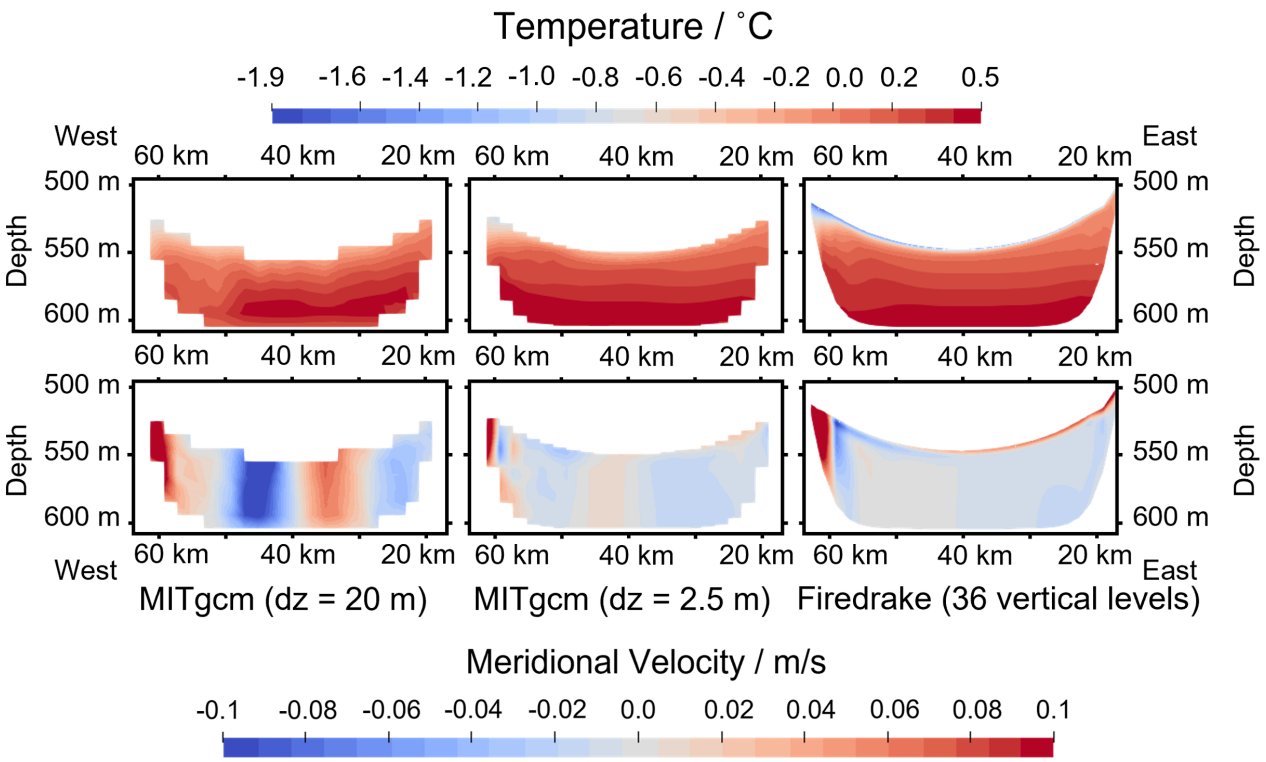


Figure 3.17: Cross section across ice shelf cavity at $x = 480$ km, close to the grounding line. Top panels show a snapshot of temperature at 50 days, bottom panels show a snapshot of meridional velocity at 50 days. Plots are stretched one hundred times in the vertical. From left to right the panels correspond to MITgcm at the target ISOMIP+ resolution for z-layer models ($\Delta x = 2$ km, $\Delta z = 20$ m), MITgcm fine ($\Delta x = 2$ km, $\Delta z = 2.5$ m), Firedrake ($\Delta x = 2$ km, 36 levels with 30 levels under the ice shelf). Note the cavity is about 100 m thick at this location, so the vertical size of the Firedrake layers is similar to MITgcm at the fine resolution shown in the middle panel. The alternating barotropic flow pattern, evident in the MITgcm meridional velocity at the ISOMIP+ target resolution (Δx: 2 km, Δz: 20 m), is reduced by plotting time averaged fields and is closer in magnitude to the simulation with finer vertical resolution. Time-mean fields of temperature are qualitatively very similar to snapshots after 50 days.

may be an Ekman layer feature based on recent progress in analytical solutions to cavity-like basins and could be a general feature of ice shelf cavities (Adrian Jenkins, personal communication at FRISP 2022). Since the ISOMIP+ domain has squashed side walls the Firedrake mesh with a fixed number of layers has much finer resolution near the boundary, so at this resolution Firedrake would be better able to resolve an Ekman layer.

In reality, melting leads to the injection of cold, and importantly, buoyant fresh water at the top of the domain. If the ice shelf base has a sufficient slope the buoyant melt water drives a shear flow and thus enhances mixing. However, melting can also lead to a negative feedback

mechanism since melt water forms a stratified layer, especially for low basal slope angles that effectively shuts down vertical mixing [McPhee, 2008, Vreugdenhil and Taylor, 2019]. This leads to reduced melt rates because the ice base is insulated from the warmer, saltier water below. It is the competition between the buoyancy driven shear and build up of stratification (as well as external sources, such as tidal currents) that controls mixing. Modelling vertical mixing accurately is therefore of fundamental importance to represent flow in ice shelf cavities and to calculate the associated melt rate. The total amount of mixing simulated by a numerical ocean model is a combination of resolved turbulence mixing, explicitly specified mixing that aims to account for the unresolved turbulent mixing (here we use a simple eddy diffusivity closure with constant diffusivity) and a grid size dependent implicit contribution, often referred to as numerical mixing. This spurious, numerical mixing is often dictated by the details of the discretisation of the advection and diffusion operators [Griffies et al., 2000]. Refining the grid resolution decreases the amount of spurious numerical mixing added by the model. In the context of ice shelf cavity modelling, reduced mixing can lead to increased stratification and in turn lower melt rates if the basal slopes are not too steep. Terrain following models, with typically higher resolution at the grounding zone resulting from the squashed vertical grids will have lower numerical mixing and thus resolve more of the stratification effects than a z-layer model run at typical coarse vertical resolutions, as seen in Figure 3.17. This means terrain following models in typical configurations tend to calculate lower melt rates. z-layer models, like MITgcm, can produce lower melt rates at higher resolution as shown in Figure 3.15. The specified vertical mixing has to be higher than the spurious mixing value for grid convergence of the melt rate, otherwise as the grid is refined the total amount of mixing generated by the model decreases, leading to a reduction in melt rates.

To further demonstrate the effects of stratification Figure 3.18 shows a vertical profile of temperature at x = 480 km, y = 40 km (midway through cross section of Figure 3.17). There is a clearly defined boundary layer for the Firedrake results (red) and the high resolution MITgcm simulation (blue), with strong stratification. However, the coarse MITgcm result (black) is nearly 1 °C warmer at the boundary compared to Firedrake and the vertical temperature gradient is much weaker. It is difficult to make a direct comparison, because the cavity thickness

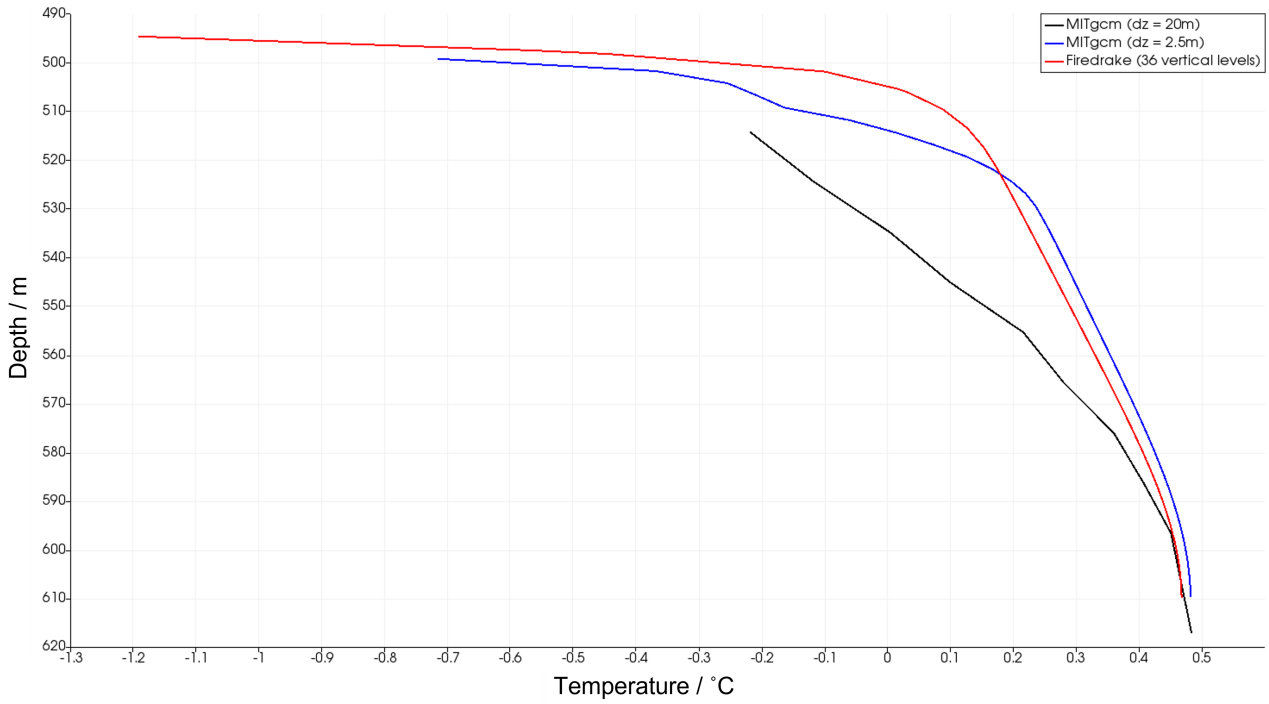


Figure 3.18: Depth profile of temperature at, x = 480 km, y = 40 km, which corresponds to the middle of Figure 3.17. Black: MITgcm (Δx = 2 km, Δz = 20 m), Blue: MITgcm (Δx = 2 km, Δz = 2.5 m), Red: Firedrake (Δx = 2 km, 36 levels).

is not exactly identical between MITgcm simulations due to the constraints of the z layer (and partial) cell representation of topography. To help with visualisation, Figure 3.19 show the meshed Firedrake domain and the mask of the only ocean cells for the MITgcm domain at two vertical resolutions.

Figure 3.20 shows the same depth profile but for three simulations using Firedrake with 18, 36 and 72 levels (15, 30 and 60 levels within the cavity). The meshing strategy used for Firedrake ensures the domain geometry is the same at each resolution and crucially, this means that the melt parameterisation samples the boundary temperature and salinity values at the same physical location as the grid is refined. Away from the ice-ocean interface all three simulations are in very good agreement. Close to the boundary the agreement is still very close relative to the MITgcm comparison of Figure 3.18, however, close inspection reveals that the stratification is strongest for the simulation with the highest number of layers as expected. This leads to a lower value of temperature (and salinity) at the boundary which when fed into the melt parameterisation returns weaker melt rates.

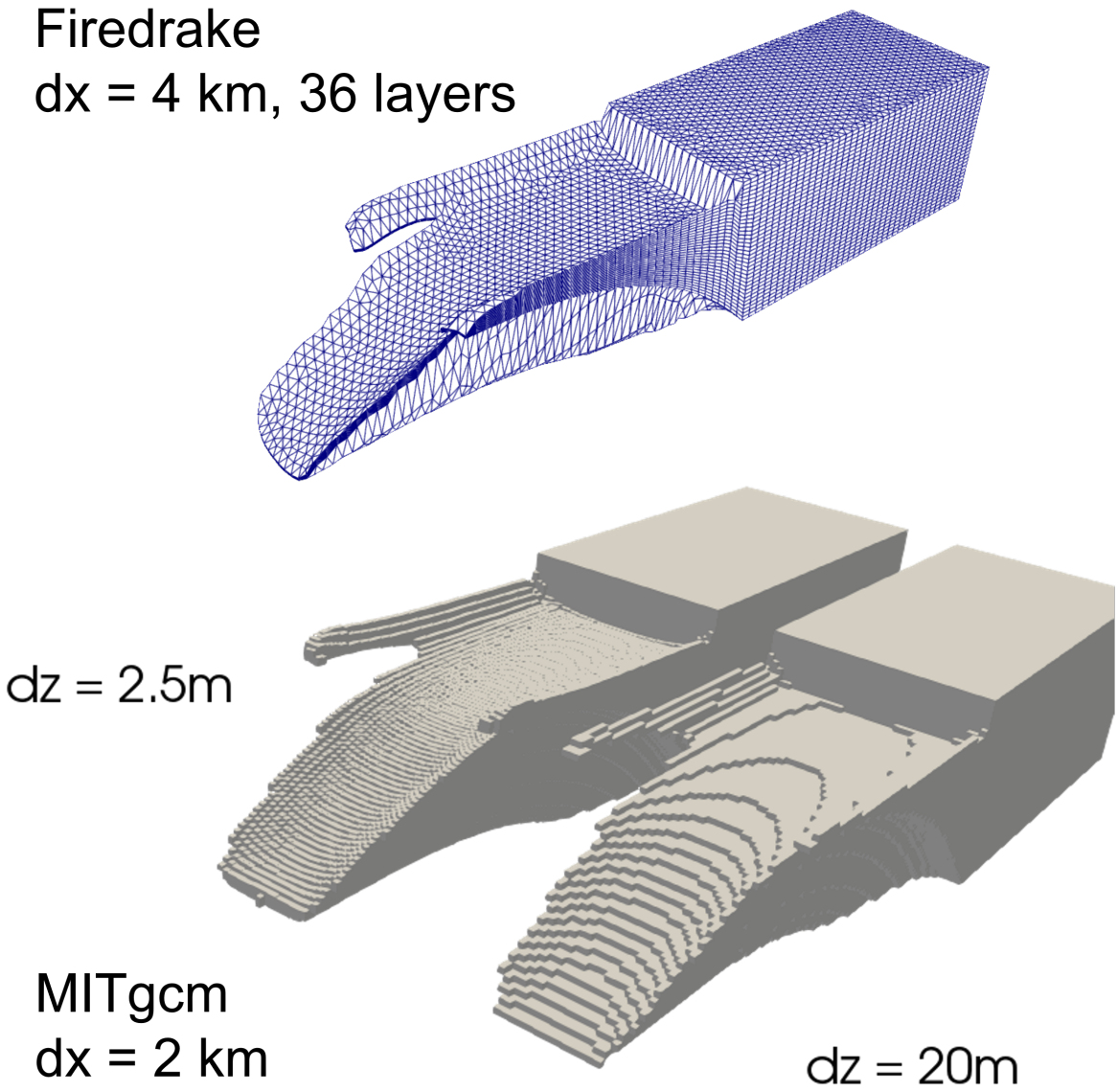


Figure 3.19: Comparison of 3D ISOMIP+ meshes. Top: 3D view of Firedrake σ-layer mesh with $\Delta x = 4$ km and 36 layers (30 layers inside the ice shelf cavity). Bottom: 3D view of MITgcm ocean domain after masking showing difference in domain shape with $\Delta z = 2.5$ m and $\Delta z = 20$ m (ISOMIP+ default) and horizontal resolution

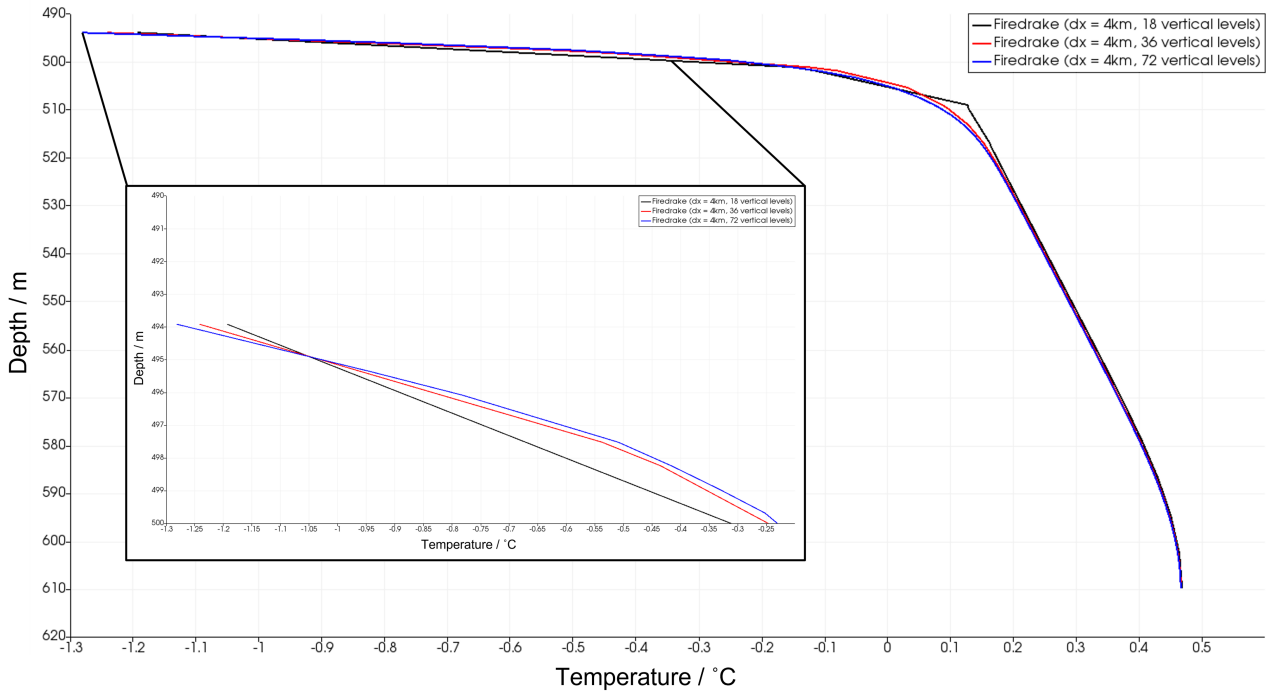


Figure 3.20: Depth profile of temperature at, x = 480 km, y = 40 km, which corresponds to the middle of Figure 3.17. Black: Firedrake (Δx = 4 km, 18 levels), Red: Firedrake (Δx = 4 km, 36 levels), Blue: Firedrake (Δx = 4 km, 72 levels). The inset magnifies the region close to the boundary between 490 m and 500 m depth.

The dependence of melt rate on vertical resolution has been investigated previously in a separate study also using the *Ocean0* experiment [Gwyther et al., 2020]. Gwyther et al. [2020] used three ocean models with different vertical coordinate systems (sigma: ROMS, z: COCO, hybrid: MPAS-O) and showed that the two models with constant, 20 m vertical resolution at the boundary specified by ISOMIP+ (COCO and MPAS-O) systematically calculate higher melt rates than the sigma-coordinate based model, which is consistent with the results presented here. Gwyther et al. [2020], following the ISOMIP+ protocol, framed the problem as a matter of tuning the turbulent exchange coefficients from the melt parameterisation to obtain the same melt rates between models. The ISOMIP+ protocol suggests a target melt rate averaging 30 m/a in the grounding zone region (where the ice draft is below 300 m depth). Gwyther et al. [2020] showed that the sigma model ROMS was incapable of reaching this target melt rate purely by adjusting the exchange coefficients. Here we take a different approach and frame the problem in terms of grid convergence, with the aim of achieving a melt rate that is unaffected by grid resolution and keeps the exchange coefficients fixed. Given the current implementation

of the melt parameterisation and turbulence closure (constant eddy diffusivity and viscosity) this means we need the temperature, salinity and velocity fields to remain constant at the boundary as the grid is refined to achieve a fixed melt rate.

Gwyther et al. [2020] show that sigma-coordinate based models can produce higher melt rates with elevated vertical diffusivity at the boundary. Figure 3.21 shows integrated melt rates for Firedrake run at different vertical resolutions and vertical diffusivities. The lines in red are the same as from Figure 3.15, which is the integrated melt rate at the target ISOMIP+ vertical diffusivity of 5×10^{-5} m^2/s. With this diffusivity the melt rate still depends on grid resolution as discussed. The blue and orange lines in Figure 3.21 show the melt rate with diffusivities of 5×10^{-6} m^2/s and 5×10^{-7} m^2/s respectively. Despite the order of magnitude change, there is no significant difference in melt rate. This suggests that numerical mixing dominates when the specified mixing is specified at these low levels. As a result, the melt rate would be expected to be highly dependent on grid resolution and we expect refining the vertical resolution further would reduce melt rates. In contrast, by increasing the vertical diffusivity by ten times the amount of the ISOMIP+ specification to 5×10^{-4} m^2/s, shown by the black lines in Figure 3.21, the integrated melt rate calculated with 72 layers is very close to the melt rate calculated with 36 layers (the specified number of layers for the ISOMIP+ profile). Since the melt rate appears to be converging towards a consistent value as the grid is refined, the specified diffusivity must be dominating over the spurious, resolution dependent, numerical mixing. This comparison suggests that a rough estimate for the numerical mixing for a Firedrake simulation with the ISOMIP+ target of 36 layers is likely between 5×10^{-6} m^2/s and 5×10^{-5} m^2/s. Since the upper bound has the same value as specified in the ISOMIP+ protocol, this serves to highlight the importance of running ice-ocean simulations at multiple resolutions to determine how sensitive the melt rate is to resolution for a particular model and setup configuration.

We also ran simulations with increased refinement at the ice-ocean interface based on a ROMS style squashing function [Song and Haidvogel, 1994]. Although, this does ensure high resolution at the ice-ocean boundary one has to be careful interpreting that the melt rate has converged based on varying the squashing parameter. The grounding zone region is the most important location to refine the mesh because this is where the melt is highest, yet this is also where

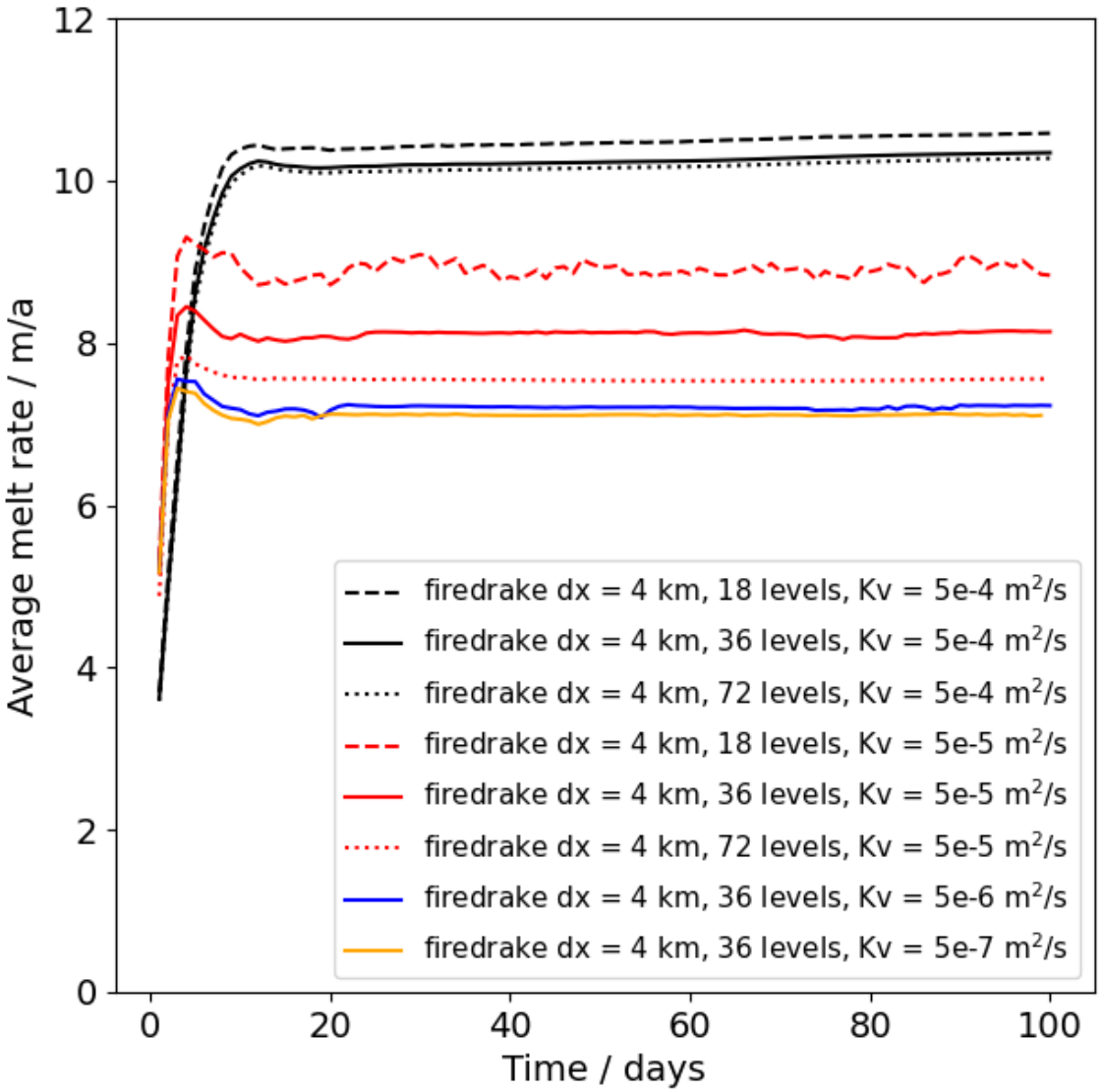


Figure 3.21: Integrated melt rate with Firedrake run at different vertical resolutions and vertical diffusivity values. The melt rate is integrated over the deepest region, where ice draft is deeper than 300 m following Asay-Davis et al. [2016]. As in Figure 3.15 the line style indicates vertical resolution: Dashed - 18 vertical levels outside the cavity, Continuous - 36 vertical levels outside the cavity (target ISOMIP+ resolution), Dotted - 72 levels outside the cavity. The colour of the lines indicates the vertical diffusivity: Black - Kv $= 5 \times 10^{-4}$ m^2/s, Red - Kv $= 5 \times 10^{-5}$ m^2/s (target ISOMIP+ diffusivity), Blue - Kv $= 5 \times 10^{-6}$ m^2/s, Orange - Kv $= 5 \times 10^{-7}$ m^2/s.

squashing functions have the least effect because the layered mesh has already been squashed due to the thin ocean thickness. The melt rates can appear to be 'converged' as the stretching factor is increased, when in actual fact the grid resolution is hardly changing.

The implicit assumption made here is that the diffusivity and viscosity do not vary close to the boundary. This is not necessarily physically justifiable, as Monin-Obukhov theory implies that with strong melting and insufficient shear turbulence to break down the stratification, the mixing rate (and hence the values of diffusivity and viscosity) will be much lower, leading to overestimated melt rates [Vreugdenhil and Taylor, 2019, McPhee, 2008]. We have assumed fixed diffusivity and viscosity primarily for reasons of simplicity: it is the ISOMIP+ specification and with a new model we wanted to have confidence that we could understand the general behaviour before implementing more complicated turbulence closure schemes. Even if the model is responsible for explicitly calculating diffusivity and viscosity, the pertinent turbulent mixing length scales close to the boundary are very small, sub metre scale [McPhee, 2008], so it may be impractical to account for these effects in large scale ocean simulations. Potentially the melt parameterisation could be modified to incorporate the enhanced stratification effects that arise during melting (similar to the transition of scalar properties across the viscous boundary later, which is accounted for; Holland and Jenkins, 1999), but this leaves the question of where to sample T_w and S_w open. Unless the sampling distance is chosen for physical reasons it seems like it will inherently depend on the grid resolution. It is not obvious how applicable an ice shelf ocean cavity simulation will be for different forcing scenarios or geometrical configurations if the exchange coefficients have been tuned specifically to compensate biases arising from a somewhat arbitrary choice of grid resolution.

Overall, our integrated melt rate of 7.6 m/a for the target ISOMIP+ resolution of 2 km horizontally and 36 vertical layers (orange line Figure 3.15) is consistent with, and within the spread of, other published results when using the same turbulent exchange coefficients in the melt parameterisation (Γ_T: 0.011 and $\Gamma_S = \Gamma_T/35$). Published results for terrain following models with 36 layers include FVCOM: 10 m/a (Figure 9 from Zhou and Hattermann, 2020) and ROMS: 4 m/a (Figure 4 from Gwyther et al., 2020). Similarly, examples of z-level models at high vertical resolution ($\Delta z = 2$ m) include COCO: 5 m/a and MPAS-O: 9 m/a (Figure 5

from Gwyther et al., 2020), as well as POP2x: 7 m/a (Figure 9 from Asay-Davis et al., 2016). Also note the MITgcm integrated melt found here for this comparison is consistent with COCO and MPAS-O, which both gave around 15 m/a with 20 m vertical resolution (Figure 5 from Gwyther et al., 2020).

3.6.2 Melt Sensitivity to Horizontal Resolution

We also investigated melt rate sensitivity to horizontal resolution within the cavity. We ran Firedrake at two coarser horizontal resolutions, additionally shown in Figure 3.15 (purple: 8 km, red: 4 km), to compare with the target ISOMIP+ resolution (orange: 2 km). The plan view of each of horizontal base mesh is shown in Figure 3.22 and the corresponding plan view of melt rates in Figure Figure 3.23. Although the spatial patterns of melt rate are still broadly equivalent, with highest melting at the grounding line seen in all simulations, the integrated melt rates have not converged to a consistent result.

To investigate the potential of a horizontally unstructured mesh to focus resolution we ran a preliminary simulation with the target ISOMIP+ 2 km resolution only along the western boundary and the grounding zone region, relaxing to 8 km outside the cavity. These areas were chosen to best capture the observed spatial melt pattern, with highest melting at the grounding line, as well as the fast-flowing western boundary current present due to Coriolis. The integrated melt rate for this non-uniform mesh is shown by the black line in Figure 3.15. A positive result is that the melt rate predicted with the non-uniform mesh is closer to the value from the uniform 2 km mesh than the 8 km, and indeed the 4 km, uniform mesh. However, considering the Degrees of Freedom (DOFs) associated with each mesh this may not be surprising. The number of combined velocity, pressure, temperature and salinity DOFs is 964,059 for the 8 km mesh, 3,448,681 for the 4km mesh and 13,693,896 for the 2 km mesh. The non-uniform mesh has 5,011,517 DOFs, which is actually 1.45 times more DOFs than the 4 km mesh. In more complicated cavity geometries, the ability to vary the mesh resolution flexibly, especially in the presence of large channels and crevasses, may prove crucial [Zhou and Hattermann, 2020].

Figure 3.24 shows the melt rates for the different vertical (Δz = 20 m to 2.5 m) and horizontal

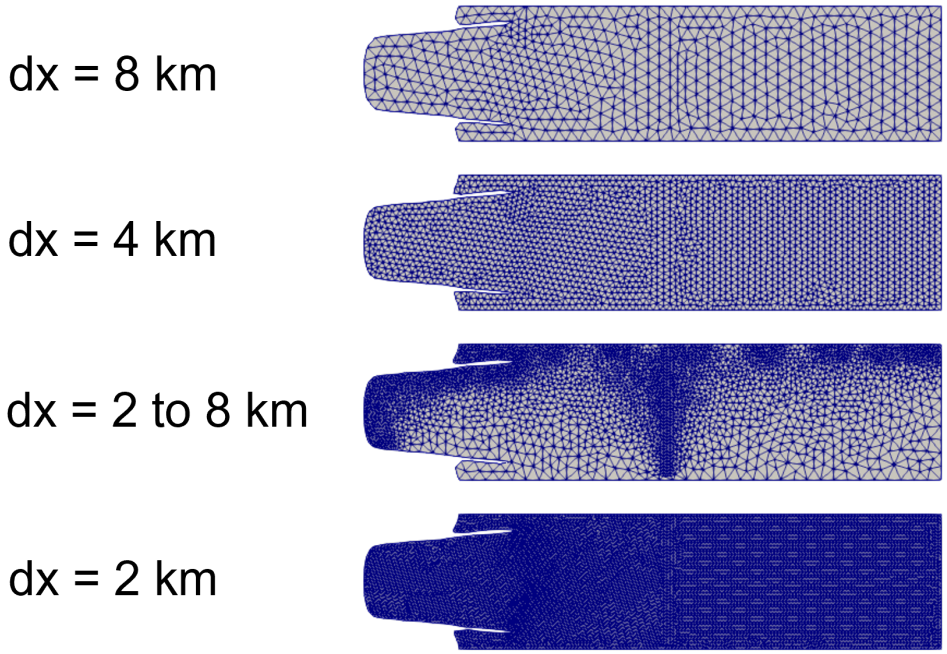


Figure 3.22: Plan view of base meshes used to investigate sensitivity to horizontal resolution. From top to bottom $\Delta x = 8$ km, 4 km, non-uniform (2 km to 8 km) and 2 km (target ISOMIP+).

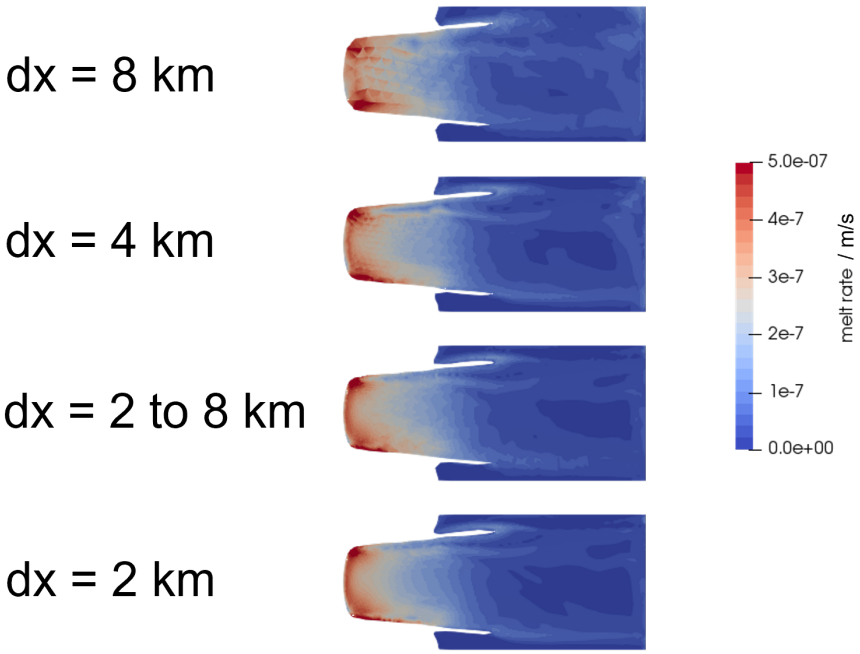


Figure 3.23: Plan view of basal melt rates for different horizontal grids. From top to bottom Δx = 8 km, 4 km, non-uniform (2 km to 8 km) and 2 km (target ISOMIP+). The units of melt rate are shown in m/s. Note that 5 $\times 10^{-7}$ m/s is roughly equivalent to 15 m/a.

(Δx = 4 km to 1 km) MITgcm runs. Note that the melt rate plotted is an average for the entire ice shelf and not just the deepest 300 m portion of the cavity as for Figures 3.15 and 3.21, so the absolute values of integrated melt rate are lower. For coarse vertical resolution (Δz = 20 m or 10 m) the integrated melt rates do not seem to have reached a steady state by 100 days, so it is difficult to detect changes due to horizontal resolution. For finer vertical vertical grid resolution, notably for Δz = 2.5 m, the dependence on horizontal grid size is actually in the opposite direction to Firedrake, i.e. refining the horizontal grid cell size for MITgcm leads to the integrated melt rate increasing, though it has still not obviously converged.

Figure 3.25 shows the melt plan view MITgcm run with Δz = 2.5 m and Δx = 4 km, 2 km, 1 km compared with the target ISOMIP+ resolution for Firedrake (Δx = 2 km, 36 layers). Qualitatively the melt pattern, especially near the grounding line, looks most similar to Firedrake for the 1 km horizontal resolution MITgcm simulation. There are differences along the western boundary, however, where MITgcm has a clearly defined high melt feature along y $\sim$ 59 km. There is also a higher patch of melt rate near the limb beginning of the western 'limb' of the ice shelf near x = 575 km. The magnitude of the melt rate feature increases with horizontal resolution, thus at least qualitatively explaining the integrated melt grid dependence in Figure 3.24.

For the MITgcm simulation with Δz = 2.5 m the minimum ocean thickness near the edge of the domain will be 5 m (two vertical grid cells). In the Firedrake domain these regions have a thickness of 10 m, which means that the vertical mesh resolution will be $\sim$ 0.33 m, almost ten times finer than the corresponding MITgcm resolution. It is likely that this leads to much stronger vertical tracer stratification for Firedrake in these regions than MITgcm and thus leads to lower melt rates as discussed earlier in this Chapter. Ultimately, this highlights that in a complicated 3D domain, which is still idealised compared with real ice shelf geometry, careful choice and testing of different mesh resolutions and meshing strategies is crucial for obtaining an accurate melt rate.

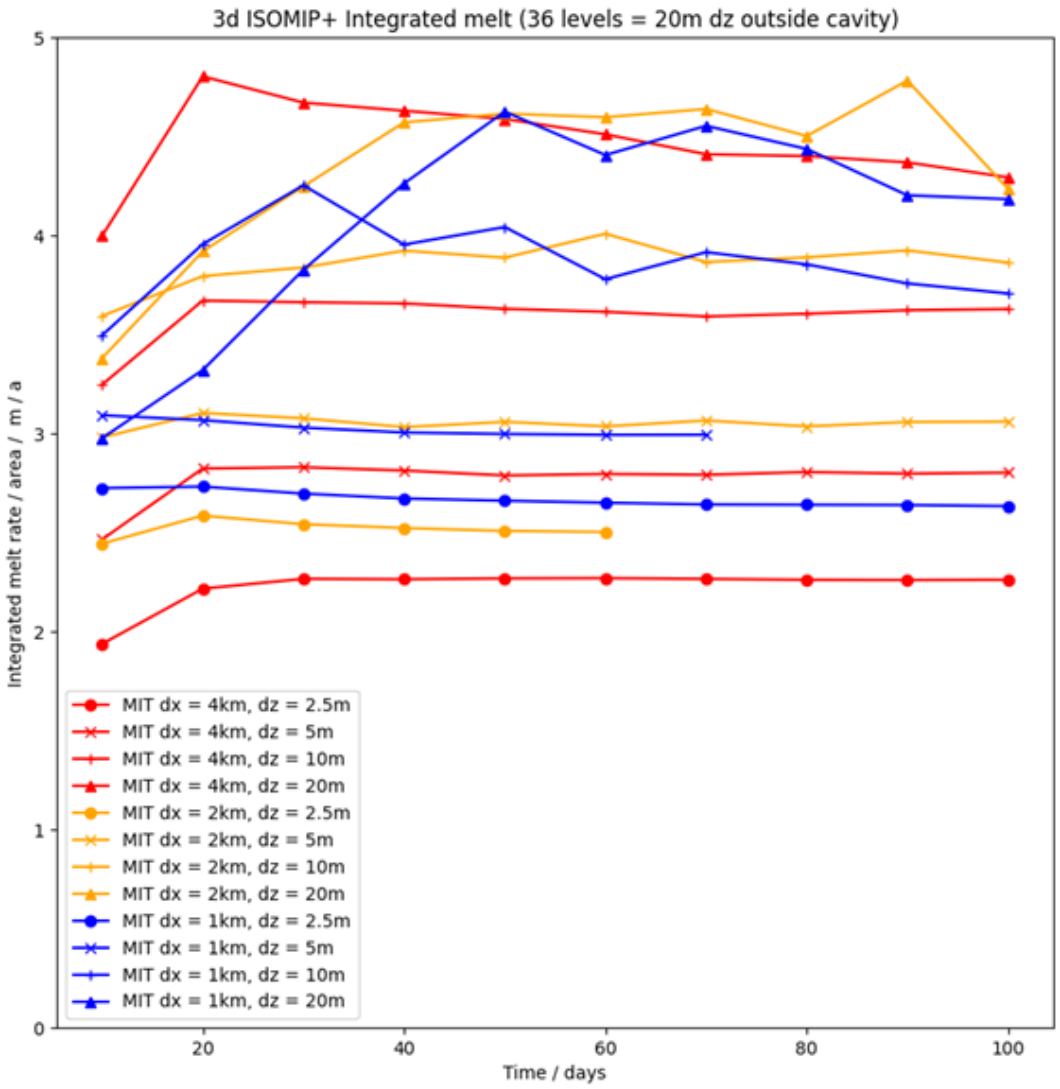


Figure 3.24: Integrated melt rate with MITgcm run at different vertical resolutions and horizontal resolutions. Note the melt rate plotted is different to Figures 3.15 and 3.21. The melt rate plotted is the average for the entire ice shelf (not just the 300 m deepest region). Colour indicates horizontal resolution: red is $\Delta x = 4$ km, orange is $\Delta x = 2$ km (default ISOMIP+) and blue is $\Delta x = 1$ km. The type of line marker shape indicates vertical resolution: triangle is $\Delta z = 20$ m (target ISOMIP+ for z-layer models), tick is $\Delta z = 10$ m, cross is $\Delta z = 5$ m and circle is $\Delta z = 2.5$ m.

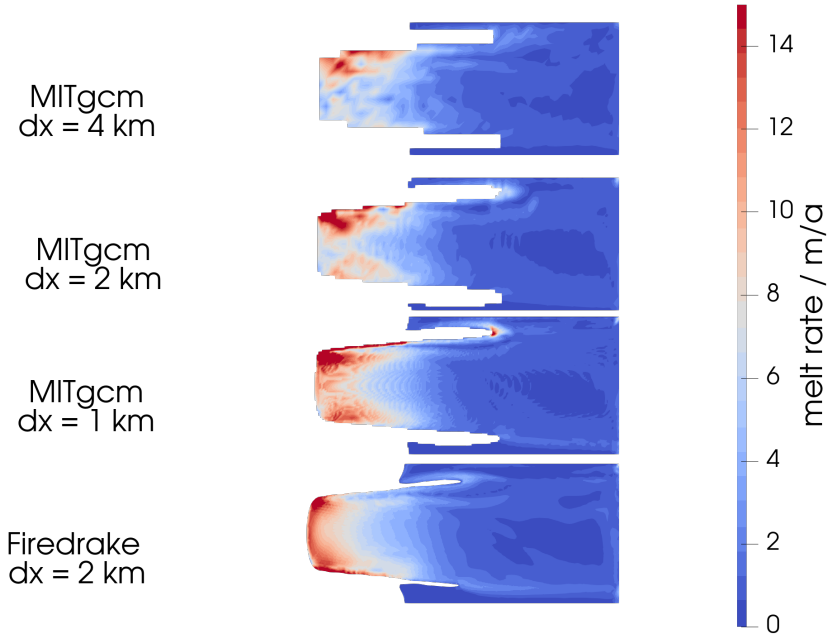


Figure 3.25: Plan view melt rate after 50 days for MITgcm run with $\Delta z = 2.5$ m and varying horizontal resolution, $\Delta x = 4$ km, 2 km, 1 km and Firedrake $\Delta x = 2$ km, 36 layers (ISOMIP+ default).

Parameter	Value	Description
L	$\sim$ 400 km	Domain length
W	$\sim$ 80 km	Domain width
D	720 m	Maximum depth
$\Delta x, \Delta y$	2 km	Default horizontal grid size
nz	36 (30) layers	Default (under ice) vertical grid for Firedrake
Δz	20 m	Default vertical grid size for MITgcm
$h_{\sigma\min}$	10 m	Minimum ocean thickness for Firedrake
$h_{z\min}$	40 m	Minimum ocean thickness for MITgcm
Δt	900 s	Timestep
μ_h	6 m^2 s^{-1}	Horizontal viscosity
μ_v	1×10^{-3} m^2 s^{-1}	Vertical viscosity (ISOMIP+ switch)
κ_h	1 m^2 s^{-1}	Horizontal diffusivity
κ_v	5×10^{-5} m^2 s^{-1}	Default vertical diffusivity (ISOMIP+ switch)

Table 3.1: Parameters and constants used for the ISOMIP+ *Ocean0* test case in Section 3.6.

Section	Model	Δx (km)	Δz	Δt (s)	No. of cores	Wall time (hours)
3.2	MITgcm	0.05	z: 1 m	30	10*	2.2
3.2	Firedrake	0.05	z: 1 m	30	8*	69.2
3.4	MITgcm	2	z: 20 m	900	1*	0.13
3.4	MITgcm	1	z: 10 m	450	1*	1.5
3.4 / 3.5	Firedrake	4	z: 40 m	1800	1*	0.83
3.4 / 3.5	Firedrake	2	z: 20 m	900	1*	3.3
3.5	Firedrake	4	σ: 18 layers	1800	1*	1.1
3.5	Firedrake	2	σ: 36 layers	900	1*	5

Table 3.2: Performance benchmarking vs MITgcm for 2D simulations. * indicates that the simulation was run on a desktop workstation. This is an Intel(R) Core(TM) i9-7900X CPU @ 3.30GHz with 10 cores.

3.7 Performance Benchmarking vs MITgcm

This section provides preliminary timing results for the 2D and 3D simulations presented in this Chapter. Table 3.2 gives the wall clock time for the 2D test cases for MITgcm and Firedrake. In 2D Firedrake seems to be consistently ~ 25 times slower compared with MITgcm when run at the same resolution and timestep, for instance comparing rows 1 and 2 or comparing rows 3 and 6. This is not surprising given that at the same resolution Firedrake contains more degrees of freedom per cell than MITgcm. Arguably, the melt rate calculated by Firedrake's 4 km by 40 m simulation (Row 5 of Table 3.2) is as accurate and within the spread of MITgcm's 1 km by 10 m (Row 4 of Table 3.2) result as shown by Figure 3.12. Comparing these simulations Firedrake is actually almost twice as fast as MITgcm. Firedrake's σ-layer runs seem to be about 1.5 times slower than the corresponding z-layer simulation with the same horizontal resolution.

Table 3.3 gives the timings for the 3D ISOMIP+ runs for MITgcm and Firedrake. Comparing exact run times is slightly difficult as the simulations were run on two different computers and across a number of different cores. For instance, rows 11 and 12 have the same resolution and timestep (2 km, 20 m, 900 s) but the first simulation is carried out on a desktop workstation: an Intel(R) Core(TM) i9-7900X CPU @ 3.30GHz on 8 cores, while the second is carried out on one node of Imperial College's HPC with 32 cores. Since the second simulation has four times the number of cores one might expect the simulation to be four times faster, when in fact it is only about twice as fast. Although this could be that the problem size is too small to scale well it seems unlikely because there are 3,448,681 degrees of freedom and a conservative rule of

thumb is that Firedrake should continue to scale well for problems with roughly 50,000 degrees of freedom per core.

Comparing the target ISOMIP+ runs of MITgcm (row 1) with Firedrake (row 14), Firedrake seems to be much slower, almost by a factor of 4300 (or 2100 if Imperial's HPC is slower than a desktop workstation). However, given that that the MITgcm results with 1 km horizontal resolution and 2.5 m vertical resolution (row 8) are closer to the target ISOMIP+ Firedrake run (row 14), perhaps a fairer and definitely more flattering comparison is that Firedrake is 15 (or 7.5 – see above) times slower than MITgcm. As the resolution in Firedrake increases in the horizontal the slowdown is mostly as expected. From 8 km (row 9) to 4 km (row 11) the slowdown is a factor of 2.7, so actually less than the expected 4 times. From 4 km to 2 km the slowdown is a factor of 4.8 so slightly worse than the theoretical four times. In the vertical, refining to 72 layers (row 13) from 36 layers (row 12) slows the model by a factor of 2.5 so again slightly worse than the expected two times slower. Interestingly, the non-uniform mesh (row 15) contains 2.5 times less degrees of freedom than the 2 km model but is actually about 3.6 times faster.

Unfortunately, the singularity container used to run Firedrake on Imperial College's HPC did not work across multiple nodes, so future work would extend this to investigate how the solvers scale to many more cores. It is also worth emphasising again that the ISOMIP+ test case is not really the type of ocean modelling this model is designed for. It is a fairly structured domain so being able to employ an unstructured mesh is not such an advantage, whereas for real ice shelf cavities the complex geometry of grounding zones and basal crevasses (see Chapter 5) may prove that a fully flexible unstructured mesh is essential.

3.8 Code Availability

The code is available at https://github.com/thwaitesproject/thwaites.git.

Run scripts for 2D and 2.5D benchmark in idealised grounding zone domain (Sections 3.2 and 3.3):

Section	Model	Δx (km)	Δz	Δt (s)	No. of cores	Wall time (hours)
3.6	MITgcm	2	z: 20 m	900	1*	4.5
3.6	MITgcm	2	z: 10 m	900	1*	11.3
3.6	MITgcm	2	z: 5 m	450	4*	8.8
3.6	MITgcm	2	z: 2.5 m	450	4*	23.0
3.6	MITgcm	1	z: 20 m	450	1*	58.1
3.6	MITgcm	1	z: 10 m	450	4*	49.0
3.6	MITgcm	1	z: 5 m	450	4*	87.7
3.6	MITgcm	1	z: 2.5 m	225	4*	314.1
3.6	Firedrake	8	σ: 36 layers	900	8*	88
3.6	Firedrake	4	σ: 18 layers	900	32^+	<72
3.6	Firedrake	4	σ: 36 layers	900	8*	239
3.6	Firedrake	4	σ: 36 layers	900	32^+	125
3.6	Firedrake	4	σ: 72 layers	900	32^+	316
3.6	Firedrake	2	σ: 36 layers	900	32^+	606
3.6	Firedrake	2 - 8	σ: 36 layers	900	32^+	170

Table 3.3: Performance benchmarking vs MITgcm for 3D simulations. * indicates that the simulation was run on a desktop workstation. This is an Intel(R) Core(TM) i9-7900X CPU @ 3.30GHz with 10 cores. $^+$ indicates that the simulation was run on a single node of the HPC cluster at Imperial College provided by the Research Computing Service.

- thwaites/run_scripts_testing/ice_shelf_cavity/ice_shelf_cavity_structured_z_closed_HJ99gamma_ufric_iterative.py
- thwaites/run_scripts_testing/ice_shelf_cavity/ice_shelf_cavity_structured_z_closed_HJ99gamma_ufric_iterative_with_coriolis.py
- MITgcm setup files can be found in the folders 'run_dt30s_gamma_fric_snapshot_50days_1hour_output' and 'run_dt30s_gamma_fric_snapshot_50days_1hour_output_2.5d' in this Zenodo directory: https://doi.org/10.5281/zenodo.7863601
- MITgcm build files for all runs are in the folder MITgcmbuild in this Zenodo directory: https://doi.org/10.5281/zenodo.7584700 along with a README file

Run scripts for 2D ISOMIP-like domain (Sections 3.4 and 3.5):

- thwaites/run_scripts_testing/ice_shelf_cavity/isomip_plus/isomip_plus_coarse_2d_extruded.py
- thwaites/run_scripts_testing/ice_shelf_cavity/isomip_plus/isomip_plus_coarse_2d_extruded_layered.py

- MITgcm setup files can be found in the folder 'run_isomip_2d_resz_dx2kmz_dz20mz_stratified' in this Zenodo directory: https://doi.org/10.5281/zenodo.7863601
- MITgcm build files for all runs are in the folder MITgcmbuild in this Zenodo directory: https://doi.org/10.5281/zenodo.7584700along with a README file

Run scripts for 3D ISOMIP+ test case (Section 3.6):

- thwaites/run_scripts_testing/ice_shelf_cavity/isomip_plus/isomip_plus_3d_extruded_layered.py
- MITgcm setup files can be found in the folder 'MITgcm_run_isomip_2km.zip' in this Zenodo directory: https://doi.org/10.5281/zenodo.7584700
- MITgcm build files for all runs are in the folder MITgcmbuild in this Zenodo directory: https://doi.org/10.5281/zenodo.7584700 along with a README file

3.9 Conclusion

This chapter compared our Firedrake based model with MITgcm for a range of scenarios from a small scale 2D grounding zone domain to a full 3D representation of an entire ice shelf cavity. The agreement in 2D for the idealised overturning 'ice pump' flow proved to be very good, with Firedrake displaying better robustness and accuracy compared with MITgcm. In fact, MITgcm exhibited spurious grid scale features that were very obvious in the vertical velocity field. The model comparison after adding rotational effects for the 2.5D formulation proved equally good. We also compared Firedrake with MITgcm for an extended 'ISOMIP-like' 2D domain, this time using the tensor product discretisation developed in Chapter 2 for large scale domains. The agreement between MITgcm and Firedrake was again very consistent. Due to the flexible nature of our model formulation, we were able to directly compare the effects of having a vertical meshing strategy with constant vertical resolution, z, compared with a mesh comprised of a fixed number of vertical layers representing a σ-layer mesh with the same underlying equations being solved. Since the agreement in the calculated melt rate between

the two methods was very similar we progressed to 3D with only the σ-layer meshing strategy to avoid the 'stripy' melt rate pattern associated with z-layer models.

The 3D ISOMIP+ study highlighted the importance of modelling mixing processes accurately, because the melt rate parameterisation is so sensitive to vertical grid resolution and diffusivity, in accordance with the literature. This is primarily due to the build up of stratification at the ice-ocean interface that acts as a negative feedback on melt rates when the basal slope is not too steep and the resulting buoyancy driven flow is not too dynamic. Our default modelled melt rate was within the range of other published σ-layer results giving confidence to our model implementation. However, it was much harder to initially obtain consistent melt rates between MITgcm and Firedrake given the default options suggested by the ISOMIP+ protocol than for the 2D experiments. By refining MITgcm in the vertical we were able to obtain much better qualitative spatial agreement of the melt fields between the two models, though neither model showed signs of melt rate convergence. By comparing Firedrake simulations with a different number of layers and vertical diffusivities we showed that with the default ISOMIP+ parameters the magnitude of the spurious mixing in Firedrake was of a similar order to the specified diffusivity. With an order of magnitude higher explicit diffusivity we were able to achieve a converged melt rate. The melt rate dependence on horizontal grid resolution was not as striking as the vertical grid resolution dependence, but still present. This seems to be linked to the representation of the ocean column near the grounding line on the side walls, highlighting the importance of using accurate and consistent meshing strategies when investigating ice shelf cavities with complex geometries. Uncertainties in the melt parameterisation in its current form, especially regarding the representation of stratification at the ice-ocean boundary, mean that finding a grid independent melt rate is not necessarily a physically meaningful result. However, if melt rate simulations are tuned to compensate biases from a particular grid resolution it is not obvious how applicable the simulations will be to different forcing scenarios or geometrical configurations, so there is a need for an improvement to the melt parameterisation that is aware, in some sense, of grid resolution.

As emphasised in Section 5.1 one of the main motivations for using the Firedrake framework to simulate ice-ocean interactions is the availability of an automatically generated adjoint model.

In the next chapter we show our first steps at using this capability to investigate sensitivities in an ice shelf ocean cavity context.

Chapter 4

Adjoint Sensitivity Calculations Applied to Ice Shelf Cavities

Highlights

- Adjoint simulations capture the reverse overturning 'ice pump' mechanism.

- The sign difference between the adjoint sensitivity and temperature fields highlights the importance of buoyancy forcing.

- We find the grounding zone is a consistently high region of sensitivity of melt to temperature, salinity, viscosity, diffusivity and the turbulent exchange coefficient in the melt parameterisation, Γ_T.

- By using an objective function based on depth profile measurements of temperature and salinity we build towards inverting for a range of unknowns relevant for grounding zone observations in the future.

Summary

This chapter shows preliminary adjoint sensitivity calculations based on the idealised grounding zone domain of Section 3.2. The adjoint sensitivity results provide a different view of the characteristic 'ice pump' mechanism, this time seen in reverse. The sign difference in sensitivity to temperature and salinity implies the melt rate is most sensitive to buoyancy changes, not thermodynamics, as confirmed by previous studies. The grounding zone is consistently a region of high sensitivity, reinforcing the importance of modelling these regions accurately. We finish with a sensitivity experiment using an objective function relating the difference between two forward synthetic runs to differences in the boundary forcing. This is inspired by observations taken at the Thwaites grounding zone and the work here is building towards incorporating these measurements into an unstructured ocean model in the future. Crucially, Firedrake's symbolic representation of the discretised equations means that as the forward model is developed, these changes can be automatically carried over to the adjoint model, leading to an efficient adjoint in terms of both computational expense and software development time.

Publications and Disclaimer

The adjoint sensitivity calculations in Section 4.3 are adapted from Scott et al. [2023].

4.1 Introduction

As emphasised in Chapter 1 the availability of an automatic adjoint model is a significant motivation for using the Firedrake finite element framework. Calculating the sensitivity of model outputs to input parameters is evidently useful in its own right. For PDE constrained optimisation, gradient information is essential for obtaining efficient numerical algorithms. The adjoint method is able to calculate the gradient of a functional with respect to model input parameters, in a solution time independent of the number of input parameters. This result

is essential for ensuring that calculating gradient information is tractable when the number of input parameters is large. This is especially true of grid based PDE discretisations, like ocean models, where the unknown input parameter fields are typically spatially dependent and thus scale with the number of grid nodes. Section 4.2 revises how this result is obtained.

The key idea is that Firedrake represents the discretised weak forms of the equations (see Chapter 2) symbolically, so Pyadjoint, the automatic differentiation library implemented by the Dolfin-Adjoint project [Farrell et al., 2013, Mitusch et al., 2019], can be applied directly to the high-level symbolic representation of the discrete mathematical problem written in UFL. This means that in theory the time taken to solve the adjoint system is as fast as one linear solve of the forward model [Farrell et al., 2013]. For nonlinear forward models, that may require multiple linear solves as part of an iterative solution strategy, the associated adjoint cost can be less than the forward model. For the runs presented in this chapter, the combined adjoint solution time (involving assembly and solve time) is about six times slower than the forward model. It is worth emphasising that these simulations are coarse and the adjoint solution time is dominated by fixed costs per timestep due to symbolic factorisation. When using a finer mesh from the idealised grounding zone comparison with MITgcm from Chapter 3 we find the adjoint run is 2.7 times slower than the forward model. For a more detailed discussion and a breakdown of the adjoint timing results see Section 4.5 at the end of this Chapter. We expect for problems with more degrees of freedom, in particular in 3D, the symbolic cost of the adjoint solve will become relatively small compared to the time for solving the linear systems, thus the time taken for the adjoint solution will approach the forward model. For comparison Goldberg et al. [2020] suggest that the adjoint implementation for MITgcm with the highly optimised, commercial TAF automatic differentiation tool is three to six times slower than the forward model, whereas the open source OpenAD tool, which has not had the same level of bespoke optimisation, can be an order of magnitude slower. This is in part due to the slowing of the primal code by introducing new variables. As alluded to in Chapter 1, automatic differentiation based on a 'line by line' approach is often referred to as 'algorithmic' differentiation given the need for user intervention to maintain a computationally efficient adjoint model. This highlights the advantage of the Firedrake framework, that is not only open source, but automatically

generates an efficient adjoint model.

The work here is still preliminary; we only demonstrate capability in 2D, primarily due to memory constraints, as a robust checkpointing algorithm was only added to Firedrake's adjoint capability in the latter stages of this thesis. However, the work here represents a step towards a unique tool able to calculate adjoint sensitivities in fully unstructured ice shelf ocean cavity domains, that is flexible to extensions in the model formulation (for instance changes to the melt parameterisation) and importantly, remains computationally efficient despite this flexibility.

4.2 Adjoint System of Equations

This section derives the discrete form of the adjoint equation to build intuition for sensitivity calculations discussed in the following sections. The derivation follows the mathematical background documentation from the Dolfin Adjoint project [Farrell]. We are interested in finding the sensitivity of some functional $J(\mathbf{u}(\mathbf{m}), \mathbf{m})$ with respect to the model parameters $\mathbf{m} \in \mathbb{R}^M$ (vector of length M). $\mathbf{u} \in \mathbb{R}^U$ (vector of length U) is the solution to a system of PDEs (i.e. incompressible Navier Stokes with tracer advection) given by

$$\mathbf{F}(\mathbf{u}, \mathbf{m}) = \mathbf{0}. \tag{4.1}$$

Here we focus on discrete adjoints after the finite element approximation has been carried out, where the constraint $\mathbf{F}$ represents the discrete equations, and the derived adjoint equations take the form of matrix-vector systems. It is possible to consider the continuous equations and derive the adjoint equations using variational calculus. The advantage of the discrete adjoint approach, however, is that because the adjoint solution is based on the discrete equations the obtained gradient is consistent with the discrete forward problem [Wallwork, 2021].

Rewriting the objective function purely in terms of $\mathbf{m}$ gives

$$\hat{J}(\mathbf{m}) = J(\mathbf{u}(\mathbf{m}), \mathbf{m}), \tag{4.2}$$

where $\hat{J}(\mathbf{m})$ is referred to as the reduced functional. This makes it clear that we are aiming to calculate the sensitivity of the functional to the parameters $\mathbf{m}$ and finding the solution $\mathbf{u}$ is an intermediate step towards this. Applying the chain rule to $\hat{J}(\mathbf{m})$ gives

$$\underset{1\times M}{\frac{\mathrm{d}\hat{J}}{\mathrm{d}\mathbf{m}}} = \underset{1\times U}{\frac{\partial J}{\partial \mathbf{u}}} \underset{U\times M}{\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{m}}} + \underset{1\times M}{\frac{\partial J}{\partial \mathbf{m}}}. \tag{4.3}$$

Although $\partial J/\partial \mathbf{u}$ and $\partial J/\partial \mathbf{m}$ can be evaluated using algorithmic differentiation techniques, if the number of parameters M is very large then the calculation of the matrix $\mathrm{d}\mathbf{u}/\mathrm{d}\mathbf{m}$ becomes prohibitively expensive. Taking the derivative of Equation 4.1 w.r.t $\mathbf{m}$ and rearranging gives the Tangent Linear Model (TLM)

$$\underset{U\times U}{\frac{\partial \mathbf{F}(\mathbf{u},\mathbf{m})}{\partial \mathbf{u}}} \underset{U\times M}{\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{m}}} = -\underset{U\times M}{\frac{\partial \mathbf{F}(\mathbf{u},\mathbf{m})}{\partial \mathbf{m}}}. \tag{4.4}$$

This is a linearisation of the forward model. If there are only a few parameters, i.e. the length of $\mathbf{m}$ is small, then an efficient way to calculate $\mathrm{d}\mathbf{u}/\mathrm{d}\mathbf{m}$ is to solve Equation 4.4. However, we are looking for an efficient method when $\mathbf{m}$ is a vector with many unknowns arising from a grid discretisation of an unknown spatial field in an ocean model. The next step towards the adjoint system is to imagine that the TLM system is invertible so that the solution Jacobian is expressed as

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{m}} = -\left(\frac{\partial \mathbf{F}(\mathbf{u},\mathbf{m})}{\partial \mathbf{u}}\right)^{-1} \frac{\partial \mathbf{F}(\mathbf{u},\mathbf{m})}{\partial \mathbf{m}}. \tag{4.5}$$

Substituting this expression for the solution Jacobian in Equation 4.3 gives

$$\frac{\mathrm{d}\hat{J}}{\mathrm{d}\mathbf{m}} = -\frac{\partial J}{\partial \mathbf{u}}\left(\frac{\partial \mathbf{F}(\mathbf{u},\mathbf{m})}{\partial \mathbf{u}}\right)^{-1} \frac{\partial \mathbf{F}(\mathbf{u},\mathbf{m})}{\partial \mathbf{m}} + \frac{\partial J}{\partial \mathbf{m}}. \tag{4.6}$$

The mathematical trick to make progress is to now take the adjoint of Equation 4.6

$$\underset{M\times 1}{\frac{\mathrm{d}\hat{J}^T}{\mathrm{d}\mathbf{m}}} = -\underset{M\times U}{\frac{\partial \mathbf{F}^T}{\partial \mathbf{m}}} \underset{U\times U}{\frac{\partial \mathbf{F}}{\partial \mathbf{u}}}^{-T} \underset{U\times 1}{\frac{\partial J^T}{\partial \mathbf{u}}} + \underset{M\times 1}{\frac{\partial J^T}{\partial \mathbf{m}}}. \tag{4.7}$$

Note that because the vectors here are real taking the adjoint is equivalent to transposing the matrices. For complex vectors one would also need to take the complex conjugate.

Defining an adjoint vector $\boldsymbol{\lambda}$ as

$$\boldsymbol{\lambda} = \left(\frac{\partial \mathbf{F}(\mathbf{u}, \mathbf{m})}{\partial \mathbf{u}}\right)^{-T} \frac{\partial J^T}{\partial \mathbf{u}}, \tag{4.8}$$

gives the adjoint system of equations

$$\left(\frac{\partial \mathbf{F}(\mathbf{u}, \mathbf{m})}{\partial \mathbf{u}}\right)^{T} \boldsymbol{\lambda} = \frac{\partial J^T}{\partial \mathbf{u}}. \tag{4.9}$$

The key improvement that this rearranging has achieved is that the dimension of the matrix vector system is independent of the size of the vector $\mathbf{m}$. $\boldsymbol{\lambda}$ is a vector of length U so the cost of solving for $\boldsymbol{\lambda}$ becomes comparable to solving the forward model (4.1).

Rewriting Equation 4.7 in terms of the adjoint vector $\boldsymbol{\lambda}$ finally gives the gradient of the functional as

$$\underset{M\times 1}{\frac{\mathrm{d}\hat{J}^T}{\mathrm{d}\mathbf{m}}} = -\underset{M\times U}{\frac{\partial \mathbf{F}^T}{\partial \mathbf{m}}}\,\underset{U\times 1}{\boldsymbol{\lambda}} + \underset{M\times 1}{\frac{\partial J^T}{\partial \mathbf{m}}}. \tag{4.10}$$

We will now consider an example of 1D advection. The simplest finite difference explicit discretisation for a 1D advection problem is given by

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + v\frac{u_i^n - u_{i-1}^n}{\Delta x} = 0, \tag{4.11}$$

where u is the unknown quantity being advected, v is the velocity ($v > 0$ implies flow to the right), Δt is the timestep and Δx is the grid size. The subscript indicates the grid location and the superscript represents the time level. This can be simplified to

$$u_i^{n+1} + (k-1)u_i^n - ku_i^n = 0, \tag{4.12}$$

where $k = v\Delta t/\Delta x$.

Simplifying the system to just three grid points and three time steps gives the following system

of equations

$$\underbrace{\begin{pmatrix} I & 0 & 0 \\ K & I & 0 \\ 0 & K & I \end{pmatrix}}_{9\times 9} \underbrace{\begin{pmatrix} u_0^0 \\ u_1^0 \\ u_2^0 \\ \hline u_0^1 \\ u_1^1 \\ u_2^1 \\ \hline u_0^2 \\ u_1^2 \\ u_2^2 \end{pmatrix}}_{9\times 1} = \underbrace{\begin{pmatrix} u_0^{IC} \\ u_1^{IC} \\ u_2^{IC} \\ \hline u_{BC}^1 \\ 0 \\ 0 \\ \hline u_{BC}^2 \\ 0 \\ 0 \end{pmatrix}}_{9\times 1} \tag{4.13}$$

where I is the 3×3 identity matrix, and the block matrix K is given by

$$K = \underbrace{\begin{pmatrix} 0 & 0 & 0 \\ -k & (k-1) & 0 \\ 0 & -k & (k-1) \end{pmatrix}}_{3\times 3}. \tag{4.14}$$

The forward system in Equation 4.13 is lower triangular due to a boundary condition on the left hand side and the upwinded discretisation of advection towards the right. Since this system is linear the equivalent adjoint system in Equation 4.15 is just the transpose of the operator. Now the matrix appearing in the adjoint equation is upper triangular. This implies that information starts with a boundary condition at the right hand side and 'flows' to the left. In other words the flow of information has reversed for the adjoint model, with sensitivities evolving in the

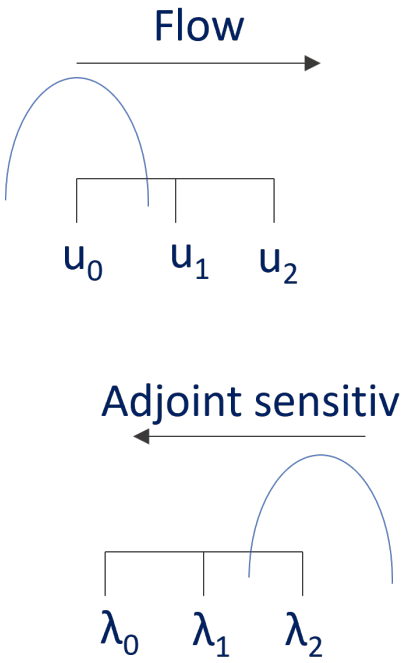


Figure 4.1: Schematic for an adjoint advection problem from Equations 4.13 and 4.15 showing the spatial discretisation.

opposite sense to the forward model.

$$\underbrace{\begin{pmatrix} I & K^T & 0 \\ 0 & I & K^T \\ 0 & 0 & I \end{pmatrix}}_{9\times 9} \begin{pmatrix} \lambda_0^0 \\ \lambda_1^0 \\ \lambda_2^0 \\ \hline \lambda_0^1 \\ \lambda_1^1 \\ \lambda_2^1 \\ \hline \lambda_0^2 \\ \lambda_1^2 \\ \lambda_2^2 \end{pmatrix}_{9\times 1} = \begin{pmatrix} \partial J/u_0^0 \\ \partial J/u_1^0 \\ \partial J/u_2^0 \\ \hline \partial J/u_0^1 \\ \partial J/u_1^1 \\ \partial J/u_2^1 \\ \hline \partial J/u_0^2 \\ \partial J/u_1^2 \\ \partial J/u_2^2 \end{pmatrix}_{9\times 1} \tag{4.15}$$

This will be seen in Section 4.3 for sensitivity calculations of the melt rate to the evolving temperature and salinity fields.

4.3 Adjoint Sensitivity Calculations in a 2D Idealised Grounding Zone Domain

We present preliminary results of sensitivity information obtained with the adjoint to our model. A simplified domain has been chosen to make interpretation of the adjoint sensitivity patterns easier. The domain is a vertical 2D slice within 10 km of an idealised grounding zone domain, with a 2 m wall at the grounding line and a 100 m wall at the open ocean. The domain has a uniform seabed depth of 600 m. The temperature and salinity are initialised with constant temperature and salinity of 1 °C and 34.4, respectively. The temperature and salinity fields are relaxed to these values in a sponge region, which linearly ramps up over the final four grid cells to the right hand side of the domain with a restoring period of one day at the boundary. The grid consists of triangles arranged in columns, similar to a z-layer discretisation. For an example the reader is referred to Kimura et al. [2013]. The grid resolution is 500 m in the horizontal and 2 m in the vertical. Horizontal viscosity is 0.25 m^2/s and vertical viscosity is 1×10^{-3} m^2/s, matching the grid aspect ratio. Diffusivity for both tracers is set equal to viscosity. We use a P1DG-P2 discretisation for velocity and pressure, and P1DG is used for the tracers. The timestep is 300 s. The forward simulation is run for 50 days to reach a spun-up state. We apply Robin boundary conditions at the ice ocean boundary as described in Section 2.6 that drives a buoyancy driven flow within the domain. Figure 4.2 shows the spun-up state at 50 days. The flow is characteristic of the ice pump mechanism, similar to the simulation from Section 3.2, with a clockwise overturning flow, concentrated near the ice ocean boundary, driven by buoyant melt water. At the grounding line the water column is well mixed in the vertical, though the temperature field becomes progressively more stratified away from the grounding line. This type of flow is similar to that described in Holland [2008] where background tidal mixing dominates at the grounding zone if the overturning circulation weakens. In this instance it is likely caused by the coarse grid resolution and constant eddy diffusivity values, because at the 2 m high grounding zone wall on the left hand side of the domain there is only one cell to resolve the overturning circulation. It is likely that refining the grid resolution (effectively reducing implicit spurious mixing as described in Section 3.6) or

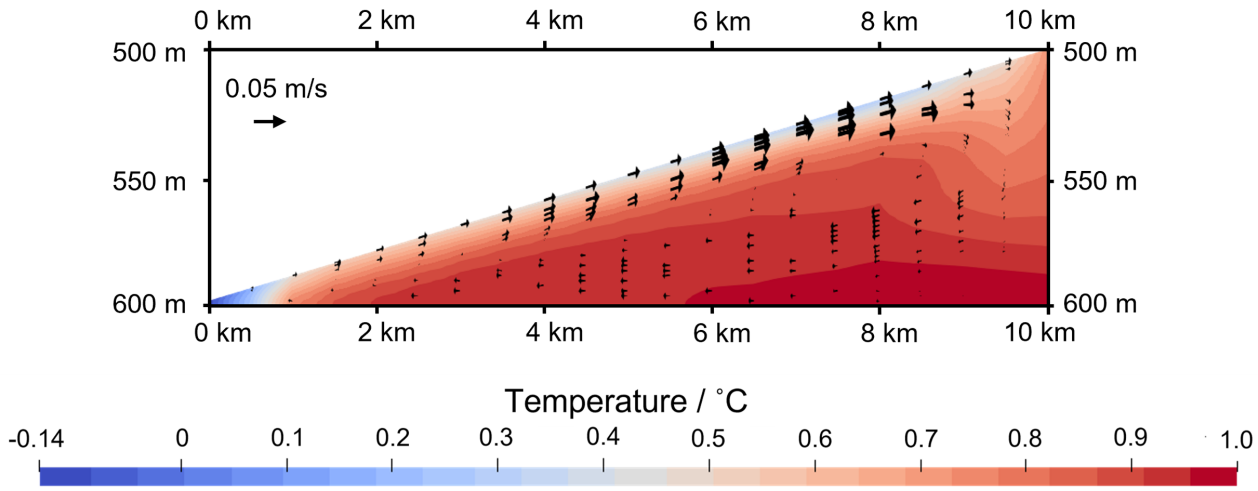


Figure 4.2: Temperature and velocity fields at 50 days after the forward model has spun up. After 30 days of simulation time there is negligible change in the fields or melt rate (not shown).

using a turbulence model would change the rate of mixing in this region and the water column would become more (vertically) stratified.

The model is run for a further 20 days with the adjoint model. An objective functional is chosen to be the total basal melt beneath the ice shelf at the final time step. Although using the average basal melt beneath the ice shelf may have been a more conventional quantity, this can be obtained by dividing the results by the length of the ice shelf. We calculate sensitivities of this functional to perturbations of the spatially varying temperature and salinity fields at each time level, as well as to spatially varying perturbations of the viscosity, diffusivity and Γ_T (turbulent exchange coefficient from the melt parameterisation) fields which are kept fixed in time. In the context of this adjoint sensitivity analysis these fields are referred to as 'controls' [Errico, 1997].

The accuracy of the adjoint sensitivity can be verified by a Taylor remainder convergence test [Farrell et al., 2013]. If the gradient is correct then with $J(m)$ as the functional, m as the control, and δm as a small perturbation in an arbitrary direction, $|J(m+\delta m) - J(m) - \nabla J \cdot \delta m|$ should converge to zero with second order accuracy as the magnitude of the perturbation is reduced,

Control	h	$0.5h$	$0.25h$	$0.125h$
temperature	1.99e-9 -	4.95e-10 2.01	1.22e-10 2.02	2.98e-11 2.04
salinity	2.38e-8 -	5.98e-9 2.01	1.44e-9 2.03	3.47e-10 2.06
ν	2.59e-7 -	6.62e-8 1.99	1.68e-8 1.98	4.24e-9 1.98
κ_T	1.21e-7 -	3.07e-8 1.98	7.73e-9 1.99	1.94e-9 2.00
κ_S	3.85e-9 -	9.72e-9 1.99	2.44e-9 1.99	6.10e-10 2.00
Γ_T	1.24e-5 -	3.10e-6 2.00	7.75e-7 2.00	1.93e-7 2.00

Table 4.1: Taylor remainder convergence tests for temperature, salinity, viscosity, temperature diffusivity, salinity diffusivity and turbulent exchange coefficient control fields. In each column the perturbation to the control is successively halved. $|J(m + \delta m) - J(m) - \nabla J \cdot \delta m|$ is evaluated for a random perturbation with maximum magnitude h to every component of the control vector. For temperature, salinity and Γ_T, h is 0.01, and for viscosity and diffusivity h is 0.0001 to ensure that the perturbation is small relative to the magnitude of the control. Beneath the error the order of convergence is given, which is two for all controls, verifying the adjoint model is calculating gradient information accurately.

i.e. halving the size of the perturbation should cause the result to decrease by a factor of four. We have carried out such Taylor remainder convergence tests for each control field, with the melt functional evaluated after ten timesteps (3000 s) from the 50 day checkpoint. The results shown in Table 4.1 confirm that the errors converge at second order and verify that the gradient information is being calculated accurately by the adjoint model.

Before analysing the sensitivity fields it is worth briefly explaining the interpretation of units, which is important when making quantitative comparisons between sensitivities for different controls. Pyadjoint returns the sensitivity of the functional with respect to the discrete control vector. This gradient is sometimes referred to as being calculated with respect to the 'small' l_2 norm. This definition of sensitivity is inherently mesh dependent because the magnitude of the discrete gradient with respect to the vector of model parameters will decrease as the grid is refined, because each node has successively less influence on the overall functional. Instead we show sensitivities, $\partial J/\partial m$, that are calculated with respect to the L_2 norm, such that $< \partial J/\partial m, \delta m >= \delta J$, where $< \cdot, \cdot >$ implies taking the inner product and integrating over the domain. By using this integral definition of sensitivity, the sensitivities are not dependent on

the mesh resolution, which is an important consideration if the grid size varies within the mesh. Note that for uniform resolution meshes this generally means that there will be a constant factor correction to the absolute value of the gradient based on the area (or volume) of an element.

As an example, for the experiments here the units for adjoint temperature calculated with respect to the L_2 norm are $(\mathrm{K\ s})^{-1}$. This is because the objective function, J, has units of m^2/s (volume flux) from integrating the melt rate (units of m/s) along the 1D ice ocean interface. We also have a 2D vertical slice domain, so the L_2 integral is over a 2D surface. Finally, the temperature perturbation field, δm, has units of K.

Figure 4.3 shows adjoint sensitivity of the final total melt flux (note defined here as a volume flux, not a mass flux) to temperature and salinity at three times in the simulation. The adjoint model is effectively a transpose of a linearised version of the full, non-linear forward model. This linear forward model is referred to as the Tangent Linear Model (TLM) and it relates perturbations of the control fields to changes in the objective function. The adjoint model reverses the flow of information (because it is the transpose of the TLM), instead calculating the sensitivity of the objective function to the controls and intermediate variables. For time dependent models the adjoint sensitivities thus 'flow backwards' in time, from the time when the objective function is evaluated towards the earlier times, provided that the controls had an impact on the objective function [Errico, 1997]. The top panel of Figure 4.3 shows six hours before the final time, where the sensitivity fields are concentrated at the ice-ocean boundary, where the objective function is evaluated. Since we have evaluated the objective function at the final time, this is equivalent to adding an 'initial' condition to the adjoint model. This can be seen from the example in Equation 4.15, as the initial condition for the adjoint solution, $\lambda^2_{0,1,2}$, only depends on the derivative of the objective function w.r.t the final state, $\partial J/\partial u^2_{0,1,2}$. As such even though the forward model has reached a steady state, the adjoint sensitivities evolve backwards with time in the opposite sense to the forward model. At two days before the final time (middle panel) the sensitivity patterns are concentrated closer to the grounding line and at the bottom of the cavity. Finally, five days before the final time (bottom panel) most of the sensitivity of the objective function to the temperature and salinity fields has been lost. The counterclockwise motion of the sensitivity fields is effectively the ice-pump mechanism, but

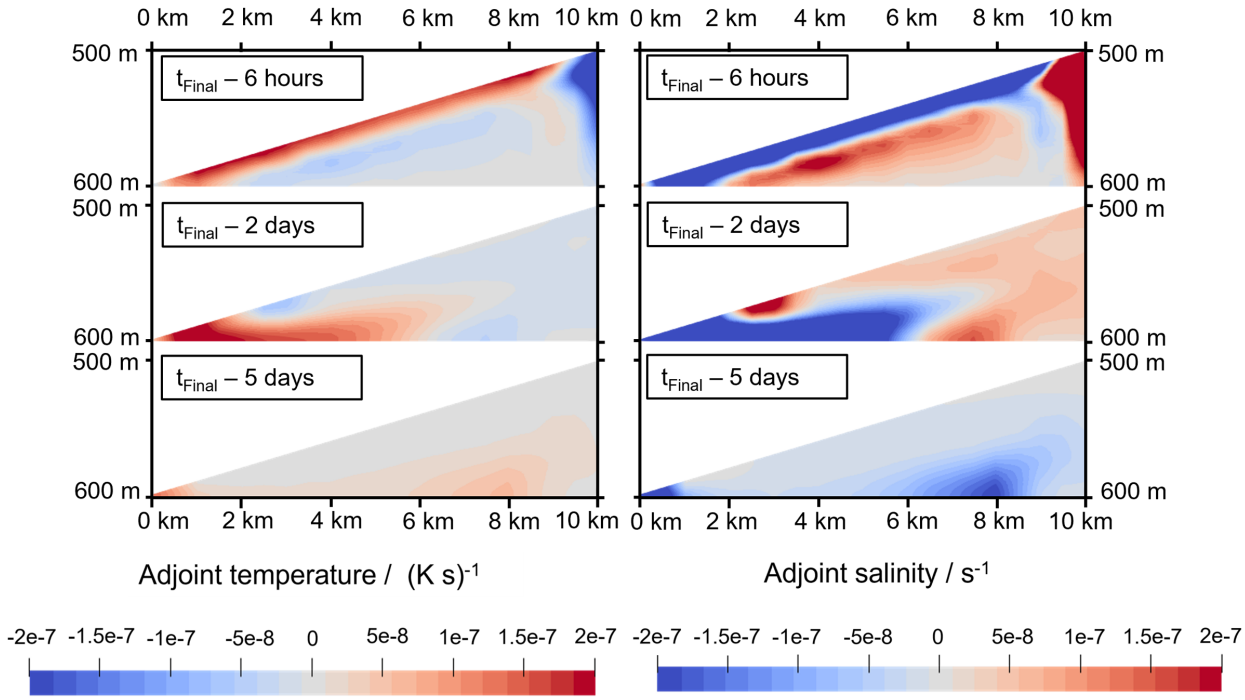


Figure 4.3: Adjoint sensitivity fields of final total melt flux with respect to temperature (left) and salinity (right) at three time intervals: 6 hours before the final time, two days before the final time and five days before the final time. Note the opposite sign of adjoint temperature and salinity. The sensitivities $\partial J/\partial m$ are calculated with respect to the L_2 norm, such that $< \partial J/\partial m, \delta m >= \delta J$, where $< \cdot, \cdot >$ implies taking the inner product and integrating over the domain. By using this integral definition of sensitivity, the sensitivities are not dependent on the mesh resolution (an important consideration if the grid size varies within the mesh). The units for adjoint temperature in this case become 1 / (K s), given a objective function, J with unit m^2 /s (volume flux) and a temperature perturbation field, δm, with units of K.

viewed in reverse. The time for the sensitivity fields to be lost is consistent with the forward velocity. Flow speeds are on the order of 0.05 m/s, and an advection path length of 20 km (twice the domain length) gives a circulation time of 4.6 days.

An interesting feature of Figure 4.3 is that the signs of adjoint temperature and salinity are opposite. In the same region a small positive perturbation to temperature is required to increase the final total melt flux, whereas an increase in salinity would decrease the final total melt flux. Initially this seems counter intuitive, since increasing salinity decreases the freezing point, which should lead to more melting and thus have the same effect as increasing temperature. This dependency has been found before in an adjoint simulation of Pine Island Glacier ice shelf cavity [Heimbach and Losch, 2012]. Heimbach and Losch [2012] suggest that buoyancy effects may account for the sensitivity difference, since increasing temperature and decreasing salinity

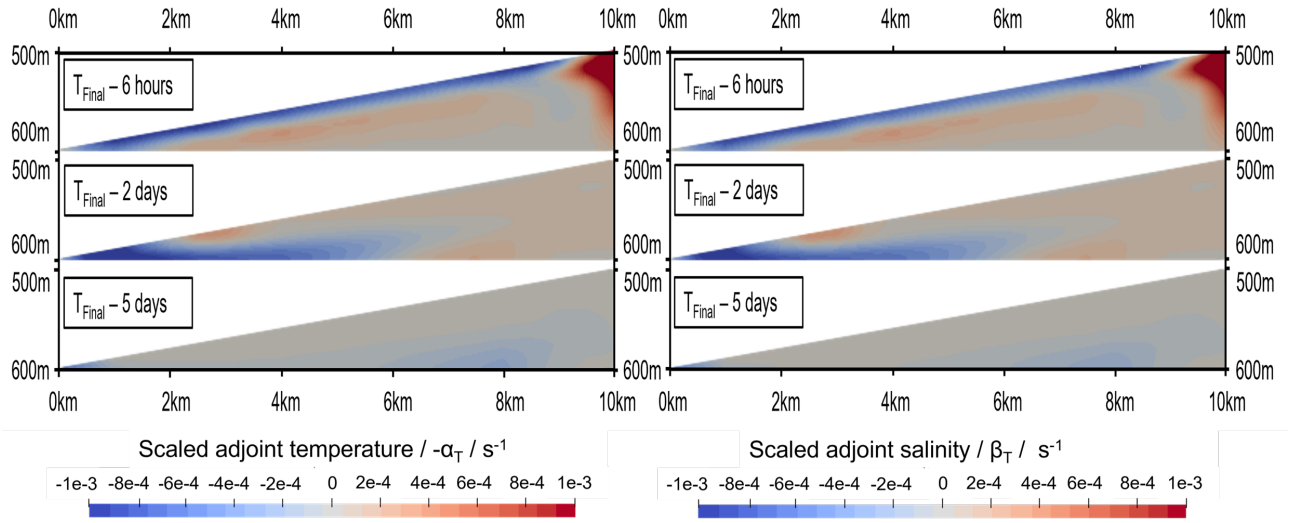


Figure 4.4: Adjoint sensitivity fields of final total melt flux with respect to temperature (left) and salinity (right) scaled by the expansion coefficients for the equation of state ($\alpha_T = 2 \times 10^{-4}\,^{\circ}\mathrm{C}^{-1}$, $\beta_S = 7 \times 10^{-4}$). The sensitivities are calculated with respect to the L_2 norm (see the caption of Figure 4.3 for an explication of units). The units for both fields are 1 / s. The plots are at the same three time intervals as 4.3: 6 hours before the final time, two days before the final time and five days before the final time.

both increase buoyancy. This in turn should strengthen the overturning ice pump and lead to higher melt rates. Figure 4.4 shows that dividing the sensitivity fields by the respective expansion coefficients in the linear equation of state ($\alpha_T = -2 \times 10^{-4}\,^{\circ}\mathrm{C}^{-1}$, $\beta_S = 7 \times 10^{-4}$) the two sensitivity fields are almost identical, in sign and magnitude strongly suggesting that buoyancy is controlling the sensitivity pattern. Although this is a highly idealised experiment, the adjoint model shows potential to investigate the importance of subglacial hydrology and the formation of buoyant plumes at the grounding line and hence the overall effect on melting, as a complimentary tool to the forward model.

Figure 4.5 shows patterns of adjoint sensitivity for vertical viscosity, temperature diffusivity and salinity diffusivity. Since the value of the vertical components are 250 times smaller than the respective horizontal components (matching the grid aspect ratio), a relatively 'small' (e.g. 10 %) perturbation to both components would only give the same response to the objective functional if the sensitivity to the vertical component is 250 times stronger than the horizontal components. The sensitivity with respect to the vertical components is actually about 1000

times stronger than the sensitivity with respect to the horizontal components (not shown). This seems to imply that the relative uncertainty in the vertical components are more important than the horizontal, which is not surprising considering how sensitive the melt rate and hence the flow is to vertical stratification as explored in Section 3.6.

Sensitivity for all three fields is concentrated towards the grounding line and the ice-ocean interface. This is where the biggest changes in velocity, temperature and salinity properties occur, as seen from Figure 4.2, so it makes sense that changing the diffusivity and viscosity in these regions should affect the final melt rate. Equally, away from the ice ocean boundary and the grounding zone, where gradients in the flow properties are much weaker, the adjoint results show that changing the diffusivity and viscosity would have much less of an effect on the final melt rate.

At the ice ocean interface away from the grounding line there is a pattern of positive sensitivity with respect to temperature diffusivity but negative sensitivity with respect to salinity diffusivity. This is similar to the earlier change in sign between the adjoint temperature and salinity fields, which again suggests that maybe this can be interpreted as a buoyancy effect. Increasing the temperature diffusivity would bring warmer, less dense water to the interface and likewise, decreasing the salinity diffusivity would slow down the mixing of denser, high salinity water from below into the 'ice pump' boundary plume.

The sign of the adjoint diffusivity fields at the grounding zone have reversed compared with the ice ocean interface; now increasing temperature diffusivity would decrease the melt rate at the final time, whereas increasing the salinity diffusivity would increase the melt rate at the final time. It is not immediately obvious how to interpret the sign of both these fields in terms of buoyancy or thermodynamic effects. Possibly there is a change in sign because the stratification near the grounding line is different to the ice ocean interface as shown by Figure 4.2 within 2 km from the grounding line. At the ice ocean interface the water column is vertically stratified, whereas near the grounding zone the temperature and salinity fields are much more uniform in the vertical direction. Results from Yeager [2018] using a k-ϵ Reynolds Averaged Navier-Stokes (RANS) model to investigate ocean conditions near grounding lines,

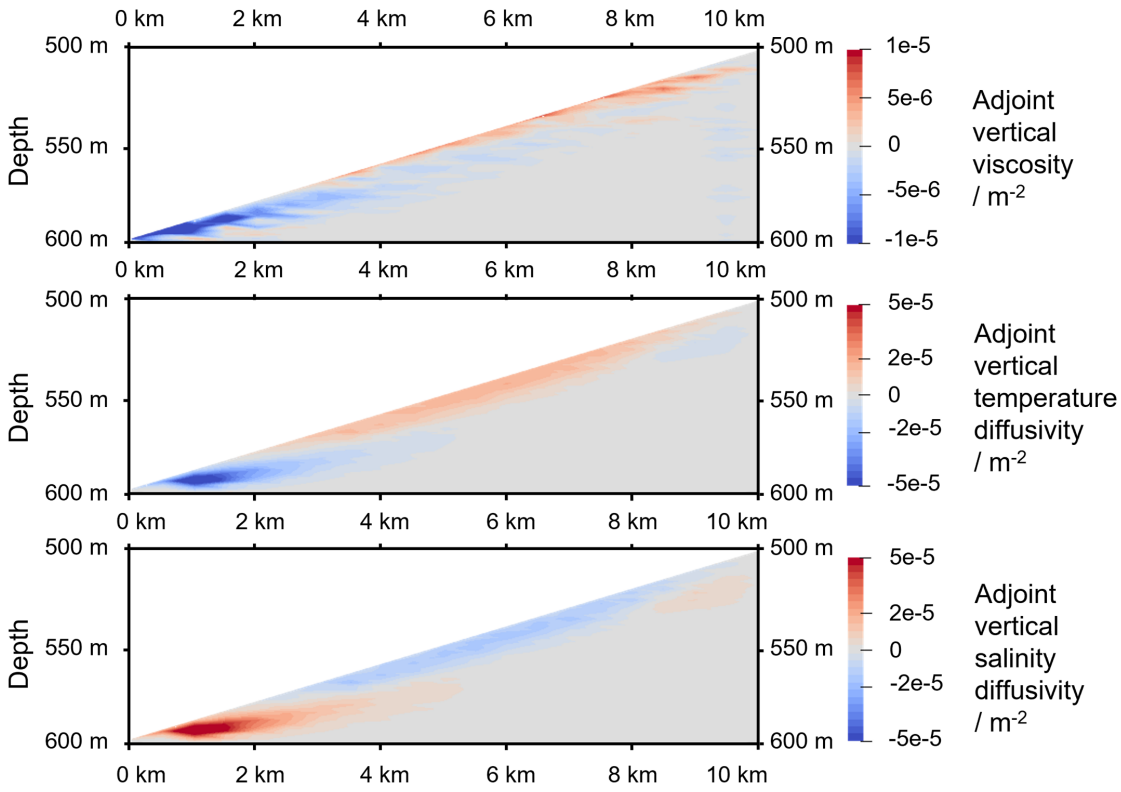


Figure 4.5: Sensitivity of total basal melt at the final time step to vertical viscosity and eddy diffusivity for temperature and salinity calculated with respect to the L_2 norm. (see the caption of Figure 4.3 for an explication of units). Note the change in colour scale for the adjoint vertical viscosity compared with adjoint temperature and salinity vertical diffusivity.

suggest that changes in diffusivity of several orders of magnitude may occur within 1 km of the grounding line. Although these large changes in diffusivity would not linearly relate to changes in melt rate, the fact that total melt rates are sensitive to grounding line eddy diffusivity values, which probably vary significantly, highlights the importance of modelling these values accurately.

Figure 4.6 shows the sensitivity of final total melt rate to the turbulent exchange coefficients in the melt parameterisation (see Section 2.6). For this forward run Γ_T is set to a constant value of 0.011 and $\Gamma_S = \Gamma_T/35$, following Asay-Davis et al. [2016]. As before with viscosity and diffusivity, these parameters are constant in time and space, but we are calculating the (time independent) spatial sensitivity field. The left axis of Figure 4.6 plots the melt rate, w_b, which has an average magnitude of $\sim$ 16 m/a. The right axis shows the sensitivity of the total melt rate with respect to the Γ_T field, $\partial J/\partial \Gamma_T$, multiplied by a scalar, $\Delta\Gamma_T$. In this case we have chosen $\Delta\Gamma_T = \Gamma_T$. Broadly the pattern of the scaled sensitivity follows the pattern of the melt

rates over individual cells, though the scaled sensitivity does not seem to be increasing away from the grounding line in line with the melt rate. However, since the magnitude of the melt rate is very similar to the magnitude of the scaled sensitivity (~ 13 m/a), this implies that the melt rate is approximately linearly proportional to Γ_T i.e. $w_b \approx \partial J/\partial \Gamma_T \, \Gamma_T$, where w_b is the local melt rate. This seems intuitive given the linear dependence of the turbulent heat flux, Q_w^T, on Γ_T and that this term should balance exactly with the latent heat term, Q_{latent}^T, in the heat conservation equation (Equation 2.72) because we have set the heat flux into the ice as zero following Asay-Davis et al. [2016].

Note that $\partial J/\partial \Gamma_T$ is now calculated with respect to the L_2 'boundary' norm. This means the integration is only taken over the boundary (in this case the ice ocean interface) because Γ_T is only defined at the boundary. This is sometimes referred to as using the right Riesz representation. If the standard L_2 representation of the gradient is used, where there is an integration over the entire domain this to leads a spurious, grid dependent oscillating pattern in the sensitivity fields near the relevant boundary. By integrating over the entire domain we would need a Dirac delta function to ensure that only the gradient at the surface is evaluated. In the discrete setting we are evidently not able to apply an infinite value at the boundary, so there would be a discretisation error where the sensitivity leaks into the domain with an undershoot and the value at the boundary overshoots. The L_2 gradient will therefore be grid dependent and not converge to the true value as the grid is refined.

This work shows that it may be possible to carry out an inversion for the unknown Γ_T values based on an objective function measuring the error between satellite and model estimates of melt rate. There would likely be significant challenges to overcome, however, when carrying out an inversion especially when the state of the forward model is so uncertain and grid dependent owing to the melt parameterisation as seen in Section 3.6. Probably the full Navier-Stokes implementation here would be expensive in 3D in terms of both run time and memory costs for full cavity or regional sized model domains. Possibly adapting the Firedrake based (hydrostatic) coastal ocean model Thetis [Kärnä et al., 2018] that also inherits the adjoint capability would be a viable approach.

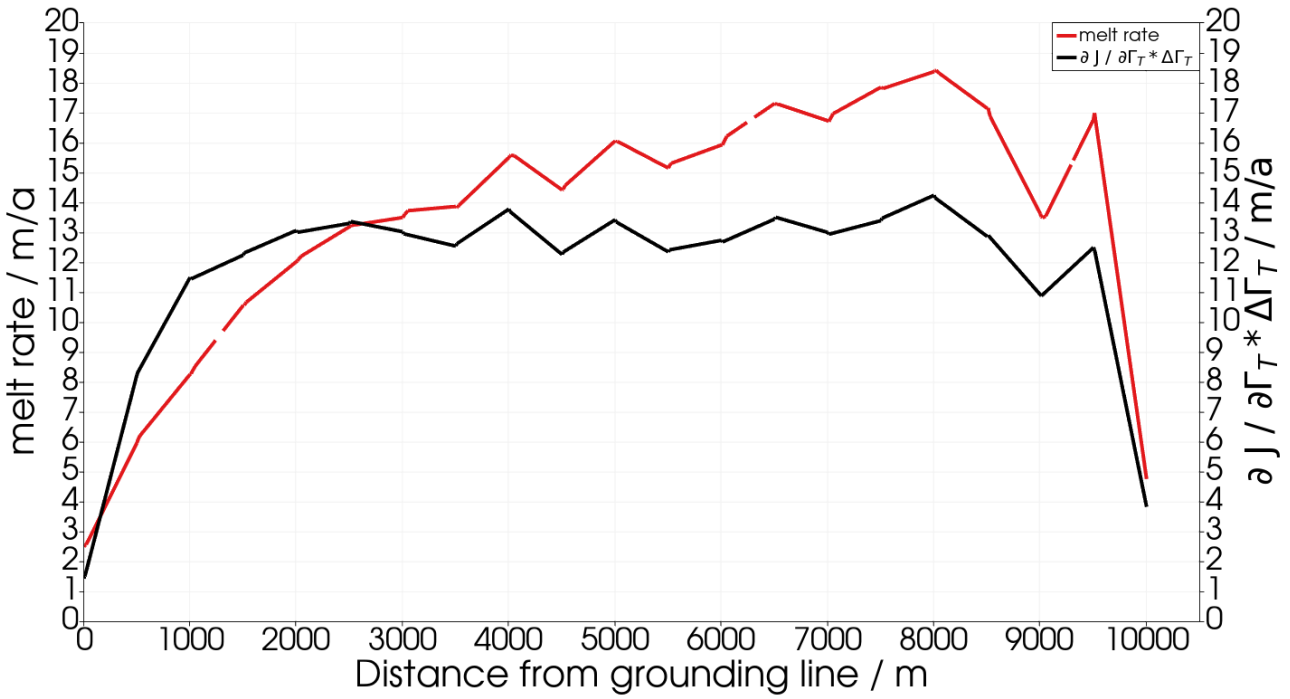


Figure 4.6: Sensitivity of total basal melt at the final time step to the turbulent heat exchange coefficient Γ_T, calculated with respect to the L_2 'boundary' norm (see main text for explanation). Note the two axis on the plot: the left axis is for the melt rate and the right axis is for sensitivity of the total melt rate with respect to the Γ_T field, $\partial J/\partial\Gamma_T$, multiplied by a scalar, $\Delta\Gamma_T$. In this case we have chosen $\Delta\Gamma_T = \Gamma_T$. The magnitude of the melt rate is very similar to the scaled sensitivity implying that the melt rate is linearly proportional to Γ_T.

Parameter	Value	Definition
L	10 km	Domain length
H	2 - 100 m	Domain height
D	600 m	Domain depth
Δx	500 m	Horizontal grid size
Δz	2 m	Vertical grid size
Δt	300 s	Timestep
T_{init}	1 °C	Initial and restoring temperature
S_{init}	34.4 °C	Initial and restoring salinity
t_{restore}	1 day	Restoring period
μ_h	0.25 $\text{m}^2\ \text{s}^{-1}$	Horizontal viscosity
μ_v	$1\times10^{-3}\ \text{m}^2\ \text{s}^{-1}$	Vertical viscosity
κ_h	0.25 $\text{m}^2\ \text{s}^{-1}$	Horizontal diffusivity
κ_v	$1\times10^{-3}\ \text{m}^2\ \text{s}^{-1}$	Vertical diffusivity
α_T	-2 $\times10^{-4}$ °C^{-1}	Temperature expansion coefficient
β_S	7 $\times10^{-4}$	Salinity contraction coefficient
Γ_S	0.011	Turbulent heat exchange coefficient

Table 4.2: Parameters and constants used for adjoint sensitivity calculations in the idealised grounding zone domain from Section 4.3. Note the values used for the expansion and contraction coefficients for temperature and salinity are changed from Table 2.5.

Given the importance of grounding zone processes and the associated uncertainty there is still real value in having a smaller scale model to constrain ocean dynamics in the vicinity of grounding zones. The next section discusses the applicability of the adjoint to detect changes in vertical stratification as measured by sub ice shelf borehole measurements through an idealised synthetic experiment.

4.4 Adjoint Sensitivity Calculations with a Depth Profile Objective Function

This section shows a further set of idealised sensitivity experiments from the same domain as Section 4.3, but with a different objective function. Here the functional is based on the squared difference of temperature and salinity depth profiles between two forward experiments

$$J = \sum_n \frac{1}{2n}\frac{1}{s^2}\left((-\alpha_T \Delta T)^2 + (\beta_S \Delta S)^2\right), \tag{4.16}$$

$$s = 0.5\beta_S, \tag{4.17}$$

$$\Delta T = T_{\text{forward 1}} - T_{\text{forward 2}} \tag{4.18}$$

$$\Delta S = S_{\text{forward 1}} - S_{\text{forward 2}}, \tag{4.19}$$

where n refers to the number of sample points in the depth profile and the subscripts refer to the two forward experiments. The two forward synthetic simulations differed in the applied stratification at the open RHS boundary, with constant and linear vertically stratified temperature and salinity respectively. The constant case has boundary temperature and salinity of 1 °C and 34.4 respectively. The linear case has temperature varying from 1 °C at 600 m depth to -0.5 °C at 500 m depth. The salinity varies from 34.8 at 600 m depth to 34.4 at 500 m depth. The motivation is to lay the ground work towards carrying out an inversion, inspired by CTD (Conductivity, Temperature, Depth) profiles taken at the MELT drilling site at Thwaites grounding zone as part of the International Thwaites Glacier Collaboration.

We use Firedrake's 'vertex only mesh' feature [Nixon-Hill et al., 2023] to sample the solution at 14 discrete points at y = 7550 m, from 525 m below the surface to 595 m. A 'vertex only mesh' is equivalent to a point cloud and the values associated with each point are represented by a DG0 finite element function, i.e. a constant value associated with each cell, or in this case point. For the user, accessing values at point locations is as simple as interpolating from the solution defined on main mesh to the 'vertex only mesh'. Crucially, as a function on the 'vertex only mesh' inherits all the usual features of a Firedrake function it means that these values can be used as part of an adjoint model automatically [Nixon-Hill et al., 2023].

The squared temperature and salinity differences are scaled by the coefficients in the linear EOS to ensure that the units of the functional are consistent. If working with real data it would be more appropriate to scale the differences by the uncertainty in the observations (and possibly model error if known). Also note that scaling the functional by a characteristic density perturbation (equivalent to a salinity difference of 0.5) ensures that the Taylor test convergence verification tests do not fail due to numerical round off errors, specifically for the gradient with respect to the salinity boundary condition. The Taylor test residuals and convergence rates

Control	h	$0.5h$	$0.25h$	$0.125h$
temperature	3.83e-6	9.58e-7	2.40e-7	5.99e-8
	-	2.00	2.00	2.00
salinity	4.68e-5	1.17e-5	2.93e-6	7.33e-7
	-	2.00	2.00	2.00
ν	4.39e-8	1.12e-8	2.82e-9	7.08e-10
	-	1.97	1.99	1.99
T_{bc}	6.67e-11	1.67e-11	4.16e-12	1.04e-12
	-	2.00	2.00	2.00
S_{bc}	8.95e-10	1.43e-10	1.76e-11	4.44e-12
	-	2.65	3.02	1.99

Table 4.3: Taylor remainder convergence tests for temperature, salinity, viscosity, boundary temperature and boundary salinity control fields for the depth profile objective function. Each column the perturbation to the control is successively halved. $|J(m+\delta m) - J(m) - \nabla J \cdot \delta m|$ is evaluated for a random perturbation with maximum magnitude h. For temperature and salinity h is 0.01, and for viscosity h is 0.0001 to ensure that the perturbation is small relative to the magnitude of the control. Beneath the error the order of convergence is given, which is two for all controls, verifying the adjoint model is calculating gradient information accurately.

after ten timesteps are given in Table 4.3 and show that the gradients are being calculated accurately (see Section 4.3 for an explanation of Taylor remainder convergence tests).

Sensitivity is calculated with respect to the forcing conditions for temperature and salinity on the boundary, as well as vertical diffusivity and viscosity. To enable the calculation of sensitivity to the boundary conditions required some forward model changes to the simulation from Section 4.3. In Section 4.3 the RHS boundary is closed, here the boundary is open. The applied pressure Dirichlet condition is calculated by integrating the difference between temperature and salinity along the boundary and the reference values from the equation of state in order to obtain a hydrostatic pressure profile. Note that this means there is a time lag, due to evaluating the boundary conditions from the scalar fields at the previous timestep. Dirichlet boundary conditions for temperature and salinity are only applied on the advection term where there is inflow. Neumann boundary conditions for temperature and salinity are set to zero. The restoring sponge region at the RHS is also switched off for simplicity. Apart from these boundary changes, the parameters used for forward simulation are identical to Section 4.3

Figure 4.7 shows the melt rate for the two forward simulations along the ice shelf after 50

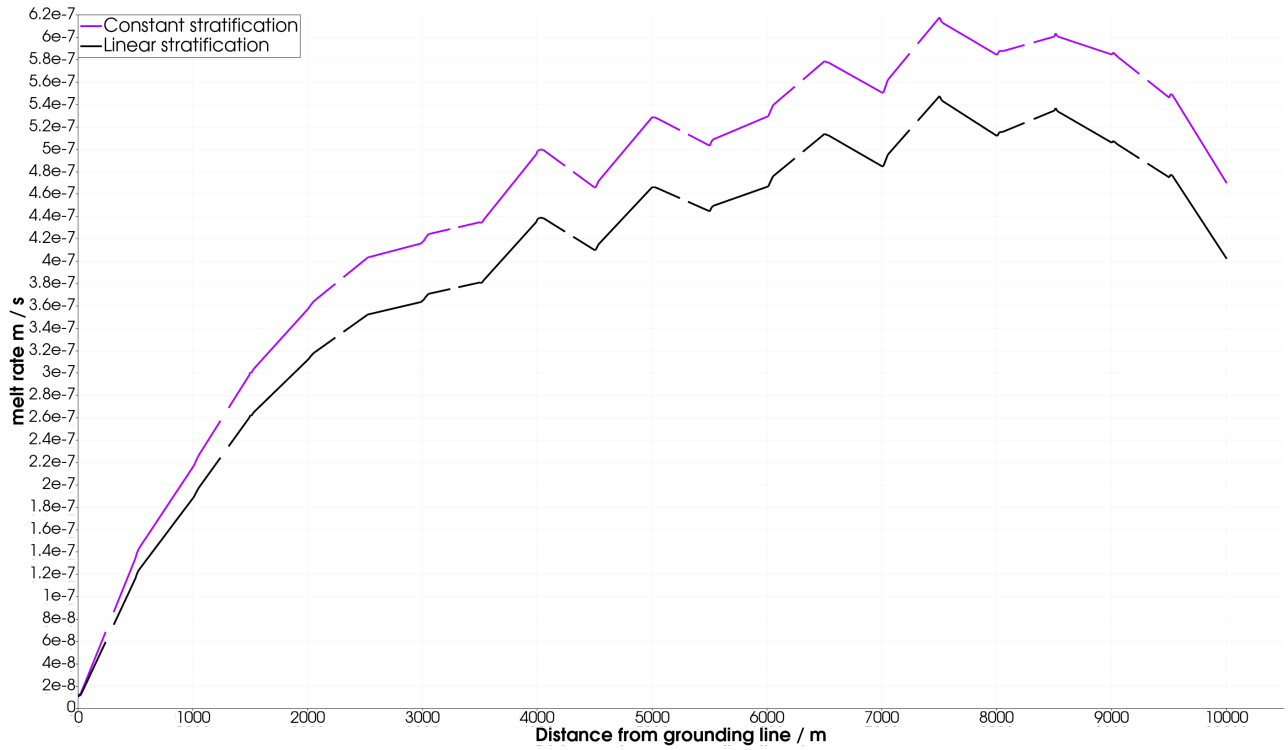


Figure 4.7: Melt profile at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (purple) and linear stratification (black).

days. Note that both models spun up within about 20 days of simulation time. The shape of the melt transect is very similar between both models with the magnitude of the constant case slightly higher than the linearly stratified case. The velocity fields are almost identical, as shown by Figure 4.8. Figures 4.9 and 4.10 show that the salinity (and temperature) fields are quite different though. The salinity field for the constant stratification case has the more straightforward salinity pattern. The isopycnals are clearly aligned with the ice base, with freshest (and coldest) water near the ice base and constant salinity at depth corresponding to the boundary values. The isopycnals seem to be more plausible than the the forward simulation in Section 4.3 because there is no change in the salinity stratification at the RHS wall, due to the homogenous Neumann boundary conditions and the absence of a sponge region. For the linearly stratified case the stratification is more complicated with an obvious 'kink' in the vertical salinity profile. This is because there is an imbalance between the bottom boundary condition for salinity (and temperature) enforcing no gradients in the vertical direction and the linear stratification on the RHS.

Figure 4.11 shows adjoint temperature and salinity 1 hour before the final time, where the

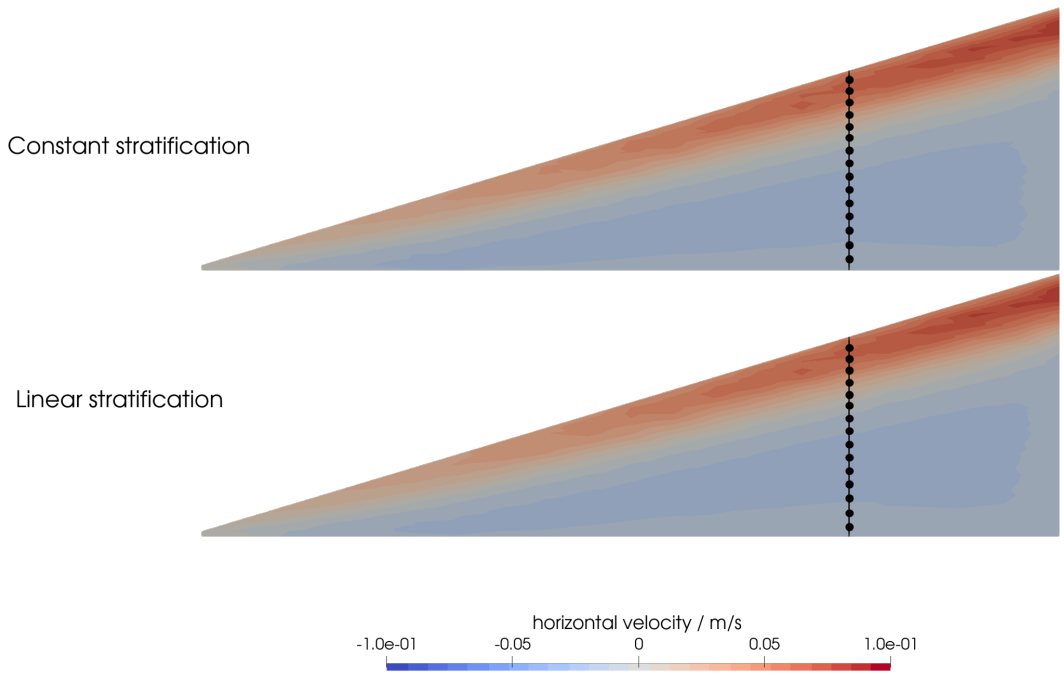


Figure 4.8: Horizontal velocity at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (top) and linear stratification (bottom). Note the black vertical line corresponds to the location of the depth profile used for the objective function and the dots indicate the location of the vertex only mesh.

sensitivities are concentrated at the location of the depth profile, shown by the black vertical lines. The sensitivity of the functional J to the scalar fields is consistent with the forward models. The initial conditions for salinity is 34.4, so the salinity at all locations of the depth profile needs to increase to improve the fit with the linear stratification and in turn reduce J. This implies that the gradient of $\partial J/\partial S$ should be negative, consistent with Figure 4.11. For the temperature field we have the opposite situation. The initial conditions of 1 °C are higher than the linearly stratified simulation. Therefore $\partial J/\partial T$ should be positive in the location of the depth profile, and this is confirmed by Figure 4.11.

Figure 4.12 shows adjoint temperature and salinity 6 hours before the final time, where the sensitivities are offset from the location of the depth profile. By 1 day before the final time the sensitivity signal has propagated away from the depth profile, as shown in Figure 4.13. The sensitivity signal close to the ice ocean interface has moved towards the grounding line, whereas the sensitivity signal at depth has moved towards the open boundary. As before, this anticlockwise circulation is reflecting the 'reversed ice pump' as referred to in Section 4.3. By 5

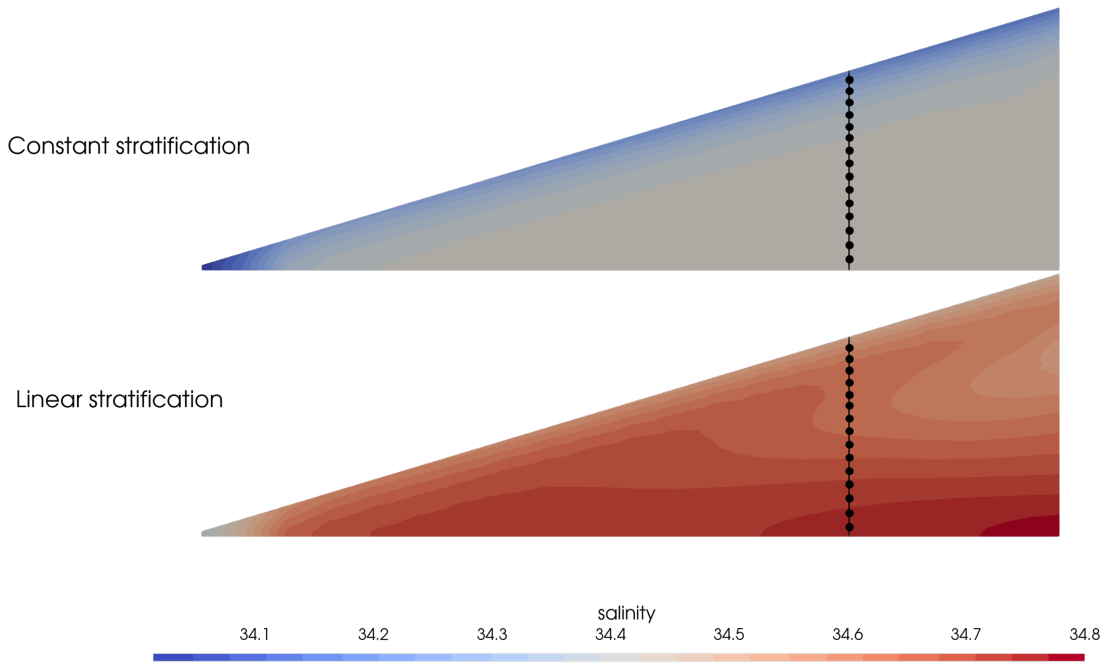


Figure 4.9: Salinity field at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (top) and linear stratification (bottom). Note the black vertical line corresponds to the location of the depth profile used for the objective function and the dots indicate the location of the vertex only mesh.

days the sensitivity to temperature and salinity has disappeared from the domain as shown by Figure 4.14. This is consistent with the simulations from Section 4.3, where the residence time was about 5 days. It is tempting to explain the sign difference between adjoint temperature and salinity in terms of buoyancy based on the Section 4.3, however, there is the added complication of initialising the temperature and salinity in opposite directions with respect to their respective target profiles. For this reason, unlike with Figure 4.4, we have not plotted the adjoint salinity and temperature scaled by the expansion and contraction coefficients from the linear equation of state.

Figure 4.15 shows the sensitivity of the objective function to boundary temperature and salinity values that are fixed in time. Note that this sensitivity is calculated with respect to the L_2 boundary norm as described in Section 4.3. The sensitivity is concentrated at the bottom depths, corresponding with the region of inflow. There is no sensitivity to the temperature and salinity boundary values in the outflow region at the top of the domain (see Figure 4.8), which gives confidence in the result. The sign of the adjoint boundary salinity field also seems to be

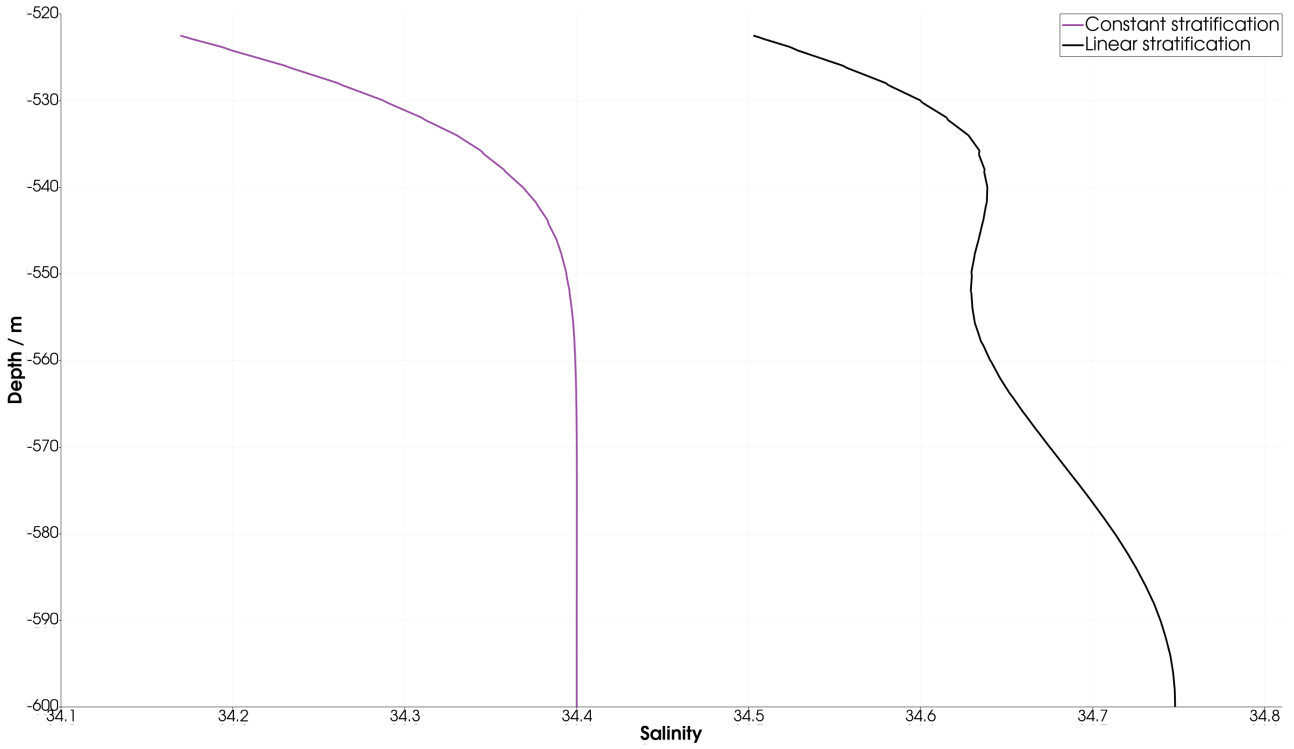


Figure 4.10: Depth profile of salinity field at 50 days for the two forward synthetic simulations with constant stratification forcing on the open boundary (purple) and linear stratification (black). Note this depth profile corresponds to the location of the depth profile used for the objective function.

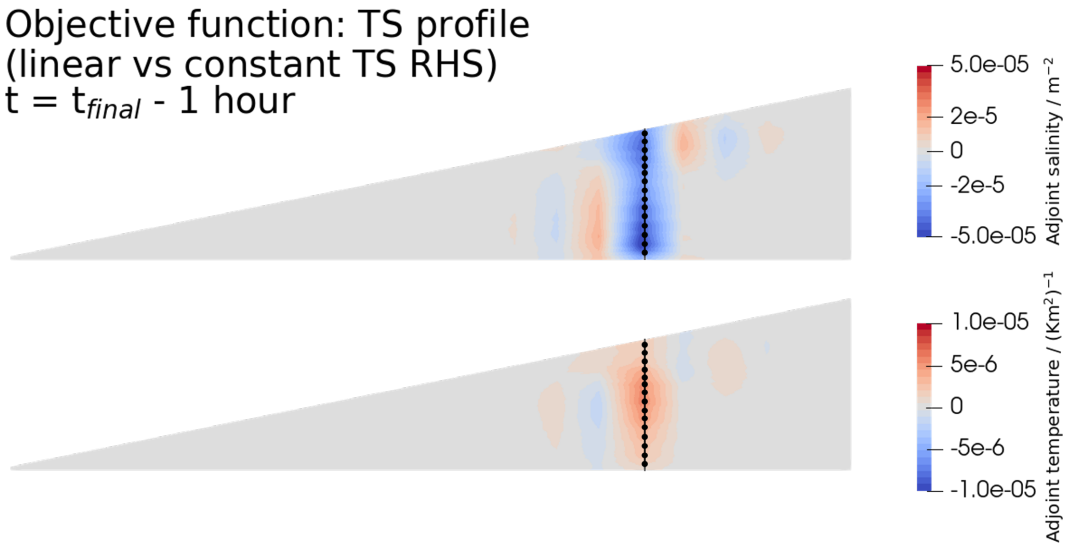


Figure 4.11: Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 1 hour before the final time calculated with respect to the L_2 norm. See Section 4.3 for an explication of units. Note the difference in colour scale between adjoint salinity and adjoint temperature.

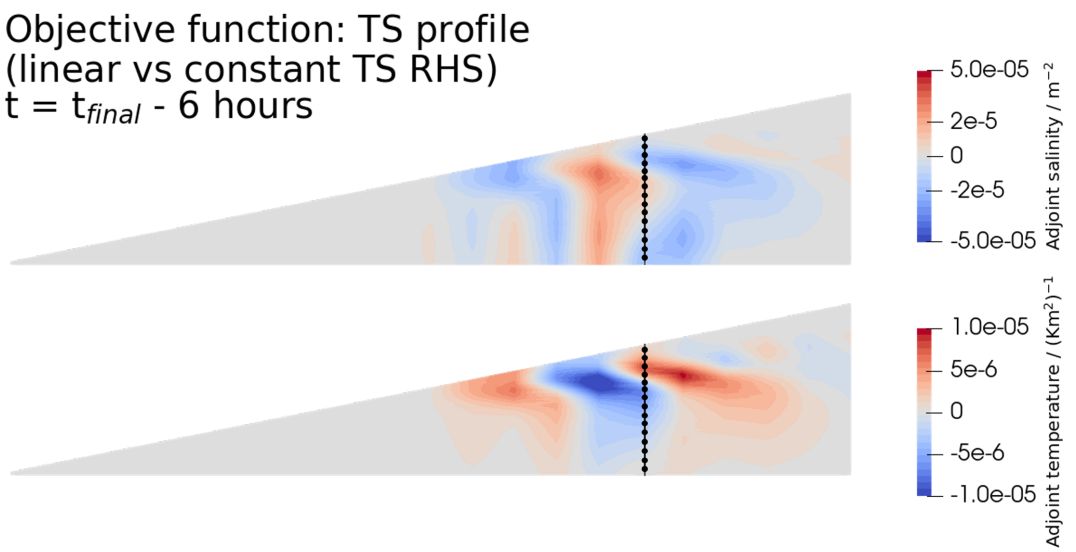


Figure 4.12: Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 6 hours before the final time calculated with respect to the L_2 norm. See Section 4.3 for an explication of units. Note the difference in colour scale between adjoint salinity and adjoint temperature.

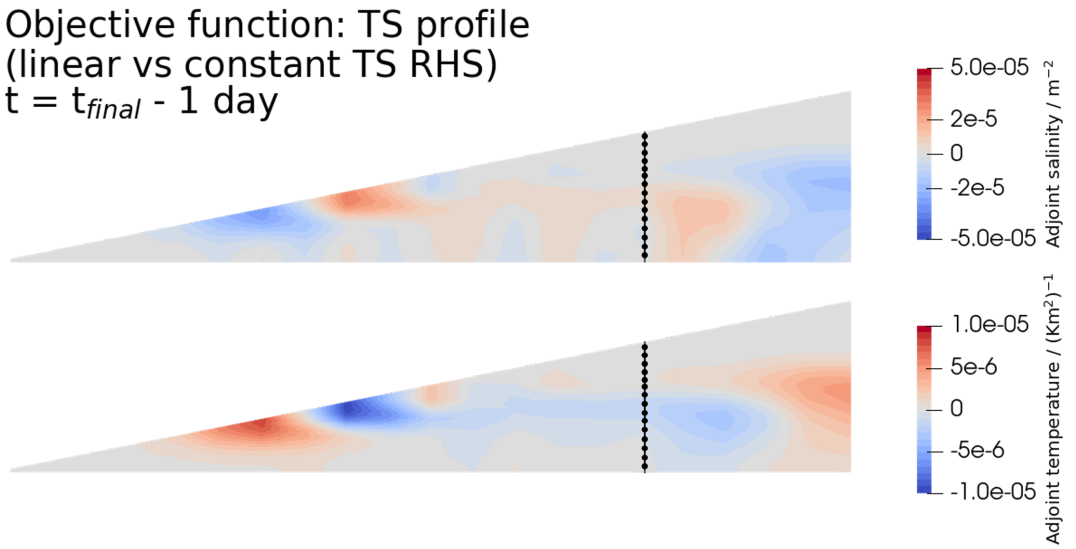


Figure 4.13: Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 1 day before the final time calculated with respect to the L_2 norm. See Section 4.3 for an explication of units. Note the difference in colour scale between adjoint salinity and adjoint temperature.

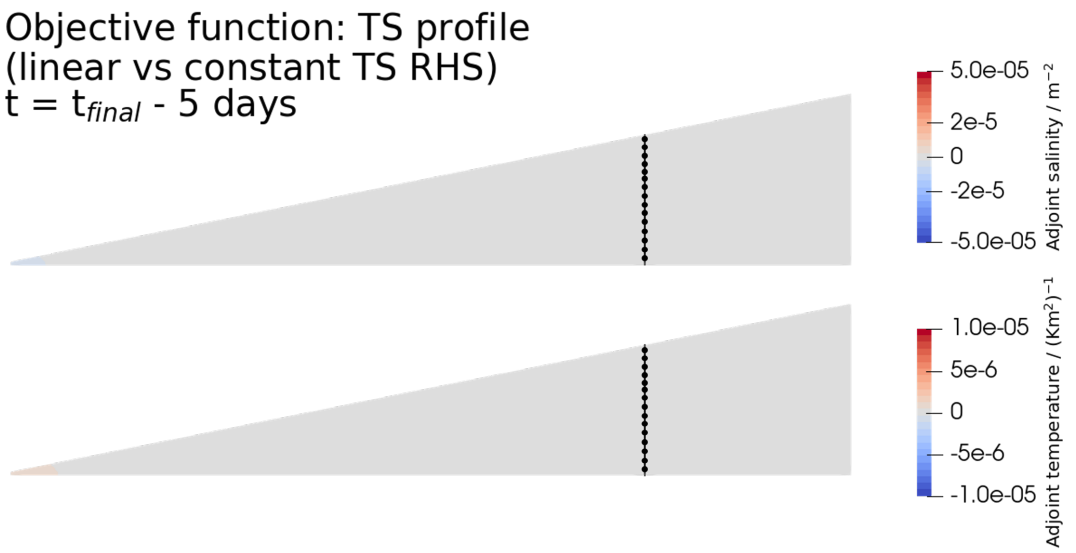


Figure 4.14: Sensitivity of objective function based on depth profile to time evolving salinity (top) and temperature (bottom) 5 days before the final time calculated with respect to the L_2 norm. See Section 4.3 for an explication of units. Note the difference in colour scale between adjoint salinity and adjoint temperature.

consistent, with negative adjoint salinity implying that increasing salinity at the boundary will reduce J as expected. This seems positive for carrying out an inversion for unknown inflow boundary conditions based on errors between an observed and modelled depth profile. Although the region of zero sensitivity at the top of the domain means that it would be impossible to constrain the values of temperature and salinity there, it is precisely the inflow region at depth that is controlling the simulation for this 2D case. This suggest that tuning boundary forcing to improve fit to a set of melt rate observations may be possible. The sign of the adjoint boundary temperature field is more complex with a positive pattern at mid depth, that is consistent with the forwards runs, but there is a negative pattern at the bottom of the domain. This may be related to the fact that at the bottom of the depth profile the initialised constant temperature of 1 °C exactly matches the target boundary value of temperature from the linear profile. The difference in sign between adjoint temperature and salinity at the bottom of the domain could also be related to a buoyancy effect given the results of Section 4.3.

Figure 4.16 shows the sensitivity of the objective function to the static vertical viscosity and

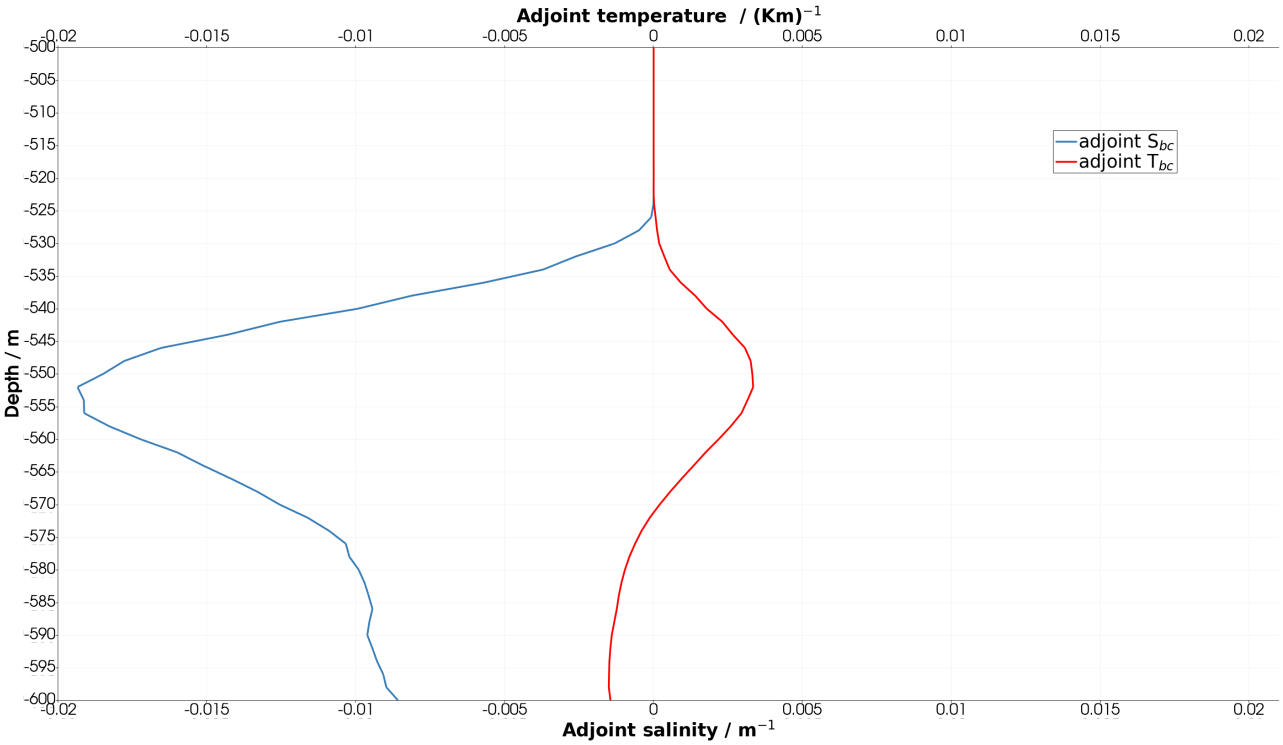


Figure 4.15: Sensitivity of objective function based on depth profile to the static salinity (blue) and temperature (red) boundary conditions on the open boundary 20 days before the final time calculated with respect to the L_2 boundary norm.

diffusivity fields 20 days before the final time. The highest sensitivities are concentrated near the depth profile where the objective function is evaluated, but have been 'smeared' out according to the overturning circulation. Similar to Figure 4.5 there is also high sensitivity near the grounding line and ice ocean interface where the gradients in flow properties are largest. Although it would be interesting to carry out an inversion based on unknown mixing values, some care would have to be taken to ensure that the localised sensitivity pattern near one depth profile does not lead to problems of over fitting.

4.5 Adjoint Performance

This section documents adjoint performance for the 2D cases in this Chapter. Table 4.4 shows that the time to solve the adjoint is about 6 times slower the forward solution time using the same parameters and mesh as in this Chapter for a simulation of ten timesteps. This is a very coarse problem ($\Delta x = 500$m, $\Delta z = 2$ m) so a significant fraction of the time is taken up during

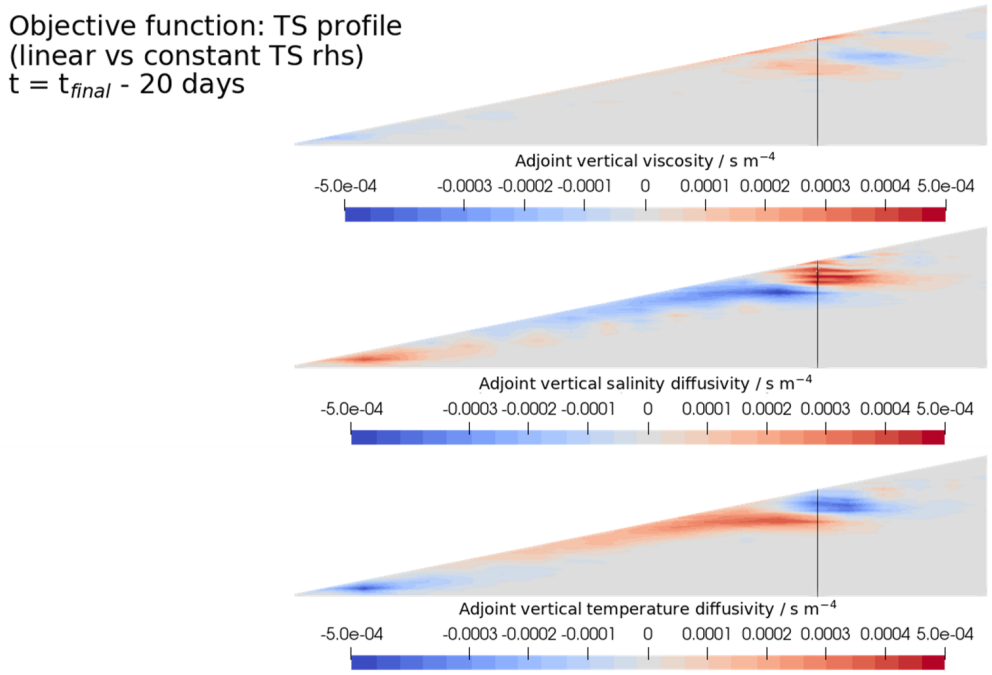


Figure 4.16: Sensitivity of objective function based on depth profile to vertical viscosity (top) and vertical diffusivity for salinity (top) and temperature (bottom) calculated with respect to the L_2 norm. See Section 4.3 for an explication of units.

assembly and symbolic operations. Table 4.5 gives the timing results using the finer mesh from Section 3.2 ($\Delta x = 50$m, $\Delta z = 1$ m). Now the solve time dominates over the assembly time and the adjoint time is only 2.6 times slower. As emphasised in the introduction to this Chapter, this is a very good result and compares favourably with a commercial, 'state of the art' adjoint implementation for MITgcm using TAF (3-6 times slower) and an open source implementation using OpenAD that is more than an order of magnitude slower [Goldberg et al., 2020]. Table 4.6 shows the timings for running the tests for 100 timesteps and confirms that the adjoint is 2.7 times slower than the forward model.

Part of the reason the adjoint solution time is slower than the forward model is that the adjoint model, as currently implemented by Pyadjoint, has no concept of timesteps. In the forward model Firedrake is able to carry out refactoring (in particular symbolic operations) during the first timestep and cache this information, so that it can be used for later timesteps. For the adjoint model, however, this must be carried out for every timestep. This pattern can be seen in rows 6 and 7 of Table 4.6. The time spent carrying out carrying out numerical operations (Row 7) is comparable for the forward and adjoint model. In contrast, the total time spent

by the forward carrying out symbolic refactoring is a factor of 100 times less than the adjoint model (1.7 s c.f. 174.8 s) because the adjoint model must carry this out for each of the 100 timesteps, as it plays through the tape in reverse. To some extent this is mitigated by using iterative solvers that do not require the same level of refactorisation. This is shown by Table 4.7 where now the adjoint solve time is only 1.4 times slower than the forward model compared with roughly 2 times slower using direct solvers. More investigation is required, but there is roughly 10 % unaccounted for time (Misc.) for all of the adjoint runs using the fine mesh. We expect for problems with more degrees of freedom, in particular in 3D where iterative solvers become necessary, the symbolic cost of the adjoint solve will become relatively small compared to the time for solving the linear systems, thus the time taken for the adjoint solution will approach the forward model.

Carrying out time dependent adjoint simulations in 3D will require using a robust checkpointing algorithm to avoid running out of memory. This is because for non-linear, time dependent models the forward the forward solution must be available to assemble the adjoint system at every timestep. When this work was carried out there was no disk checkpointing capability in Firedrake so our adjoint simulations were limited to (coarse) 2D simulations due to memory constraints. Table 4.8 shows the combined forward and adjoint memory consumption for the coarse and fine meshes used in this Chapter. Currently Firedrake implements the simplest strategy to store all adjoint variables on disk, though more sophisticated strategies combining disk storage with recomputation are in progress and we intend to use these in future work.

4.6 Code Availability

The code is available at https://github.com/thwaitesproject/thwaites.git.

Run scripts for the melt sensitivity calculations for the idealised 2D grounding zone domain (Section 4.3):

- thwaites/run_scripts_testing/ice_shelf_adjoint/ice_shelf_coarse.py

Variable	Assembly (s)	Solve (s)	Misc. (s)	Total (s)	Symbolic (s)	Numeric (s)
$\underline{\mathbf{u}} - \underline{p}$	1.4	0.4	0.3	2.1	0.05	0.3
$\underline{T}$	0.9	0.1	0.2	1.2	0.008	0.1
$\underline{S}$	0.4	0.2	0.1	0.7	0.004	0.1
Forward	2.7	0.7	0.6	4.0	0.062	0.5
Adjoint	5.9	2.8	6.2	24.9	0.4	0.56

Table 4.4: Adjoint performance for simulations using the coarse mesh ($\Delta x = 500$m, $\Delta z = 2$ m) run for ten timesteps. Wall clock times are given in seconds for the time taken to solve for each variable. $\underline{\mathbf{u}} - \underline{p}$ is the coupled velocity-pressure system for the Momentum equation (with the incompressibility constraint), $\underline{T}$ and $\underline{S}$ are the discrete solutions for temperature and salinity. The total time is split into assembly time, solve time and miscellaneous time (Total - (Solve + Assembly)). For these simulations MUMPS, a direct LU solver, is used for solving each variable. Symbolic time refers to time spent carrying out symbolic refactoring by MUMPS. Numeric time refers to the time spent carrying out numerical operations by MUMPS.

Variable	Assembly (s)	Solve (s)	Misc. (s)	Total (s)	Symbolic (s)	Numeric (s)
$\underline{\mathbf{u}} - \underline{p}$	3.1	13.3	2.3	18.7	1.1	11.4
$\underline{T}$	1.8	4.3	0.4	6.5	0.4	3.3
$\underline{S}$	1.4	4.3	0.2	5.9	0.3	3.2
Forward	6.3	21.9	2.9	31.1	1.8	17.9
Adjoint	29.7	41.4	8.9	80	17.9	18.8

Table 4.5: Adjoint performance for simulations using the fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m) run for ten timesteps ($\Delta t = 150$ s). Wall clock times are given in seconds for the time taken to solve for each variable. $\underline{\mathbf{u}} - \underline{p}$ is the coupled velocity-pressure system for the Momentum equation (with the incompressibility constraint), $\underline{T}$ and $\underline{S}$ are the discrete solutions for temperature and salinity. The total time is split into assembly time, solve time and miscellaneous time (Total - (Solve + Assembly)). For these simulations MUMPS, a direct LU solver, is used for solving each variable. Symbolic time refers to time spent carrying out symbolic refactoring by MUMPS. Numeric time refers to the time spent carrying out numerical operations by MUMPS.

Variable	Assembly (s)	Solve (s)	Misc. (s)	Total (s)	Symbolic (s)	Numeric (s)
$\underline{\mathbf{u}} - \underline{p}$	17.7	121.7	2.7	142.1	1.1	112.6
$\underline{T}$	11.4	39.4	1.1	51.9	0.3	32.4
$\underline{S}$	10.9	39.4	1.0	51.3	0.3	32.3
Forward	40.0	200.5	4.8	245.3	1.7	177.3
Adjoint	196.2	399.9	66.9	663.0	174.8	178.9

Table 4.6: Adjoint performance for simulations using the fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m) run for 100 timesteps ($\Delta t = 150$ s). Wall clock times are given in seconds for the time taken to solve for each variable. $\underline{\mathbf{u}} - \underline{p}$ is the coupled velocity-pressure system for the Momentum equation (with the incompressibility constraint), $\underline{T}$ and $\underline{S}$ are the discrete solutions for temperature and salinity. The total time is split into assembly time, solve time and miscellaneous time (Total - (Solve + Assembly)). For these simulations MUMPS, a direct LU solver, is used for solving each variable. Symbolic time refers to time spent carrying out symbolic refactoring by MUMPS. Numeric time refers to the time spent carrying out numerical operations by MUMPS.

Variable	Assembly (s)	Solve (s)	Misc. (s)	Total (s)
$\underline{\mathbf{u}} - \underline{p}$	14.1	173.1	1.4	188.6
$\underline{T}$	15.7	10.0	1.0	26.7
$\underline{S}$	15.1	10.2	1.2	26.5
Forward	44.9	193.3	3.6	241.8
Adjoint	163.8	273.0	45.0	481.8

Table 4.7: Adjoint performance for simulations using the fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m) run for 100 timesteps ($\Delta t = 150$ s) using iterative solvers. Wall clock times are given in seconds for the time taken to solve for each variable. $\underline{\mathbf{u}} - \underline{p}$ is the coupled velocity-pressure system for the Momentum equation (with the incompressibility constraint), $\underline{T}$ and $\underline{S}$ are the discrete solutions for temperature and salinity. The total time is split into assembly time, solve time and miscellaneous time (Total - (Solve + Assembly)). For these simulations, the momentum equation is solved using a predictor-corrector approach as described in Section 2.5. The prediction step for the momentum equation is solved using GMRES with BoomerAMG from the HYPRE suite as the preconditioner. The pressure projection correction step is solved using the Schur complement factorisation approach as described in Section 2.5. $\underline{T}$ and $\underline{S}$ use GMRES with SOR (Successive Over Relaxation) as a preconditioner.

Simulation	Forward (GiB)	Adjoint (GiB)	Total (GiB)
'Coarse', 10 steps	0.25	0.25	0.5
'Fine', 10 steps	1.7	1.5	3.2
'Fine', 100 steps	2.4	3.5	5.9
'Fine', 100 steps (iterative)	2.0	10.2	12.2
'Fine', 200 steps	3.4	5.4	8.8
'Fine', 400 steps	5.4	9.3	14.6

Table 4.8: Adjoint memory consumption for simulations using the coarse ($\Delta x = 500$m, $\Delta z = 2$ m) and fine mesh ($\Delta x = 50$m, $\Delta z = 1$ m). The timestep is 300 s for the coarse simulation and 150 s for the fine simulations. Memory is recorded in Gibibytes.

Run scripts for the scalar depth profile sensitivity calculations for the idealised 2D grounding zone domain (Section 4.4):

- thwaites/run_scripts_testing/ice_shelf_adjoint/ice_shelf_coarse_open.py

4.7 Conclusion

This chapter demonstrated preliminary adjoint simulations in an ice shelf ocean context, using the total melt rate as an objective function. The adjoint clearly captures the reverse overturning 'ice pump' mechanism for an idealised grounding zone domain, giving a complementary view to the forward model. Owing to the difference in sign between the adjoint salinity and adjoint temperature, the adjoint results suggest buoyancy forcing is a more important control than thermodynamics. The adjoint model consistently showed high sensitivity at the grounding zone and ice ocean interface for the diffusivity and viscosity fields, consistent with the regions in the forward model with highest stratification. We found that the turbulent exchange coefficient from the melt parameterisation, Γ_T, is approximately linearly proportional to the melt rate, which seems plausible given the linear dependence of the turbulent heat flux on Γ_T. This first step shows that inverting for the unknown spatially variant Γ_T field could be feasible based on an objective functional measuring the misfit between modelled and observed melt rates, e.g. from satellite observations. We also showed initial results using an objective function measuring the difference between two synthetic forward simulations to calculate the sensitivity with respect to a range of parameters, including the forcing conditions on the open boundary. This is in anticipation of carrying out an inversion for unknown boundary conditions in order to best capture an observed stratification profile and thus improve understanding of mixing processes beneath ice shelves.

Although much of this work is preliminary, the simulations carried out show there is a solid framework to build on. We have demonstrated the ability of Firedrake to automatically calculate sensitivities of different functional choices with respect to a range of input parameters, including boundary conditions and not just solution fields from the forward model. In ad-

dition, we have emphasised the importance of interpreting the sensitivity patterns using the right norm, i.e. the L_2 and L_2 boundary norm and using the correct units, which is essential for accurate quantitative assessments of sensitivity. This is especially true for discretisations where the mesh resolution is non uniform. Due to the flexibility of the adjoint implementation with Firedrake, future changes to the forward code, for instance improved melt or turbulence parameterisations, can be integrated seamlessly to the adjoint model, whilst maintaining good computational performance. Given the unknowns particularly in the formulation of melt and mixing within the cavity, there is much potential to use this adjoint capability to investigate and calibrate these unknown parameters in an efficient manner.

Chapter 5

Modelling 3D Ice Shelf Basal Crevasses

Highlights

- We show good agreement with previous Fluidity results for a dynamic 2D crevasse simulation, driven by frazil ice dynamics.
- We extend the simulation domain to 3D to consider effects of crevasse orientation and the Coriolis effect and show that there is an asymmetry in the melt rate patterns.
- Melting within the crevasse, however, remains suppressed compared with background levels, confirming the 2D finding that the ocean likely acts to smooth the ice base.

Summary

This chapter applies the forward model presented in Chapter 2 to an interesting and open area of ice-ocean research: the formation and evolution of basal crevasses and their effect on ocean circulation and melting under ice shelves. There is an initial, positive, comparison with Fluidity of a 2D basal crevasse, including frazil ice dynamics. This work is then extended to consider 3D geometry, to move beyond the inherent assumption that the flow hits the crevasse at right angles. By allowing the flow angle to vary, we investigate the effect of crevasse orientation and

the Coriolis effect on the flow and the melt rate within and outside the crevasse. A working hypothesis was that this might reduce the build up of stratification within the crevasse and potentially lead to higher melt rates, giving a possible stabilising, or even formation, mechanism for crevasses. The sloping side walls do give rise to higher melt on the upstream side in all cases, as well as producing an asymmetric melting pattern when the flow has a significant component along the crevasse orientation. Melt rates within the crevasse, however, remain lower than the background, implying the ocean still acts to smooth the ice base.

Publications and Disclaimer

This work has not been published.

5.1 Introduction

Investigations by Automatic Underwater Vehicles (AUV) equipped with upward looking sonar systems have shown that the landscape of basal features is far from flat. Crevasses, channels and curious step features, referred to as basal terraces, are common features found on a number of different ice shelves [Dutrieux et al., 2014]. There has been a lot of attention to basal channels in the last few years on the modelling side [Millgate et al., 2013, Sergienko, 2013] and on the observation side [Alley et al., 2016, 2023]. In some cases the basal channels are clearly identifiable from satellites with features running back to the grounding line suggesting subglacial hydrology origins [Alley et al., 2016]. For the purposes of the investigation here, we consider that crevasses do not have a noticeable slope along the ice-ocean interface due to cutting the ice shelf perpendicular to the ice flow direction.

Despite being such common features observed under ice shelves, simple questions remain about how these crevasses form, what effect they can have on ocean circulation (and melt rates) and ultimately on ice shelf stability. Part of the reason for this is that they are very difficult to study, both from an observation and modelling side, particularly using traditional structured

grid models as alluded to in Chapter 1. Terrain following meshes struggle to represent sharply varying topography and are prone to instabilities if the angle of the cells becomes too high, referred to as the Haney number [Haney, 1991, Schnaase and Timmermann, 2019]. z-coordinate models are better able to capture vertical walls, but partial cells does not always lead to the best treatment of sloping boundaries. Moreover, if the hydrostatic approximation is made then as the cells approach order one aspect ratio the model no longer faithfully represents the Navier-Stokes equations with vertical acceleration terms.

Fluidity, a finite element nonhydrostatic ocean model, was applied to crevasses by Jordan et al. [2014]. They considered a 2D vertical slice through an idealised rift based on the Jutulgryta rift on Fimbulisen ice shelf. Their investigation focused on the effects of frazil ice on circulation within a crevasse. By forcing the model with different inflow temperature the depth of the freezing point was raised and lowered thus varying the impact of frazil ice production in the domain. Yeager [2018] built on this work using moving meshes to allow the crevasse geometry to deform in response to melting, though the study did not consider the frazil ice case.

The presence of crevasses poses an interesting puzzle. On the ocean modelling side, the results from Jordan et al. [2014] clearly show that the ocean is acting to erode the crevasses and produce a smooth ice base. For freezing conditions the formation of frazil ice fills in the crevasse. For melting conditions, due to the build up of stratification within the crevasse and the depth dependent freezing point of water, melting is higher outside the crevasse than inside, thus implying again that the crevasse will be eroded. On the ice side crevasses can form in the presence of fracture in high stress regimes in the shear margins and at the grounding zone [Alley et al., 2021, Benn et al., 2022], but there are ice feedback mechanisms referred to as 'necking' which also act to smooth the ice base flat [Bassis and Ma, 2015].

A study by Watkins et al. [2021] showed that there was a positive correlation between melt rates and rougher ice shelves, as measured by radar surveys. It seems intuitive that crevasses and terraces act to promote turbulence and therefore break down stratification at the ice base leading to higher melting. They also found that there is no 'peak' frequency corresponding to a characteristic wavelength feature, possibly implying that the distinction between crevasses and

channels is unhelpful. Indeed, Vaughan et al. [2012] and Dow et al. [2018] suggest formation mechanisms of crevasses that are linked to the formation of basal channel due to enhanced melting. Although the problem of crevasse formation and evolution is an interesting problem the aim of this chapter is to focus on melting within static crevasses and their effect on surrounding ocean dynamics. We build on the work by Jordan et al. [2014] to extend the 2D geometry to 3D to consider rotational effects. The simulation domains are based on recent observations of crevasses near the Thwaites grounding zone [Schmidt et al., 2023].

5.2 Frazil Ice Dynamics

We have implemented a frazil ice model based on the work of Jordan et al. [2014]. Frazil ice contributes to the density, ρ of the water parcel according to

$$\rho = \rho_0(1 - C)(1 + \rho') + \rho_i C, \tag{5.1}$$

where C is the fraction of frazil ice, ρ_0 is the bulk density of sea water, ρ_i is the density of ice and the density perturbation, ρ', is given by

$$\rho' = -\alpha_T(T - T_{ref}) + \beta_S(S - S_{ref}). \tag{5.2}$$

S_{ref} and T_{ref} are the expansion coefficients from the linear equation of state. Temperature, T, is an active tracer governed by an advection diffusion equation

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \nabla \cdot \kappa_T \nabla T + \left(T_c - T - \frac{L}{c_p} \right) w_c. \tag{5.3}$$

The source term on the right hand side of Equation 5.3 accounts for the latent heat release when frazil ice crystals form. w_c is the melt rate for frazil ice crystals ($w_c < 0$ is freezing). T_c is the temperature at the interface between the frazil ice and the ocean. L is the latent heat of

fusion of water and c_p is the specific heat capacity of water. Similarly salinity, S, evolves as

$$\frac{\partial S}{\partial t} + \mathbf{u} \cdot \nabla S = \nabla \cdot \kappa_S \nabla S - S w_c. \tag{5.4}$$

The frazil ice is also modelled by an advection diffusion equation with an additional advection term that causes the ice to rise, equivalent to a negative 'sinking' velocity used in sediment models given by

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C + w_i \frac{\partial C}{\partial z} = \nabla \cdot \kappa_C \nabla C - w_c. \tag{5.5}$$

As described by Jordan et al. [2014] the rising velocity w_i is given by

$$w_i^2 = \frac{4Rgr\epsilon}{C_d}, \tag{5.6}$$

where g is gravity, r is the radius of the crystal, ϵ is the aspect ratio of the crystal and R is the specific gravity given by

$$R = \frac{\rho_i - \rho_0}{\rho_0}. \tag{5.7}$$

The drag coefficient C_d is given by Jordan et al. [2014], following Gosink and Osterkamp [1983] as

$$\log_{10}(C_d) = 1.386 - 0.892 \log_{10}(Re) + 0.111(\log_{10}(Re))^2, \tag{5.8}$$

where the disk Reynolds number, Re, is given by

$$Re = \frac{w_i 2r}{\nu}, \tag{5.9}$$

where ν is the viscosity of sea water. Equations 5.6 - 5.9 are solved iteratively to give a unique rising velocity for a given radius of frazil ice crystal.

The freezing rate due to frazil ice is taken to be the flux of frazil ice through the top boundary given by

$$\frac{\partial \eta}{\partial t} = w_b C_b, \tag{5.10}$$

where w_b is the velocity of the frazil ice at the top boundary and C_b is the concentration at the

top boundary. Melt and freezing rates of frazil ice are calculated based on conservation of heat and salt at the frazil interface and that the temperature at the frazil interface is at the freezing point. The conservation of heat is given by

$$(1-C)\gamma_T^c(T-T_c)\frac{2C}{r} = \frac{L}{c_p}w_c, \tag{5.11}$$

where γ_T^c is the thermal exchange coefficient and is given by

$$\gamma_T^c = \frac{Nu\kappa_T}{\epsilon r}. \tag{5.12}$$

Nu is the Nusselt number taken here to be 1 following Jordan et al. [2014] and κ_T is the molecular thermal diffusivity of sea water.

The conservation of salt is given by

$$(1-C)\gamma_S^c(S-S_c)\frac{2C}{r} = S_c w_c, \tag{5.13}$$

where γ_S^c is the equivalent salt exchange coefficient, given by

$$\gamma_S^c = \frac{Nu\kappa_S}{\epsilon r}, \tag{5.14}$$

and κ_S is the mass diffusivity of sea water.

The freezing point is given by

$$T_c = aS_c + b + cp_c, \tag{5.15}$$

where a, b and c are constants obtained from the linearisation. Importantly the freezing point depends on pressure so frazil ice is more likely to form near the top of crevasses.

5.3 2D Crevasse Simulation with Frazil Ice Dynamics

To build confidence in the model and check that the frazil ice scheme was implemented correctly we ran the baseline setup from Jordan et al. [2014]. This a dynamic setup that is a good demonstration of the ability to resolve complex overturning flow robustly on unstructured meshes. Jordan et al. [2014] carried out an investigation of an idealised 2D cavity based on the Jutulgryta rift in the Fimbulisen ice shelf. The study site was chosen because observations of temperature and salinity were available, collected by Orheim et al. [1990] and republished by Khazendar and Jenkins [2003], who also used the observations to test their plume model. An approximate freezing rates of 1 m/a was estimated by Orheim et al. [1990] due to 2 m of ice build up when retrieving the instruments two years later.

The setup is intended to match the baseline case from Jordan et al. [2014]. The rift is 340 m wide and 260 m deep. The top of the rift is 40 m below the surface. Beneath the crevasse the ocean cavity is 100 m thick with a flat ice base 300 m deep and a flat seabed 400 m deep. The total horizontal extent of the domain is 5 km. The mesh was generated using Gmsh [Geuzaine and Remacle, 2009]. The mesh is made up of triangles with variable resolution. The resolution is 5 m within the crevasse and becomes coarser outside the crevasse to 25 m. No normal flow is imposed on the top and bottom of the domain. There is an inflow on the left boundary of 0.025 m/s and on the right boundary is open. Flux boundary conditions for temperature and salinity are imposed along the ice base (including the verticals walls inside the crevasse). The fluxes are calculated using the 'three-equation' melt rate parameterisation as described in Section 2.6.

Temperature and salinity are initialised at the baseline values from Jordan et al. [2014]: -1.965 °C and 34.34 respectively. This puts the depth dependent freezing point within the crevasse and ensures that frazil ice forms within the cavity. These values are also used as Dirichlet boundary conditions on the left-hand inflow boundary. All boundary conditions are imposed weakly. Fluidity, the fluid dynamics code used by Jordan et al. [2014], is also a Finite Element model and solves the same underlying equations used here: the incompressible (nonhydrostatic) momentum equations, tracer advection diffusion schemes and frazil ice thermodynamics outlined in Section 5.2. The velocity and pressure are discretised using the P1DG-P2 finite element pair,

which is the same as used by Jordan et al. [2014], however, the tracers are discretised using P1DG and a vertex based limiter [Kuzmin, 2010], as opposed to control volume finite element method used in Jordan et al. [2014]. We use a constant viscosity and diffusivity of 1×10^{-3} m^2/s. The timestep is 5 s, following Jordan et al. [2014].

The qualitative agreement between Fluidity and our model is very good, especially considering this a complex, dynamic buoyancy driven flow. Our model follows the same initial spin-up phase. Initially melting at the ice base beneath the crevasse forms a cold, fresh layer, as well as very small quantities of frazil ice above the depth dependent freezing point at the top of the crevasse. Since this melt layer is buoyant and the background flow is to the right, the initial flow within the cavity is upwards along the left vertical wall. When the buoyant melt water reaches the top of the crevasse, above the freezing point, frazil ice production increases rapidly. This leads to dense water formation causing sinking. This complicated overturning flow evolves into a fairly steady counterclockwise flow by the third week of simulation that can be seen in both our results in Figure 5.1 and those of Jordan et al. [2014] in Figure 5.2 (reproduced from Figure 5 in Jordan et al. [2014]). Figure 5.1 shows the velocity and frazil ice concentration time averaged over the last week, along with the frazil flux through the top boundary averaged over the last week. Similar to Jordan et al. [2014] we find that frazil production is much more dominant than direct freeze on to the ice base. We find an average flux of 9.5 m/a through the top wall compared to 8.3 m/a from Jordan et al. [2014]. The shape of the frazil flux distribution is also similar with the most frazil occurring in the two corners of the domain, with less freezing in the centre. The freezing through the centre is 2.5 m/a, compared to 1 m/a estimated by Orheim et al. [1990] and 1.2 m/a reported by Jordan et al. [2014]. Compared to Jordan et al. [2014] there is more freezing concentrated on the right hand side. We also see the observed 60 m of supercooled water at the top of the domain when the vertical profile is taken through the 'calibration' profile as shown by Figures 5.3 and 5.4. However, at the edges due to the production of frazil ice the water temperature is closer to the in situ freezing point.

The average freezing rate is quite sensitive to resolution, perhaps unsurprisingly given the dynamic nature of the simulation. The freezing rate with grid resolution of 2.5 m is 7.2 m/a, which is lower than the Fluidity result. Conversely, coarsening the grid to 10 m gives a freezing

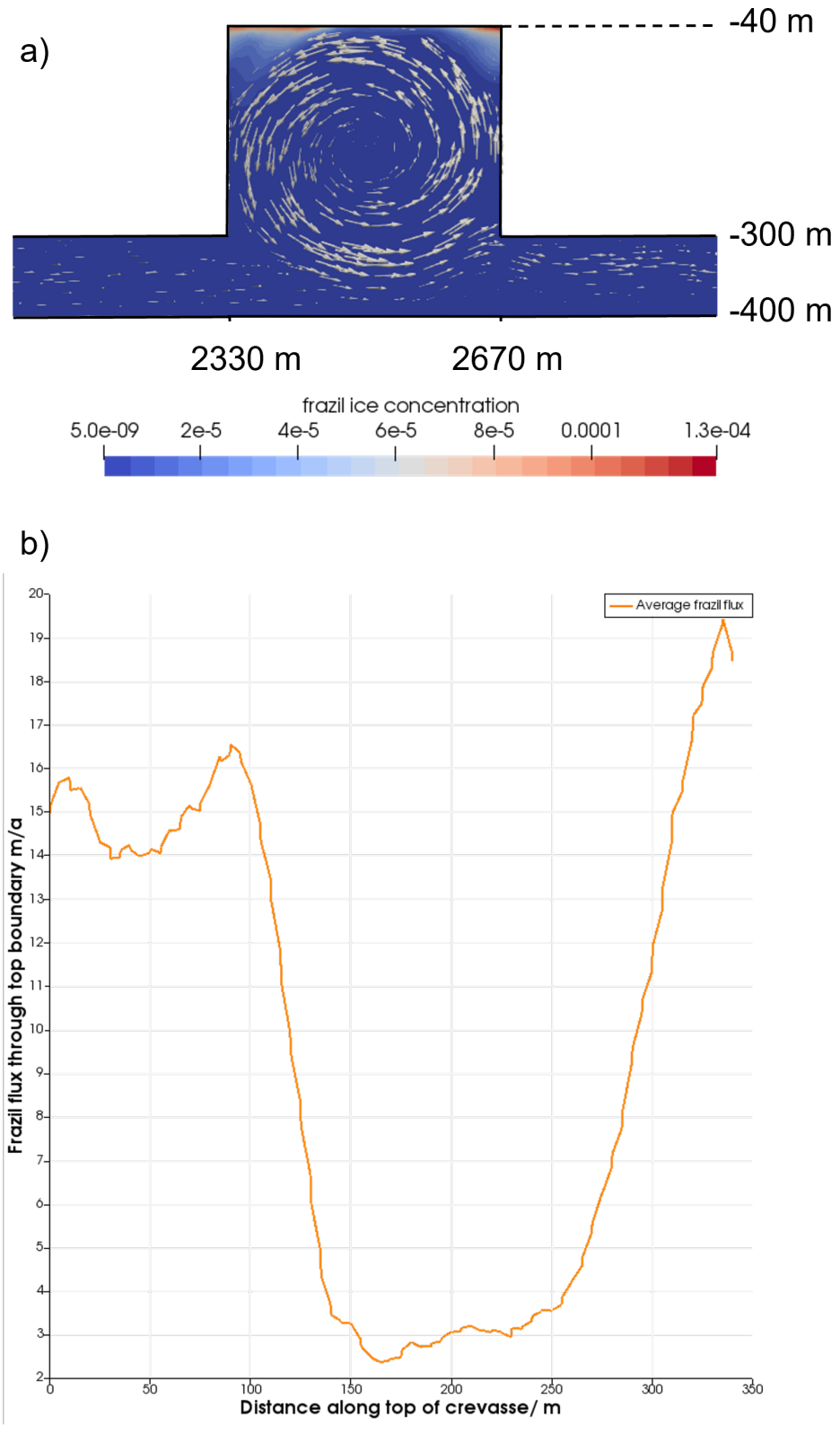


Figure 5.1: Time averaged plots of frazil ice concentration and velocity vectors for the 2D crevasse based on the study by Jordan et al. [2014]. a) In the top panel the counterclockwise overturning circulation due to the frazil ice formation, concentrated at the top corners of the crevasse, is clearly visible. b) The bottom panel gives the (time averaged) flux of frazil ice through the top boundary in the crevasse.

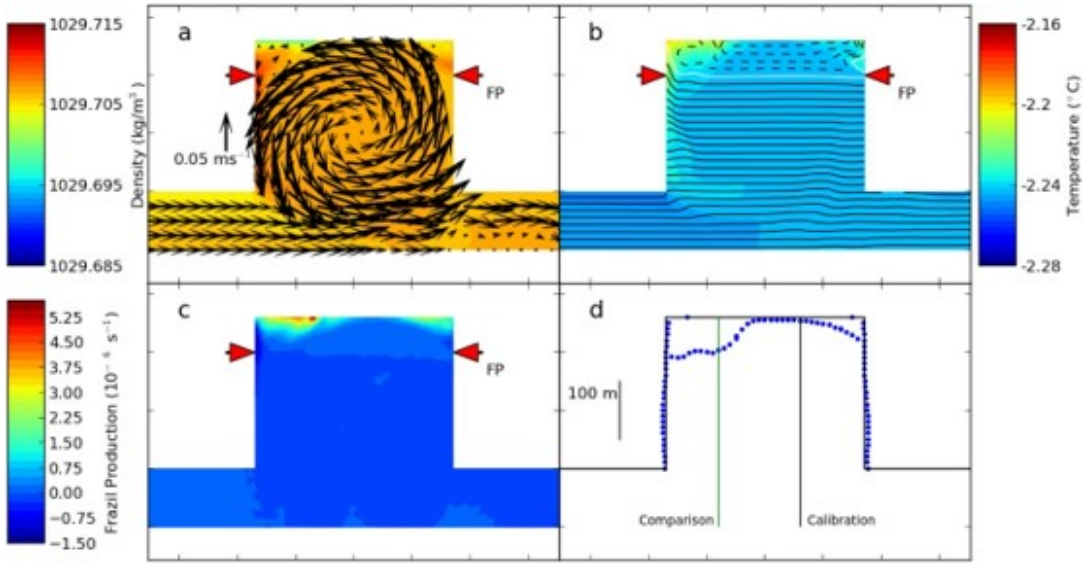


Figure 5.2: Frazil ice simulations from carried out by Jordan et al. [2014] (Figure 5). Note the same counterclockwise circulation as our results in Figure 5.1, as well as Frazil ice production concentrated in the crevasse corners.

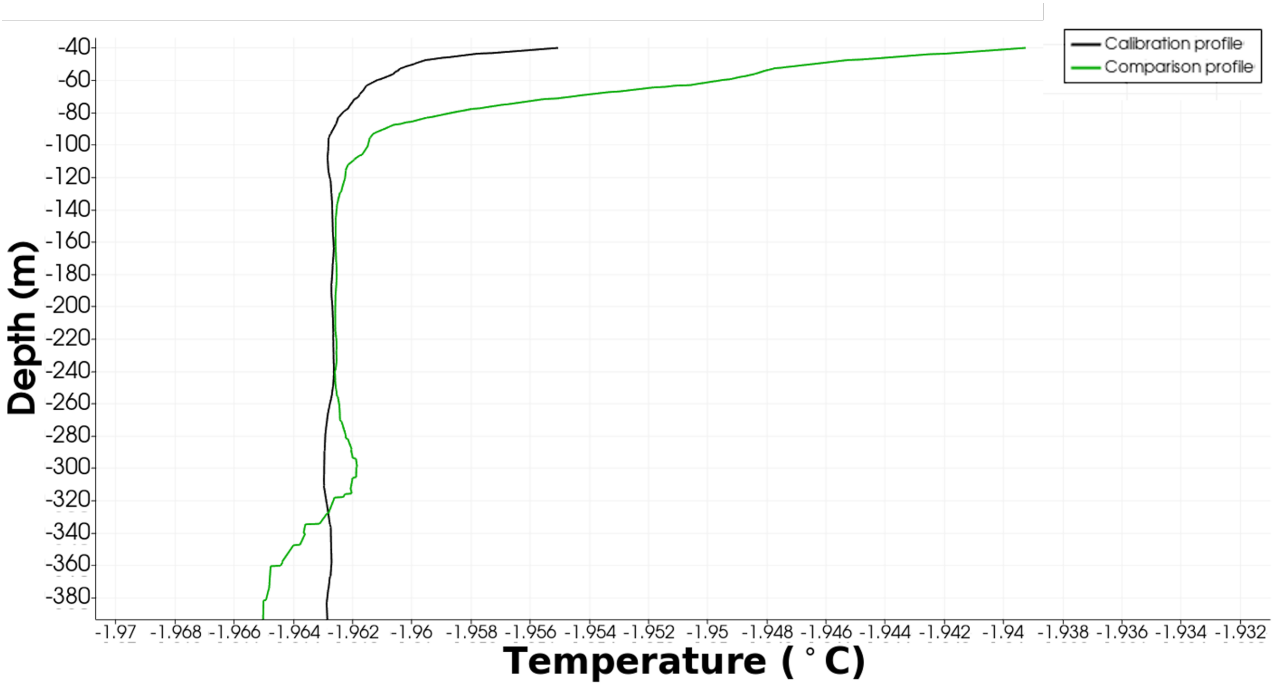


Figure 5.3: Temperature depth profile at 'comparison' and 'calibration' profile locations from Jordan et al. [2014] (shown in Figure 5.4). Note the super cooling in the 'calibration' profile plot. See Figure 5.2 for the location of the depth profiles.

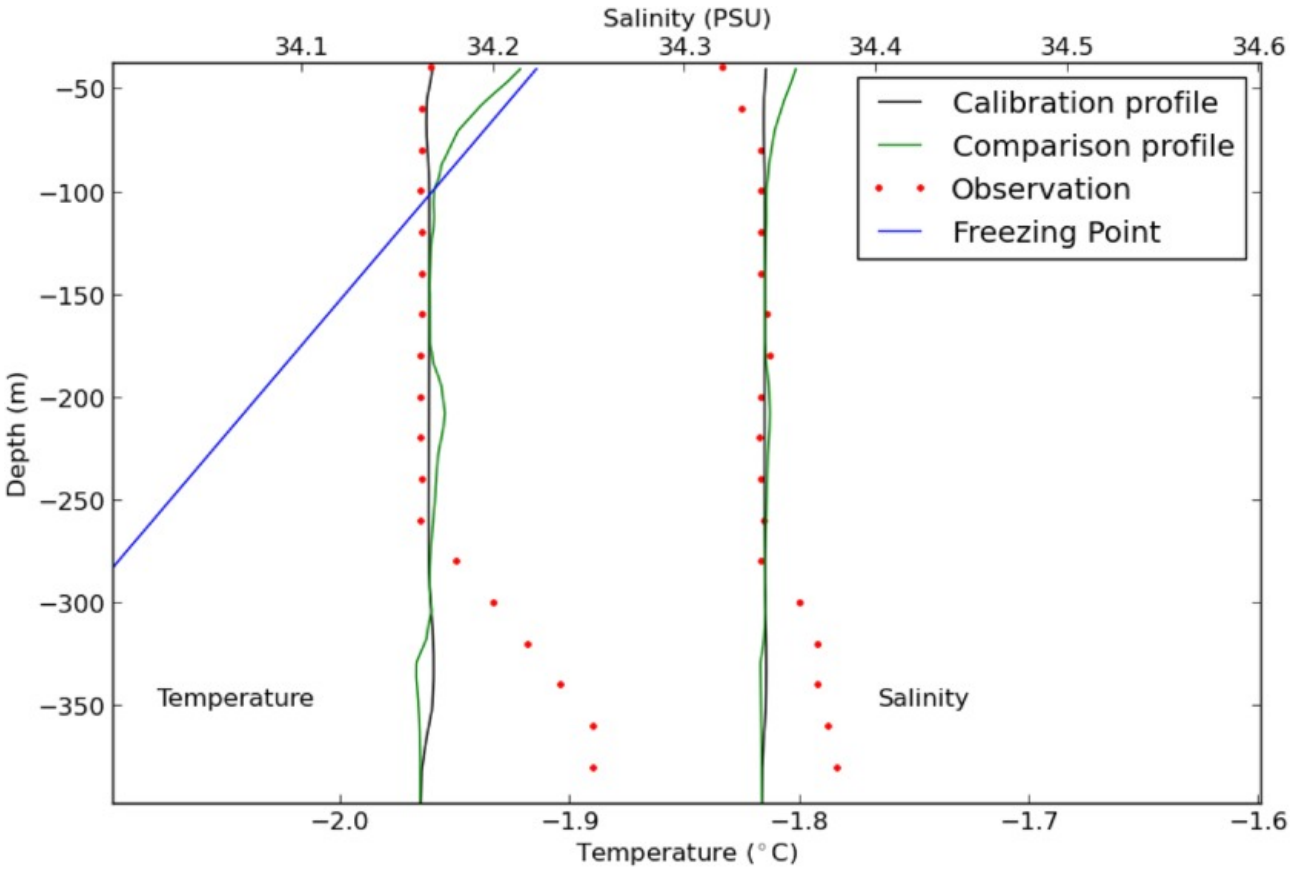


Figure 5.4: Temperature depth profile at 'comparison' and 'calibration' profile locations from Jordan et al. [2014] (reproduced from Figure 3). Note the super cooling in the 'calibration' profile plot. See Figure 5.2 for the location of the depth profiles.

rate of 11.1 m/a, so the Firedrake simulations do not seem to be approaching the Fluidity results. Jordan et al. [2014] did not give sensitivity of the results to resolution so it is difficult to make direct comparisons. One might imagine that the 10 m P1DG discretisation would be closer to the control volume discretisation with 5 m resolution, because P1DG is generally more accurate, but less diffusive than a control volume method [Hiester, 2012], but this does not seem to be case. The time discretisation is probably also an important factor and a simulation with dt = 60 s ($\Delta x = 5$ m) gave 7.1 m/a of freezing. This suggests a possible trend where refining the grid resolution decreases the freezing rate, but refining the time step increases the freezing rate. Owing to not having information about the grid and time step sensitivity from the Jordan et al. [2014] study and given that our results are qualitatively in very good agreement, a decision was taken to extend the Jordan et al. [2014] work to move forward. A limitation of the Jordan et al. [2014] was that the 2D geometry restricts the flow to hitting the crevasse at right angles. The next section considers 3D geometries to investigate the effect of crevasse orientation on the flow and melt rates.

5.4 Geometry and Rotational Effects for 3D Basal Crevasses

The experiments were designed to extend the work done by Jordan et al. [2014] to investigate the effect of 3D crevasse geometry and possible impact of the Coriolis effect. For each experiment the simulation domain is a 5 km by 5 km by 100 m box, 600 m deep. A 50 m high crevasse, 200 m wide at the base, narrowing to 50 m wide at the top is orientated within the domain. The geometry of this crevasse was inspired by observations from the Thwaites grounding zone made using the Icefin robot [Schmidt et al., 2023]. Each experiment is forced by a uniform body force in the y-direction with a magnitude $|fu_{\text{geo}}|$, where f is the Coriolis parameter, $f = -1.409 \times 10^{-4}$ s^{-1} and u_{geo} is a characteristic speed of 0.01 m/s. This body force is equivalent to a pressure gradient that will balance with the Coriolis term and thus drive a geostrophic current in the negative x-direction. The force is only applied in the ocean domain beneath the crevasse. During an investigation of boundary conditions the forcing cut-off depth had little impact on the flow within the crevasse. For this setup, with a flow of 0.01 m/s, a

crevasse width of 200 m and Coriolis parameter of 1×10^{-4} s^{-1}, the Rossby number is ~ 0.5, implying that the Coriolis term is about twice as large as the nonlinear advection term. The baroclinic Rossby radius is another measure of the importance of rotation in stratified fluids and is given by

$$L_R = NH/(\pi n f), \tag{5.16}$$

where N is the buoyancy frequency, H is the depth and n is 1 for the first radius. The buoyancy frequency is given by

$$N^2 = -\frac{g}{\rho_0}\beta_S \partial\rho/\partial z \approx -g\beta_S \Delta S/H, \tag{5.17}$$

where $\Delta S \sim -0.13$ is a salinity difference from the bottom to the top of the domain (see setup below), $\beta_S = 7.8 \times 10^{-4}$ and $g = 9.81$ then L_R is approximately 700 m. This implies that wake disturbances should not be dominated by the rotation term since the crevasse width is only 200 m.

There are four crevasse orientations. The *across* flow experiment has the crevasse aligned perpendicular to the ambient flow direction. This is equivalent to the 2D vertical slice considered by Jordan et al. [2014]. The *along* flow experiment rotates the crevasse 90° so that the background flow is now in line with the crevasse. The *positive* and *negative* experiments align the crevasse at an angle of $\sim 60°$ to the flow. Inside the crevasse the mesh is fully unstructured with an isotopic grid resolution of 5 m. This matches the study by Jordan et al. [2014]. Outside the crevasse the horizontal grid resolution varies from 250 m at the far-field boundaries refining to 20 m below the crevasse. The vertical grid resolution is 5 m matching the isotropic grid size within the crevasse, however, the tetrahedrons are aligned in columns to give a structured mesh in the vertical. This mesh choice was motivated by an instability linked to the melt dynamics that occurred when using a fully unstructured mesh with varying resolution outside the crevasse. Although, the domain extent of 5 km is evidently small compared with an entire ice shelf the conceptual idea is that this experiment is like a 'flume' experiment undertaken in a laboratory, perhaps imagining the crevasse rotating on a turntable within the 'flume'. Note that in reality the mesh for *across* and *along* flow experiments was the same and the forcing direction was rotated by 90°, whereas the meshes for the angled crevasses were different. In the

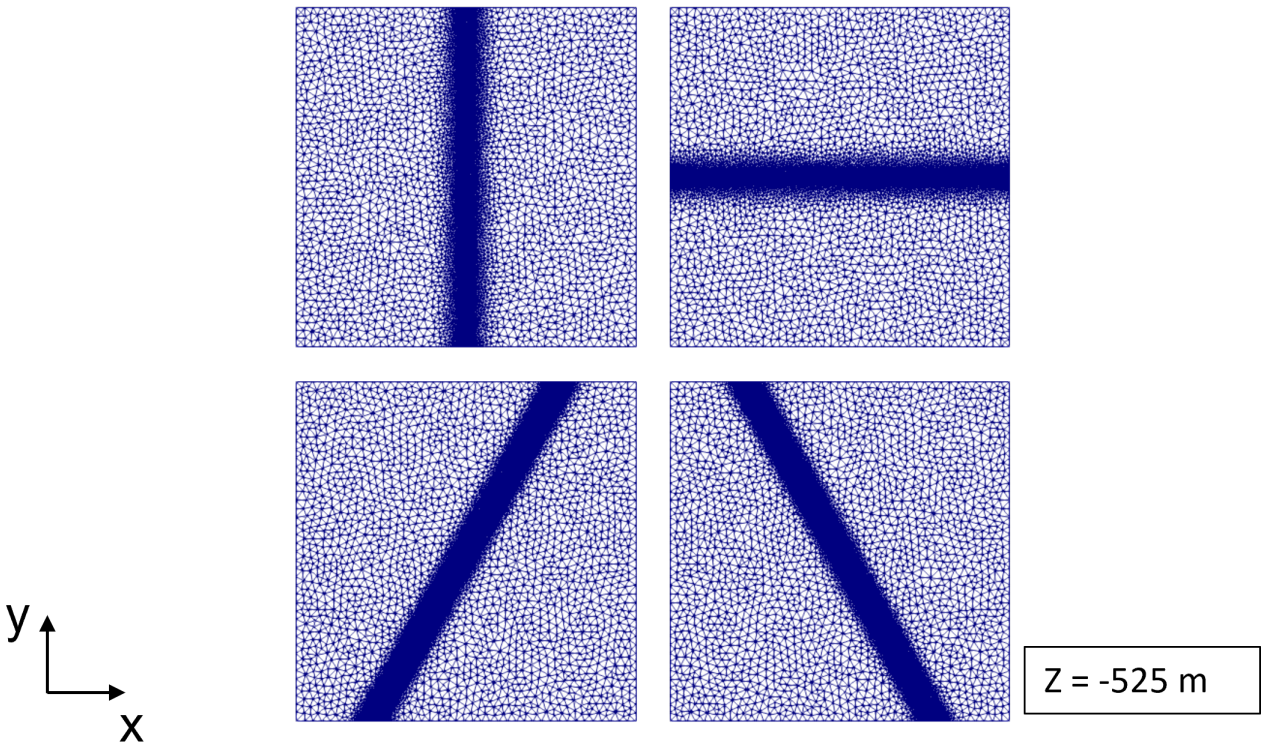


Figure 5.5: Plan view of meshes used in the 3D crevasse geometry experiment at a depth slice of 525 m. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle. The flow is predominantly in the negative x direction.

plots that follow the *along* flow experiment has been rotated to make visualisation easier. For each depth slice the figures are arranged so that the top left panel is the *across* flow experiment, the top right panel is the *along* flow experiment, the bottom left panel is the *positive* angle crevasse and the bottom right figure is the *negative* angle case. Figure 5.5 shows the plan view of the meshes taking a depth slice at 525 m. Figures 5.6 and 5.7 show the vertical cross sections of the meshes along y = 2500 m and x = 2500 m respectively.

The frazil ice model is turned off for these experiments. Although this is a simplification, by design frazil effects will not take place due to the restoring conditions being warmer than the depth dependent freezing point. These experiments are motivated by the crevasses found beneath Thwaites glacier and observed by Icefin [Schmidt et al., 2023]. On each side wall the salinity and temperature fields are given by linear profiles based on the Icefin observations. Temperature varies from -0.41 °C at 500 m depth to -0.1 °C at 600 m depth. Salinity varies from 34.37 at 500 m depth to 34.5 at 600 m depth. Robin boundary conditions for temperature and

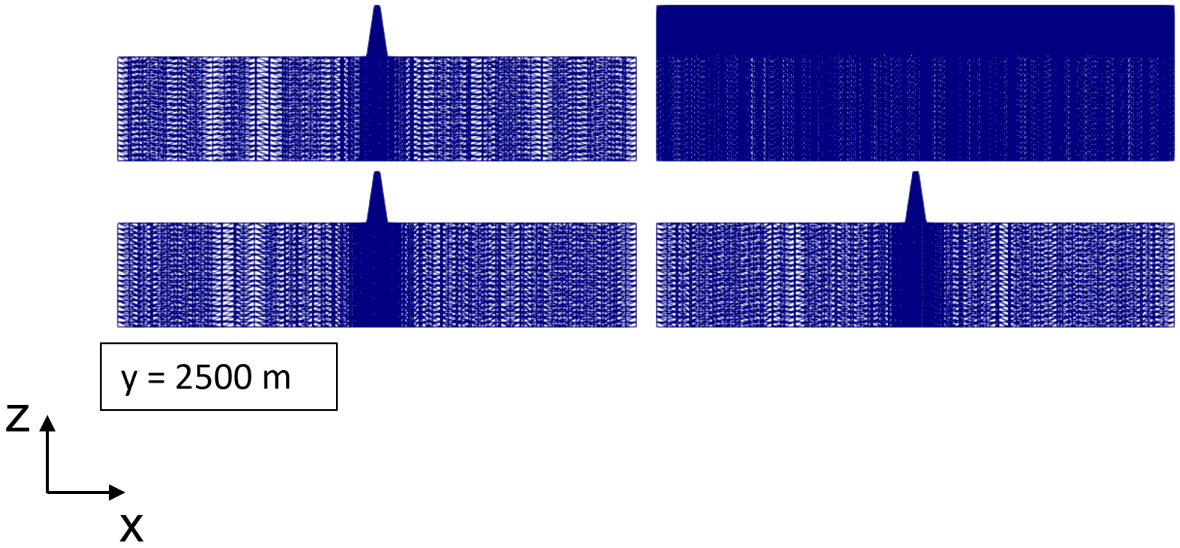


Figure 5.6: Cross section of meshes used in the 3D crevasse geometry experiment along y = 2500 m. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle. The flow is predominantly in the negative x direction. The plots have been exaggerated 10 times in the vertical direction.

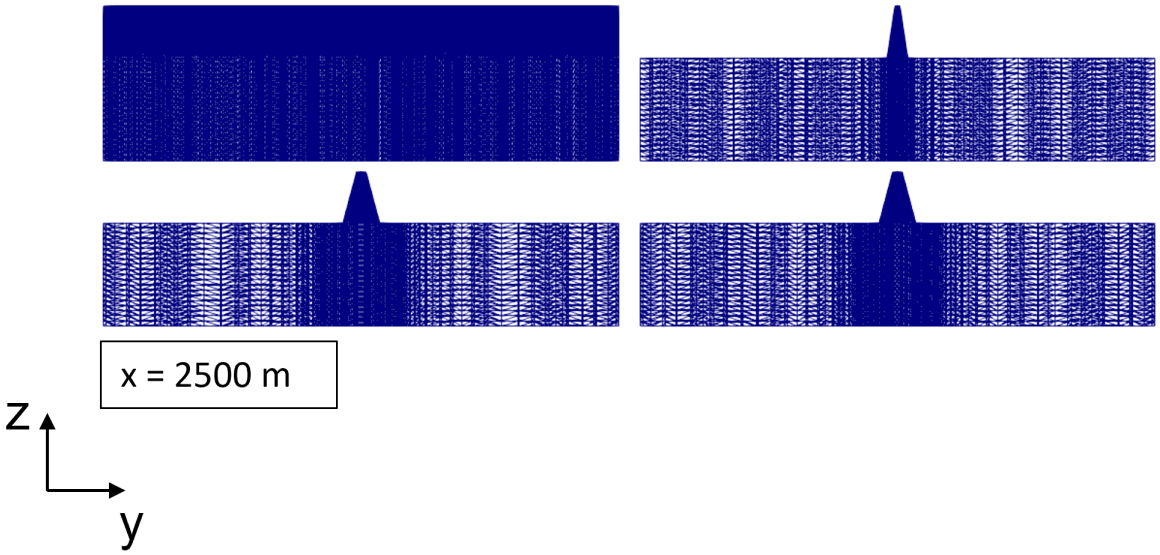


Figure 5.7: Cross section of meshes used in the 3D crevasse geometry experiment along x = 2500 m. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle. The flow is predominantly in the negative x direction. The plots have been exaggerated 10 times in the vertical direction. Note the change in crevasse thickness for the angled crevasses compared with the *along* crevasse is due to the cross section cutting the crevasse at an oblique angle.

salinity are applied using the velocity dependent melt parameterisation discussed in Section 2.6. The formulation from Holland and Jenkins [1999] is used for determining γ_T. The diffusive heat flux into the ice is assumed to be zero. Homogenous Neumann boundary conditions are applied for tracers at the bottom of the domain. A consequence of this choice is that boundary layers arise on open boundaries from having linear tracer profiles on the side walls in contrast to the bottom boundary enforcing no gradients in the vertical direction. The top and bottom of the domain are set to no normal flow boundaries, assuming a rigid lid for the top surface. For the *positive* and *negative* angle crevasses all side boundaries are open. Dirichlet boundary conditions are applied for pressure by integrating the density differences in the Dirichlet temperature and salinity boundary conditions to the (vertically uniform) reference state to give a hydrostatic pressure profile. For the *across* and *along* experiments the inflow and outflow boundaries (x = 0 m and 5000 m) are open, but the other two side boundaries are closed, no normal flow boundaries. Quadratic drag boundary conditions are applied on the top ($C_d = 9.7 \times 10^{-3}$) and bottom ($C_d = 2.5 \times 10^{-3}$) surface. Side boundaries are free slip. All boundary conditions are implemented weakly. The diffusivity and viscosity are set to 1×10^{-3} m^2 / s inside the crevasse. This is the same values as the Jordan et al. [2014] study as well as the same grid resolution. Outside the cavity the horizontal viscosity and diffusivity are linearly increased towards the boundary based on the grid cell size. This is evidently the simplest turbulence closure and for future work it would be interesting to investigate the effects of different turbulence models on these results.

Figure 5.8 shows a snapshot of velocity magnitude at 550 m depth after 10 days. Generally the flow is from right to left (negative x-direction) with a magnitude of 0.01 m/s, which matches the body force acting in the positive y-direction. On the y = 0 m boundary there is a thin boundary layer with slower flow and on the y = 5000 m boundary a significant region, extending to roughly y = 3500 m, with slower flow. Note the top two panels (*across* and *along* flow experiments) are very similar in this region primarily because they both have closed boundaries in the y-direction, whereas the bottom two panels (*positive* and *negative* angle experiments) are also similar due to having open boundaries in the positive y-direction. These boundary effects are primarily caused by having an imbalance between the pressure forcing on the open boundary and the

density stratification within the domain after the salinity (and temperature) have adjusted to the bottom boundary conditions. The increasing size of the arrows representing the velocity fields shows that the flow speeds are increased by the crevasse.

The presence of the crevasse becomes much more noticeable in Figure 5.9, a snapshot of velocity at 510 m, 10 m below the ice-ocean interface. In the top left the *across* flow experiment, equivalent to the 2D experiments of Jordan et al. [2014] shows the velocity decreasing slightly on the leading edge of the crevasse, but increasing after. The angle of the flow is largely unaffected by the presence of the crevasse. In the top right panel, the *along* flow experiment shows that there is an acceleration on the 'left hand' (y = 2400 m) side of the crevasse looking in the along flow direction to roughly 0.02 m/s, twice the background flow. Underneath the crevasse the flow is noticeably slower than the ambient flow. In the bottom left experiment, the *positive* angle experiment has clear acceleration along the crevasse. On reaching the crevasse the flow is deflected to the left, appearing to align with the orientation of the crevasse. In the bottom right panel, the *negative* angle crevase is clearly marked by a region of slow flow similar to the *across* experiment. The flow is deflected to the left when it reaches the crevasse, effectively hitting the crevasse at a right angle due to the orientation of the crevasse.

The snapshot of velocity at 502.5 m at 10 days is shown in Figure 5.10. This depth slice is within the final grid cell next to the ice-ocean interface. Although the patterns are very similar to the 510 m slice in Figure 5.9, differences in flow speeds are more pronounced. The upstream sides of the *across*, *positive* and *negative* crevasses have a clearly defined leading edge with a velocity faster than 0.02 m/s, but slower flow once underneath the crevasse. In the *positive* angle crevasse this fast flow along the leading edge is nearly continuous along the entire length of the crevasse. For the *across* and *negative* the fast flow is concentrated where y > 2500 m. In the *along* flow experiment the flow is nearly stationary for y > 2500 m. Beyond this, the velocity increases, though not as fast as the flow outside the 'left hand' wall of the crevasse. Also a noticeable feature is that the flow has broadly rotated clockwise by 45 °, which is an Ekman boundary layer. As u slows down close to the boundary the (depth independent) pressure gradient must be balanced by a v-component flow in the positive y-direction. An experiment was not run with free slip top boundaries, so it is hard to comment

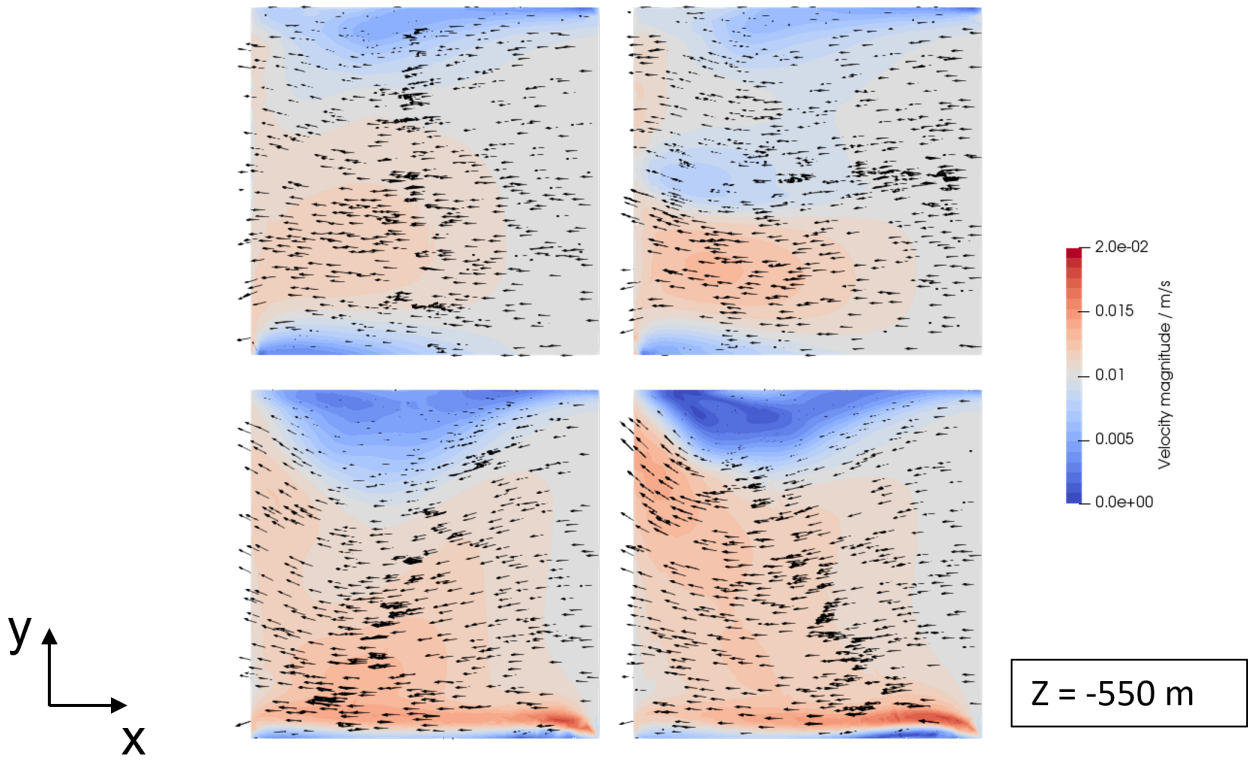


Figure 5.8: Depth slice of velocity magnitude at 550 m for the 3D crevasse geometry experiment after 10 days. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

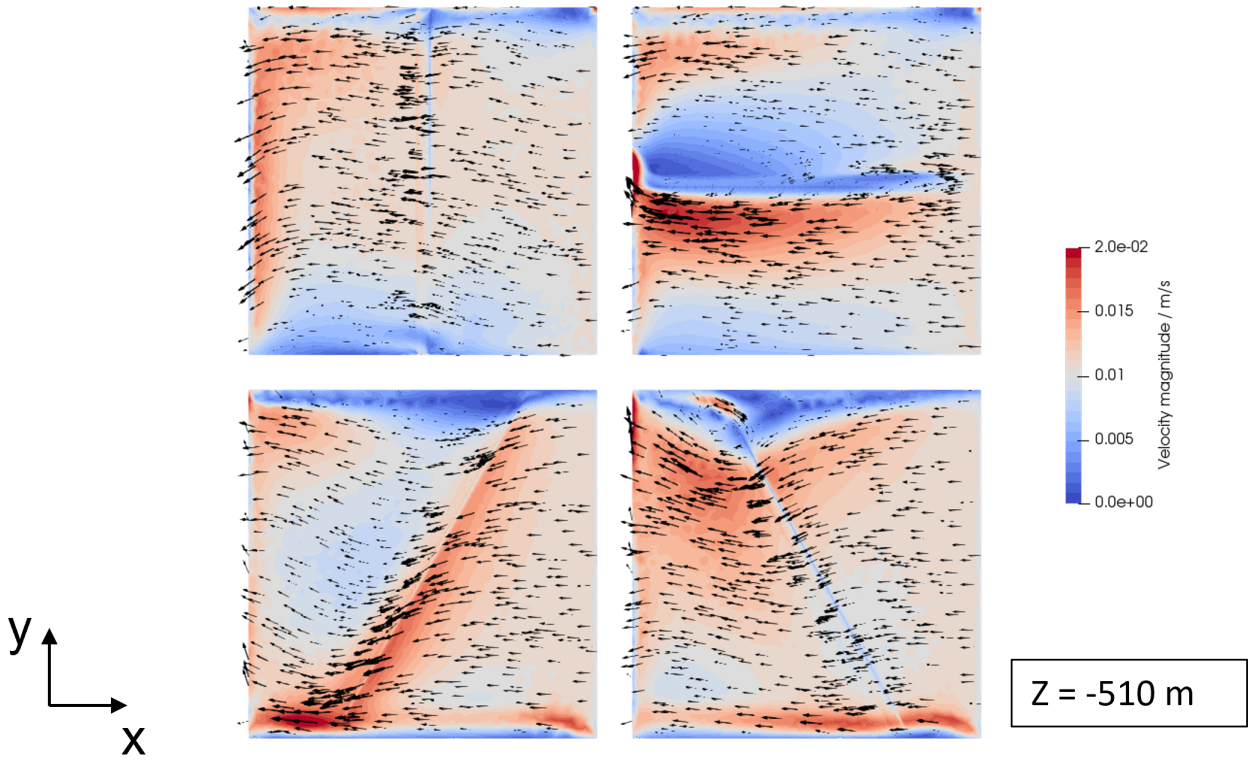


Figure 5.9: Depth slice of velocity magnitude at 510 m for the 3D crevasse geometry experiment after 10 days. This is 10 m (2 grid cells) below the crevasse. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

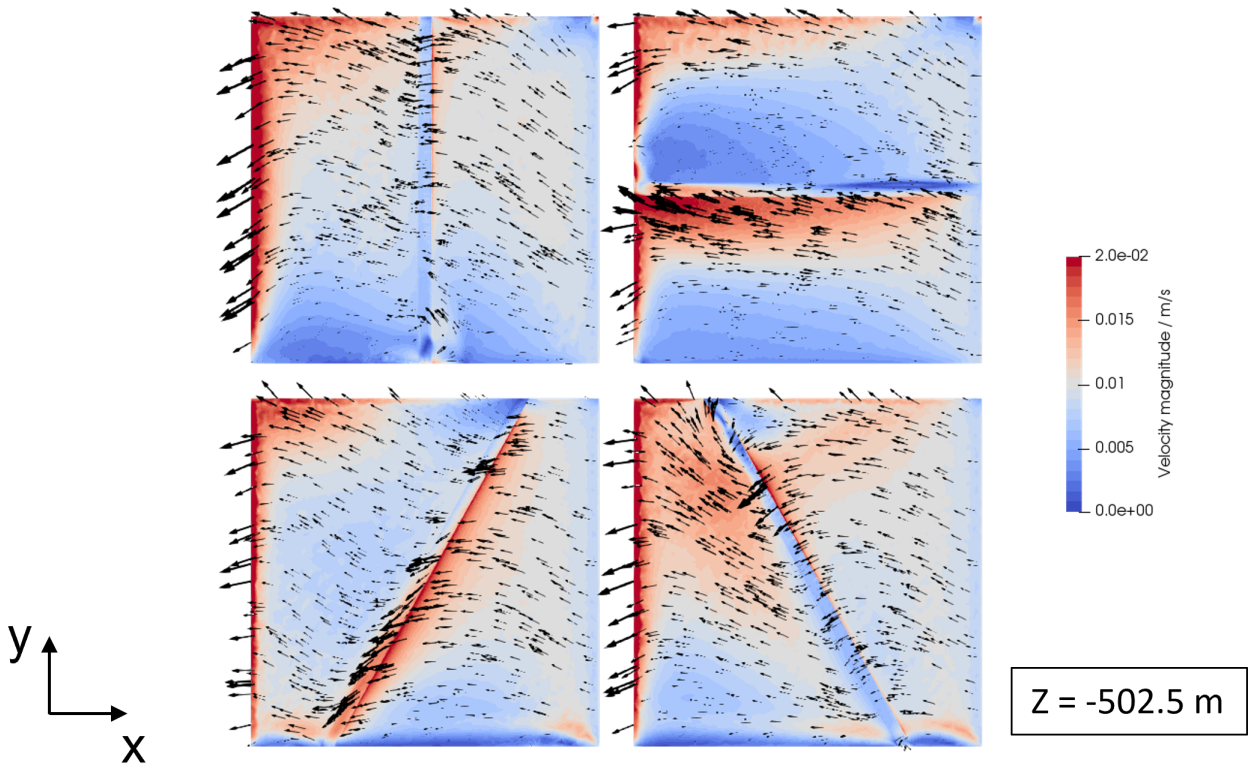


Figure 5.10: Depth slice of velocity magnitude at 502.5 m for the 3D crevasse geometry experiment after 10 days. This is within the last grid cell beneath the crevasse ($\Delta z = 5$ m). Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

on how important this rotation in the flow direction is close to the boundary. Note that drag boundary was applied using a quadratic drag stress formulation, so the tangential flow is not zero at the boundary. Considering Ekman layer dynamics, with 5 m vertical resolution and an Ekman depth, $d_E = \sqrt{2\kappa_v/|f|} = \sqrt{2 \times 1 \times 10^{-3}/1.4 \times 10^{-4}} \sim 4$ m the model is capable of capturing a crude Ekman layer thanks to linear approximations within elements.

Figure 5.11 shows the velocity at 497.5 m depth after 10 days, a depth slice through the cell immediately within the crevasse. Note that for visualisation purposes the orientation of the *negative*, *across* and *positive* angle crevasses have been rotated to the same orientation as the *along* flow crevasse. Arrows have been added to indicate the direction of flow. For the *along* flow experiment (top) and *positive* (bottom) the flow within the crevasse is very similar to the flow just underneath the crevasse. Namely, the flow accelerates along the crevasse in the direction of the flow. For the *negative* angle crevasse there is a clear region of faster flow on the very edge of the crevasse 'lip' coinciding with the concentrated flow where y $>$ 2500 m.

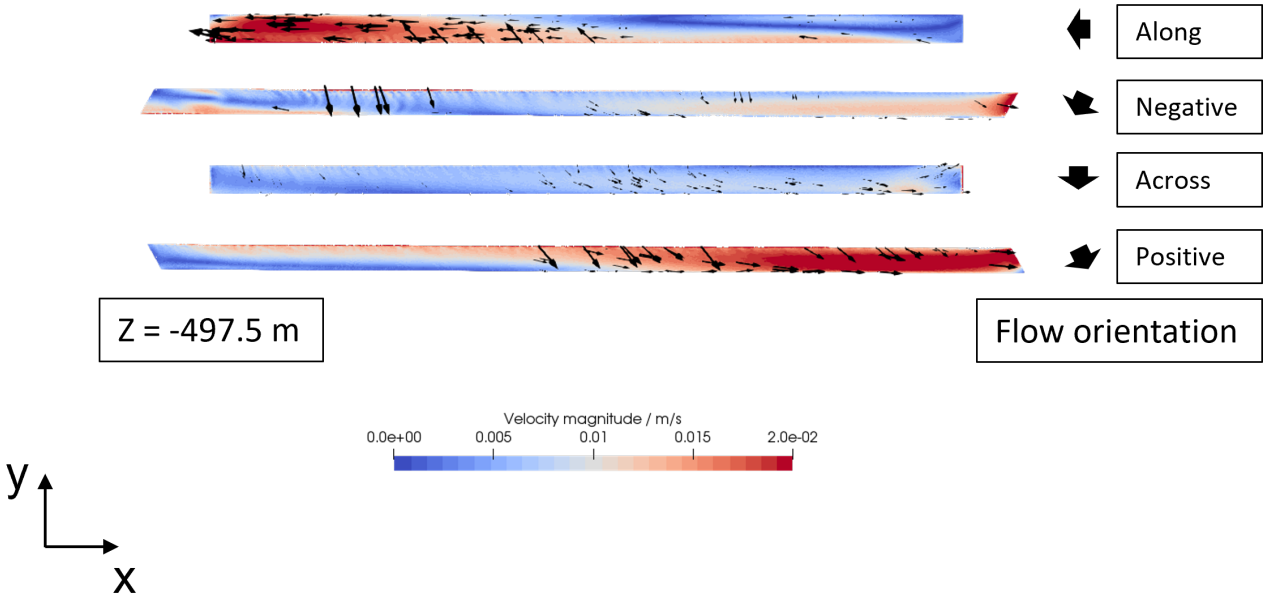


Figure 5.11: Depth slice of velocity magnitude at 497.5 m for the 3D crevasse geometry experiment after 10 days. This is within the first grid cell inside the crevasse ($\Delta z = 5$ m). The crevasses have been rotated to make visualisation easier. On the right black arrows indicate the direction of flow. From top to bottom the depth slices correspond to *along* (not rotated), *negative* angle (rotated 60 ° anticlockwise), *across* (rotated 90 ° anticlockwise) and *positive* angle (rotated 120 ° anticlockwise).

For y $<$ 2500 m, the flow is slowly increasing in the negative y-direction. Velocities are slowest within the *across* case, but there is a sign that the flow is deflected to the left within, and by the presence of, the crevasse.

Figure 5.12 shows the 10 day velocity at 475 m (mid crevasse depth). At this depth the influence of the barotropic forcing is less clear. The *across* flow experiment has noticeably lower velocity than the other experiments. Importantly, this was the only experiment to have closed boundaries preventing fluid flowing out of the crevasse sides. The flow within the *along*, *negative* and *positive* angle crevasses is qualitatively very similar, with flow out both ends of the crevasse, fastest for the *positive* and *negative* angle cases. It is likely that this flow pattern is caused by a combination of not having pressure boundary conditions which directly matches the changing stratification in the crevasse due to melting and by restricting the barotropic force to below the depth of the crevasse.

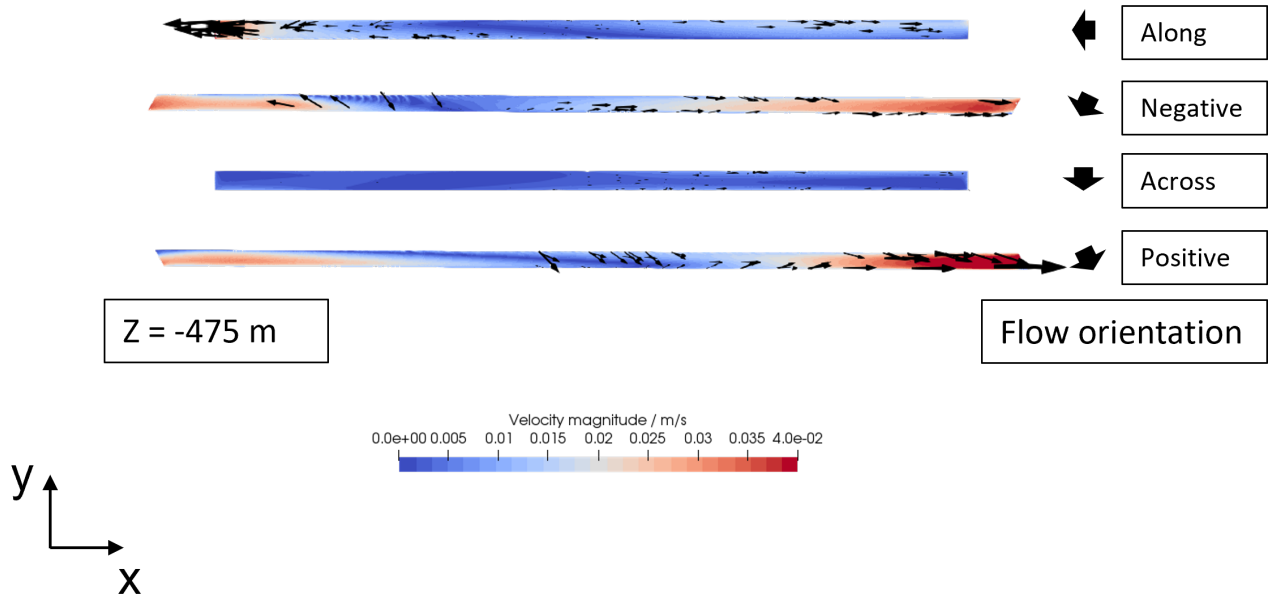


Figure 5.12: Depth slice of velocity magnitude at 475 m for the 3D crevasse geometry experiment halfway inside the crevasse after 10 days. The crevasses have been rotated to make visualisation easier as in Figure 5.11. Note the colour scale has increased from the previous plots to a maximum magnitude of 0.04 m/s.



Figure 5.13: Depth slice of vertical velocity at 550 m for the 3D crevasse geometry experiment after 10 days. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

The spatial pattern of the vertical velocity is very similar to the velocity magnitude. At 550 m depth the vertical velocity is almost negligible which is to be expected given the barotropic forcing as shown by Figure 5.13. Similar to the velocity magnitude plots, Figure 5.14 shows the vertical velocity signal becomes noticeable for the crevasse region at about 510 m depth, with a clear positive vertical velocity on the upstream, leading edge of the crevasse for the *across*, *positive* and *negative* angle crevasse cases ('left hand' wall for *along* flow experiment), with a magnitude of $\sim 1 \times 10^{-3}$ m/s at 502.5 m depth (Figure 5.15) that continues into the crevasse at 497.5 m depth (Figure 5.16). Since the mesh is isotropic within the crevasse with a grid size of 5 m, the magnitude of the vertical acceleration term across one cell is only an order of magnitude lower than the horizontal acceleration due to advection, given a characteristic horizontal velocity of 0.01 m/s. Although the nonhydrostatic vertical acceleration terms are not the leading order, their contribution is significantly greater than for typical ocean model configurations with highly anisotropic aspect ratios often on the order of 10^4.

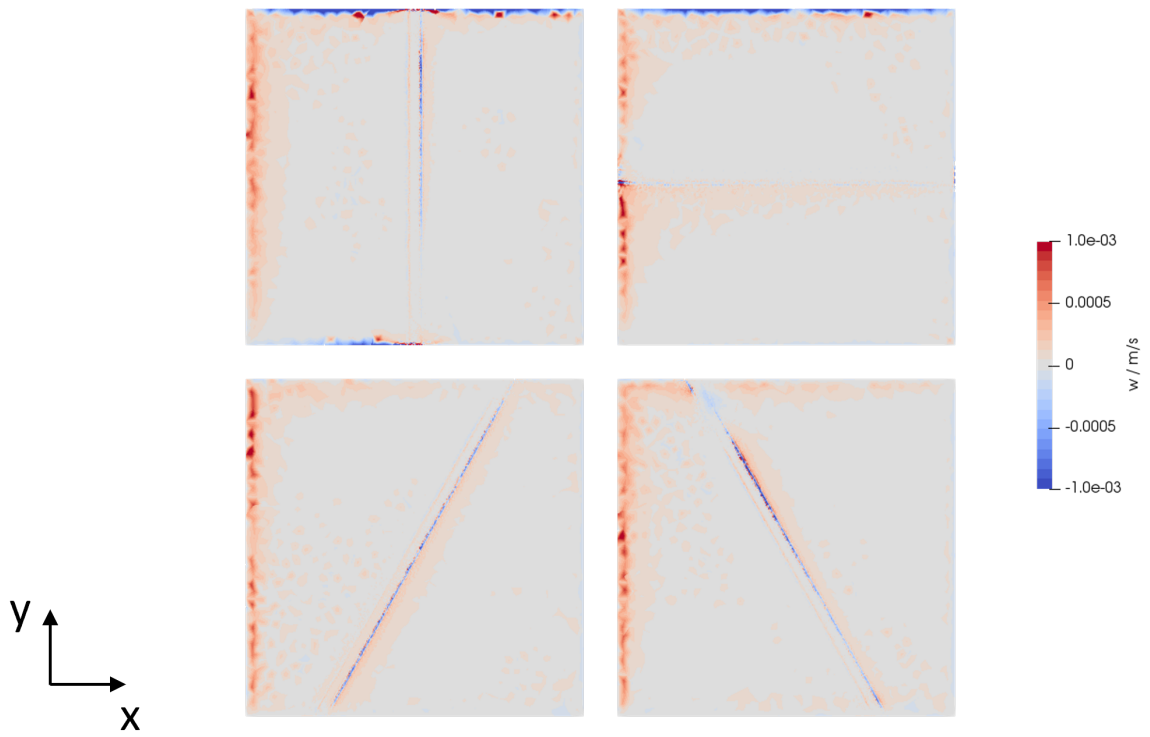


Figure 5.14: Depth slice of vertical velocity at 510 m for the 3D crevasse geometry experiment after 10 days. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

Figure 5.17 shows a plan view snapshot of melt rate after ten days. The melt rate broadly follows the velocity patterns; where the velocity is high (i.e on the upstream and left sides of the crevasse) there is higher melt. Along this leading edge the melt rates approach 10 m/a, about double the background melt rate outside the crevasse. The lowest melt appears to be on the *across* flow experiment. Similarly to the velocity in Figure 5.10, the melt rate along the upstream edge of the *positive* angle case is nearly continuous for the length of the crevasse with a value of about 8 m/a. This melt rate is also achieved for the *negative* angle crevasse, but only for a short section coinciding where the flow concentrates as shown in Figure 5.10 (y $>$ 3000 m). Along the 'left hand' side of the *along* flow crevasse (looking along the flow) the melt rate is higher than the other wall, reaching similar magnitude of melt rates to the *positive* angle experiment. There is noticeably a different background melt rate outside of the crevasse with generally higher melt rates of $\sim$ 5 m/a for y $<$ 2400 m, whereas for y $>$ 2600 m there is a patch of low melt rate ($\sim$ 1 m/a), that lines up with a region of slower flow ($<$ 0.005 m/s). The top face of each crevasse is clearly visible as a dark blue region, indicating that there is

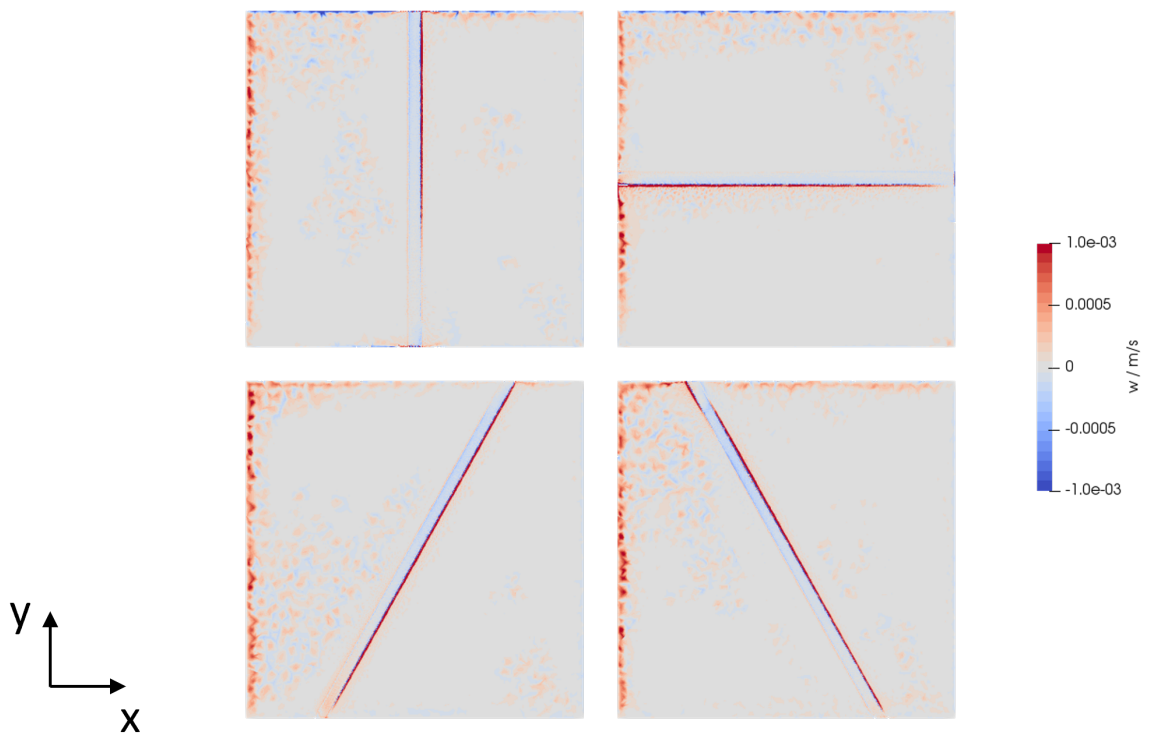


Figure 5.15: Depth slice of vertical velocity at 502.5 m for the 3D crevasse geometry experiment after 10 days. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

lower melt ($\sim$ 1 m/a) compared to the sloping side walls (up to 6 m/a).

Transects of melt rate along y = 2500 m (Figures 5.18 and 5.19) and x = 2500 m (Figures 5.20) and 5.21 confirm the above outline. Figure 5.19 clearly shows that the highest melts are on the upstream edge of the crevasse. This occurs directly at the point of initial change in slope (y = 2600 m) and is shown in Figure 5.19, with a zoomed in perspective of the crevasse region, along with guide lines for marking the shape of the crevasse (trapezoid in 2D vertical cross section). The melt rate at the top of the *along* flow crevasse is clearly lower (0.5 – 2 m/a) than the background value for the other crevasses ($\sim$ 3 m/a). It is interesting that for both the *across* and *negative* experiments the melt rate within the crevasse is nearly symmetrical, notwithstanding the peak in melt rates on the upstream edge. There is a peak that occurs on the sloping side wall of the crevasse, followed by a minimum. This is interesting as it ties in with the observations from Schmidt et al. [2023] that found higher melt on the vertical side walls of the crevasse. The magnitude of melting at the top of the crevasse then climbs for the *across*

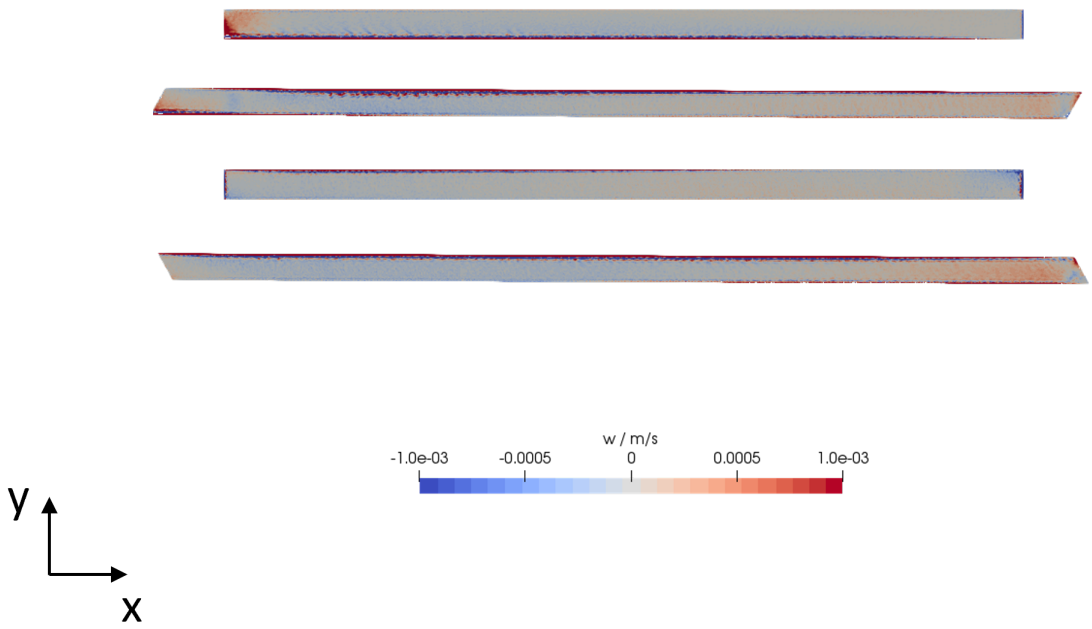


Figure 5.16: Depth slice of vertical velocity at 497.5 m for the 3D crevasse geometry experiment after 10 days. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

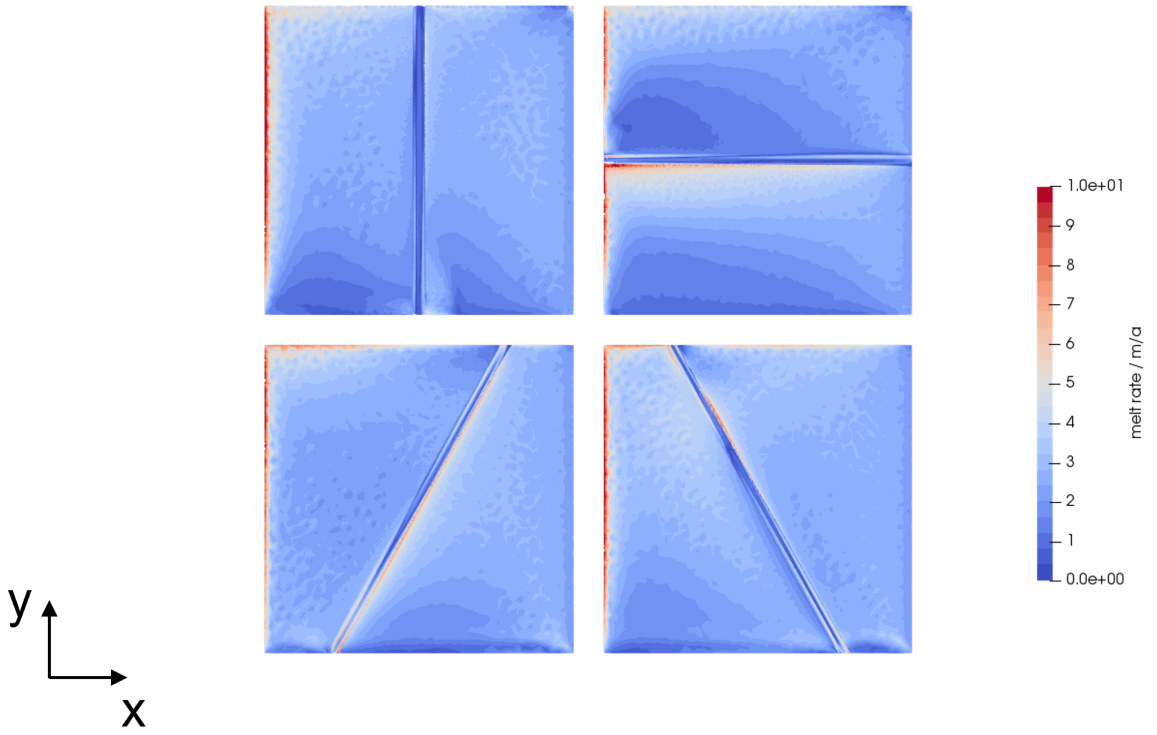


Figure 5.17: Plan view of melt rates for the 3D crevasse geometry experiment after 10 days. Top left: *across*, top right: *along*, bottom right: *negative* angle, bottom left: *positive* angle.

and *negative* both plateauing at $\sim$ 1 m/a, before repeating a similar pattern on opposite side wall. It seems likely that this melt pattern is a result of a flow independent of the barotropic current below the crevasse.

The *positive* angle experiment has a noticeably different melt structure that is more asymmetrical. There is a much less obvious decrease in melt rate at y = 2400 m compared with the *across* and *negative* cases. There is a similar, albeit offset, falling and rising melt rate pattern observed on the downstream wall of the crevasse, but along the top of the crevasse there is a quasi-linear drop in melt rate, not seen for the other crevasse orientations. On the upstream wall of the crevasse the melt rate persistently rises, with roughly double the average melt rate compared with the other cases. This seems consistent with the modelled velocity fields, in that for the *positive* angle experiment the velocity is clearly deflected to the left by the crevasse, allowing the flow to accelerate. This in turns leads to higher melt rates on the upstream wall.

The transect of melt rates through x = 2500m is shown in Figure 5.20. Note due to the

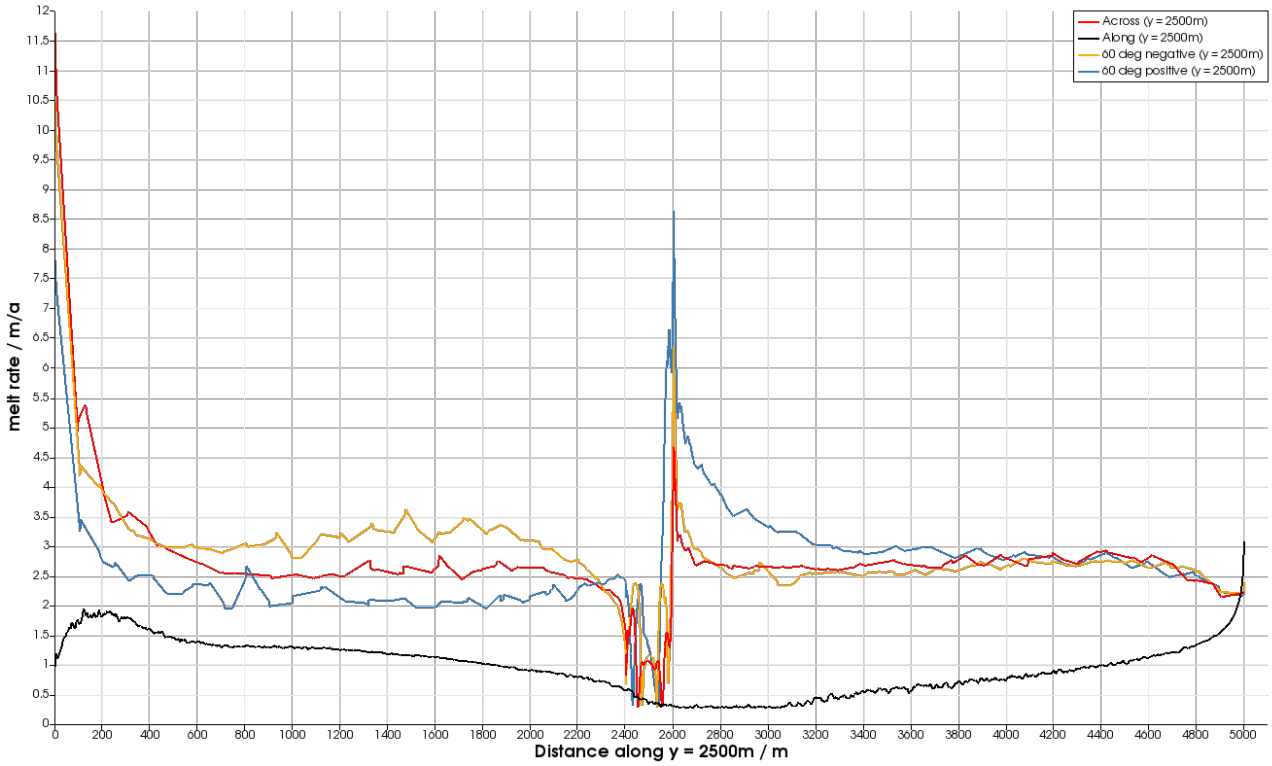


Figure 5.18: Transect of melt rates for the 3D crevasse geometry experiment along y = 2500 m after 10 days. red: *across*, black: *along*, yellow: *negative* angle, blue: *positive* angle.

orientation of the crevasse the horizontal extent of the *positive* and *negative* angled crevasses is wider than the *along* flow crevasse. Also, the upstream side of the *positive* angle crevasse is now on the left side of the figure corresponding to x < 2400 m. Again, the transect highlights that melt rate is highest on the upstream and 'left hand' side for the *along* flow experiment. The melt rate for the *across* flow experiment is fairly uniform along the top of the crevasse about 1 m/a, which, although is lower than the background value outside the crevasse, is slightly higher than the equivalent transect for the *along* flow experiment shown in Figure 5.18. The zoomed in melt transects for *negative* and *positive* experiments from Figure 5.21 have very similar pattern to Figure 5.19 before. Interestingly, the overall shape of the *along* flow experiment is similar to the *positive* angle case with an asymmetrical melting pattern across the top of the crevasse and generally higher melting on the 'left side' (x < 2500 m). This ties in with the idea that the *positive* angle crevasse orientation is allowing the flow within the crevasse to be more like the *along* flow case. The peak of 7 m/a is not exactly on the location of change in slope (y = 2400 m) but slightly offset towards the centre of the crevasse.

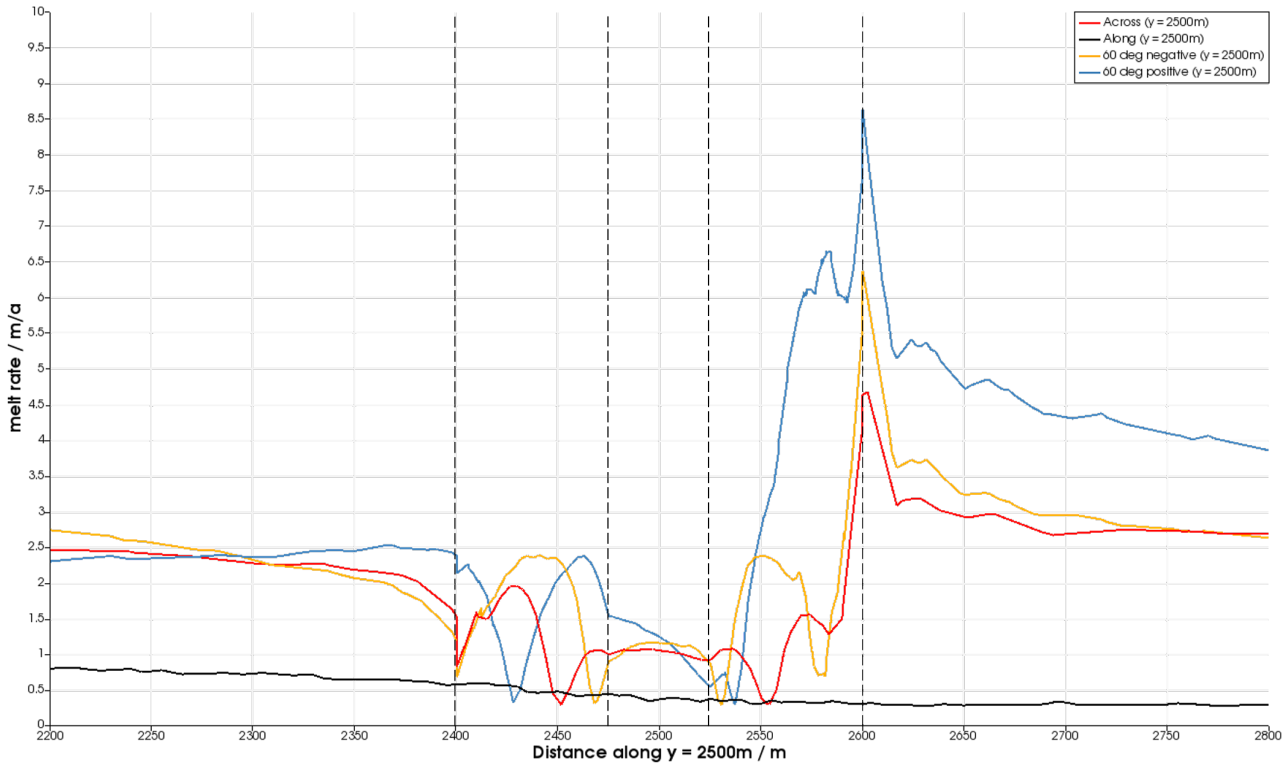


Figure 5.19: Transect of melt rates for the 3D crevasse geometry experiment along y = 2500 m after 10 days zoomed to the crevasse region. red: *across*, black: *along*, yellow: *negative* angle, blue: *positive* angle. Note the vertical dashed lines correspond to the corners of the *across* flow crevasse crevasse that is 200 m wide at the base and 50 m wide at the top with linearly sloping side walls. See Figure 5.6 for the cross section of the crevasse shape along y = 2500 m.

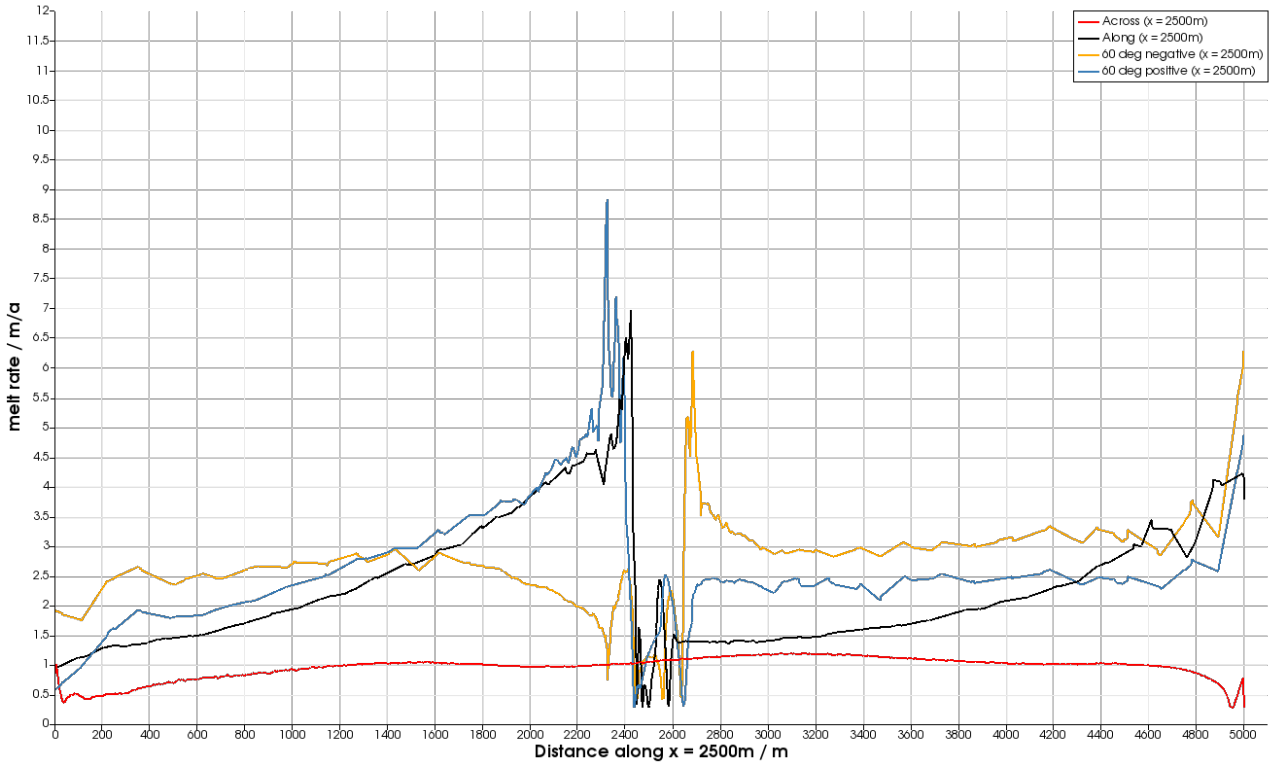


Figure 5.20: Transect of melt rates for the 3D crevasse geometry experiment along x = 2500 m after 10 days. red: *across*, black: *along*, yellow: *negative* angle, blue: *positive* angle.

Although these results are likely to be grid resolution dependent based on Chapter 3, the features do not appear to be grid artefacts. With a grid size of 5 m within the crevasse, there is always a minimum of ten cells spanning the crevasse height of 50 m and the minimum crevasse width at the top of 50 m. An obvious extension to this work would be to test the sensitivity of the results to different mesh resolutions. Note that the current setup was limited by memory constraints running on a single node of the HPC cluster at Imperial College (with 128 cores and 124GB of RAM). To fix this some further work is needed to make the Singularity recipe for running Firedrake work across multiple nodes.

Overall, the 3D experiments show that the crevasse orientation can have an effect on melt rates with nearly a 70 % increase in the melt rate on the upstream edge of the crevasse between the *positive* angle case and the *across* flow experiment. There is significant asymmetry in the melt patterns for the *along* and *positive* experiments where a flow is able to develop along the crevasse. It would be interesting to investigate if these asymmetrical patterns could generate the basal terrace topography, with higher melting on the upstream side could turn a crevasse

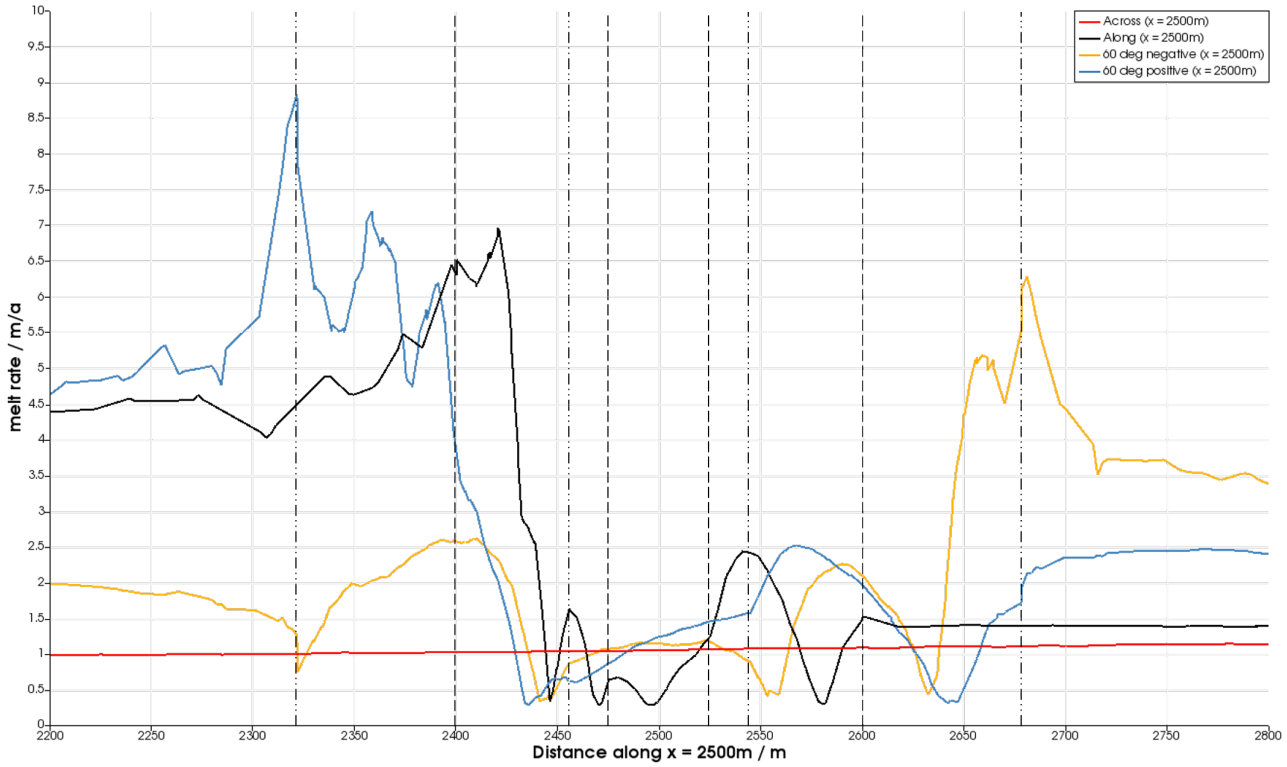


Figure 5.21: Transect of melt rates for the 3D crevasse geometry experiment along x = 2500 m after 10 days zoomed to the crevasse region. red: *across*, black: *along*, yellow: *negative* angle, blue: *positive* angle. Note the single vertical dashed lines correspond to the corners of the *along* flow crevasse that is 200 m wide at the base and 50 m wide at the top with linearly sloping side walls. The vertical lines with two dots and a single dash correspond to the corners of the *negative* and *positive* that are wider along this transect owing to cutting the crevasse at an oblique angle. See Figure 5.7 for the cross section of the crevasse shape along x = 2500 m.

into a basal terrace. We also find that the melt rate is higher on the sloping side walls than the top of the crevasse. However, with regards to the original hypothesis, broadly the story is similar to the 2D results of Jordan et al. [2014]; the effects of Coriolis in these experiment runs does not drastically change the fact that there is generally lower melting within the crevasse than the surrounding ice shelf. Thus, the ocean is primarily acting to 'erode' crevasses, not accentuate them.

A limitation of this study is the simplicity of the turbulence closure scheme used here, i.e., constant eddy viscosity and diffusivity. An obvious extension would be to use a Reynolds Averaged Navier Stokes (RANS) model, similar to Yeager [2018], to better capture the mixing within the crevasse. The 2D case with no melting is similar to a backward facing step, which is a common test case for turbulence models in the CFD literature[Kuzmin et al., 2007]. Note that during the moving mesh simulations of the crevasse domain in Yeager [2018] (the same initial domain as in Jordan et al. [2014]) the $k - \epsilon$ model is switched off.

5.5 Code Availability

The code is available at https://github.com/thwaitesproject/thwaites.git.

Run scripts for the 2D crevasse comparison with Jordan et al. [2014] including frazil ice dynamics (Section 5.3):

- thwaites/run_scripts_testing/ice_shelf_cavity/crevasse.py

Run scripts for investigation into rotational and geometry effects for 3D basal crevasses (Section 5.4):

- thwaites/run_scripts_testing/ice_shelf_cavity/10km_flume_crevasse_3d.py

5.6 Conclusion

This chapter investigated flow within ice shelf basal crevasses. We showed that our model compares favourably with Fluidity for a published 2D idealised crevasse simulation, including frazil ice effects, clearly demonstrating our capability to simulate dynamic, nonhydrostatic flows on an unstructured mesh robustly. The limitation of the Jordan et al. [2014] study, however, was that the geometry assumed that the flow hit the crevasse at right angles. We extended this earlier work to 3D to consider crevasse orientation and the effects of Coriolis on the flow. We found that melting was consistently higher on the upstream limb of the crevasse, with an asymmetric melting pattern when a component of the flow is oriented in the same direction as the crevasse. This warrants further investigation, especially if it might be linked to a formation mechanism for basal terraces. Within the crevasse, however, melt rates remained low, due to stratification build up, implying that the ocean is still acting to smooth the ice base and hence erode basal crevasses as implied by the 2D results. Future work will consider the important topic of how best to resolve and represent turbulent processes under ice shelves. This is an exciting area of open research and hopefully the modelling framework developed and presented here will provide a useful tool for future researchers to explore these questions.

Chapter 6

Conclusion

We have presented a new model highly applicable for simulating ocean flow in complex ice shelf cavity environments, owing to our flexible discretisation approach. Alongside the established P1DG-P2 ocean modelling element pair, we used a new velocity-pressure tensor product finite element pair for high aspect ratio ocean applications, that inherits the advantages of the DG framework. We also showed how slope limiters can have a non-negligble impact on melt rates, especially when applied to the velocity components, so care needs to be taken when using them for ice-ocean applications. Limiters proved crucial, however, in eliminating a spurious freezing signal at the ice front region.

We developed a novel Method of Manufactured Solutions (MMS) test that includes melting, thus rigorously verifying that the spatial discretisation of the velocity, pressure and tracer fields, and consequently the melt rate, is second order accurate as expected. We also compared our model with analytical results for an Ekman spiral, confirming the accurate implementation of rotation for both the 3D and 2.5D formulations. We believe that it is important to carry out rigorous code verification checks, for which the MMS technique is ideally suited, because comparing solely with other models may mask model errors. This is especially true considering the strong dependence of the melt parameterisation on grid resolution, as seen in the simulations of the ISOMIP+ test case.

Our comparison with MITgcm highlighted the accuracy gains from using a higher order DG

finite element discretisation over low order finite volume methods. The progression from simple to more complex domains instilled confidence in the model. We showed good agreement for the 2D (and 2.5D) idealised grounding zone domain where there is an obvious overturning 'ice pump' signal. We also compared Firedrake with MITgcm for an extended 'ISOMIP-like' 2D domain, this time using the tensor product discretisation developed for large scale domains and again, the agreement was very close between the two models. We also made a direct comparison of z (constant vertical resolution) and σ-coordinate (fixed number of layers) models using our model in 2D, owing to our flexible model discretisation and showed that the calculated melt rate between the two models was comparable, subject to the 'stripy' melt rate pattern caused by the z-coordinate grid.

The model is capable of running robustly in 3D domains of a full ice shelf cavity, demonstrated by simulations of the ISOMIP+ test case. Our simulation results compare favourably when run alongside MITgcm, and the melt rates are consistent with the spread found in the literature [Asay-Davis et al., 2016, Zhou and Hattermann, 2020, Gwyther et al., 2020]. We found melt rates are highly sensitive to grid resolution as has been observed before and showed that by refining MITgcm in the vertical we were able to achieve much better spatial qualitative agreement between the models. By comparing Firedrake simulations with a different number of layers and vertical diffusivities we showed that with the default ISOMIP+ parameters the magnitude of the spurious mixing in Firedrake was of a similar order to the specified diffusivity. With an order of magnitude higher explicit diffusivity we were able to achieve a converged melt rate, without tuning parameters within the melt parameterisation. Ultimately the balance between specified mixing and spurious numerical mixing dictates how sensitive the melt rate is to changes in the grid resolution (and discretisation choice). Our study emphasises the importance of running ice-ocean cavity simulations at multiple resolutions to gauge this sensitivity as it will always be problem specific.

As discussed in Section 3.6, given the assumption of fixed diffusivity and viscosity close to the boundary, finding a melt rate that is insensitive to grid resolution using the existing melt rate parameterisation may not be directly applicable to real ice shelf cavity environments. A detailed investigation into the melt parameterisation is beyond the scope of this work, and developing a

new, more robust melt parameterisation for large scale applications is not our immediate aim. Studies of the ice-ocean boundary layer are underway that use a hierarchy of models, from Direct Numerical Simulations [Middleton et al., 2021, Couston et al., 2021], to Large Eddy Simulations [Vreugdenhil and Taylor, 2019, Begeman et al., 2022, Vreugdenhil et al., 2022], as well as theoretical models [Jenkins, 2016, 2021]. It is likely that these combined approaches will be needed to develop a parameterisation that is consistent with the physics explicitly resolved by the numerical model (and the model's own mixing schemes) and is applicable to the unique environment of the ice shelf-ocean boundary layer where the effects of stratification and basal slope are important. Potentially a model like the one implemented here using Firedrake with flexible grids in the horizontal and vertical will be a useful tool to bridge the gap between high resolution models of boundary flow and the necessarily coarser regional and global scale ocean models.

Ultimately model validation will remain a challenge until more direct observations of sub-ice shelf ocean conditions are made in tandem with measurements of melt rate. Adjoint models are powerful tools to assess model uncertainties and incorporate observations into numerical ocean models [Errico, 1997, Heimbach and Losch, 2012, Goldberg et al., 2020, Nakayama et al., 2021b]. We showed preliminary adjoint sensitivity results for an idealised grounding zone domain, which clearly showed a 'reverse' ice-pump mechanism. The link between buoyancy and melt rate also suggests the importance of subglacial outflow into the cavity. The sensitivity of melt rate to vertical mixing and the turbulent exchange coefficients (especially at the grounding zone) was also identified by the adjoint model. This again suggests the importance of accurately modelling, or accounting for, mixing within ice shelf cavities. We plan to implement additional turbulence closure schemes such as RANS and LES to investigate these effects, with the caveat that the turbulence closure problem is inherently uncertain and thus challenging to represent faithfully. Using an objective function based on the difference in temperature and salinity between two synthetic forward simulations for a depth profile, we showed that there was a clear sensitivity signal to the boundary conditions at inflow regions at the open boundary even after sensitivity to temperature and salinity within the domain had faded. This raises the exciting prospect of being able to directly incorporate field observations into a simulation,

by performing an inversion for a range of unknown parameters based on CTD profiles. This would enable accurate simulations of stratification near the ice ocean interface (especially in the vicinity of grounding zones) and thus hopefully improve understanding of how best to represent the boundary layer in coarse scale climate models aimed at tackling the challenging problem of making accurate long term sea level predictions.

In the final chapter we applied the model to investigate basal crevasses. We performed an initial comparison with a published Fluidity 2D experiment of a ice shelf basal crevasse including frazil ice dynamics [Jordan et al., 2014]. We extended this work to consider the effect of geometry and flow rotation on basal crevasse melt rate. We found enhanced melt rate on the upstream side of the crevasse and an asymmetric melting signature where a component of the flow is aligned with the crevasse. From these experiments melting is still suppressed within the crevasse compared with background levels outside the crevasse, supporting the conclusions from the 2D experiments in that the ocean acts to smooth the ice base and erode basal crevasses. An obvious limitation of this study is the simplicity of the turbulence closure. How best to resolve and represent turbulent processes under ice shelves is an important open research question and hopefully the flexible modelling framework developed and presented here will provide a useful tool for future ice ocean researchers to explore these questions.

Bibliography

Alistair Adcroft, Chris Hill, and John Marshall. Representation of topography by shaved cells in a height coordinate ocean model. *Monthly Weather Review*, 125(9):2293–2315, 1997.

Karen E Alley, Ted A Scambos, Matthew R Siegfried, and Helen Amanda Fricker. Impacts of warm water on antarctic ice shelf stability through basal channel formation. *Nature Geoscience*, 9(4):290–293, 2016.

Karen E Alley, Christian T Wild, Adrian Luckman, Ted A Scambos, Martin Truffer, Erin C Pettit, Atsuhiro Muto, Bruce Wallin, Marin Klinger, Tyler Sutterley, et al. Two decades of dynamic change and progressive destabilization on the thwaites eastern ice shelf. *The Cryosphere*, 15(11):5187–5203, 2021.

Karen E Alley, Ted A Scambos, and Richard B Alley. The role of channelized basal melt in ice-shelf stability: recent progress and future priorities. *Annals of Glaciology*, pages 1–5, 2023.

Martin Alnæs, Jan Blechta, Johan Hake, August Johansson, Benjamin Kehlet, Anders Logg, Chris Richardson, Johannes Ring, Marie E. Rognes, and Garth N. Wells. The FEniCS Project Version 1.5. *Archive of Numerical Software*, 3(100), December 2015. ISSN 2197-8263. doi: 10.11588/ans.2015.100.20553. URL `https://journals.ub.uni-heidelberg.de/index.php/ans/article/view/20553`. Number: 100.

Martin S. Alnæs, Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. Unified form language: A domain-specific language for weak formulations of partial differ-

ential equations. *ACM Transactions on Mathematical Software*, 40(2):9:1–9:37, March 2014. ISSN 0098-3500. doi: 10.1145/2566630. URL https://doi.org/10.1145/2566630.

Robert J. Arthern and C. Rosie Williams. The sensitivity of West Antarctica to the submarine melting feedback. *Geophysical Research Letters*, 44(5):2352–2359, 2017. ISSN 1944-8007. doi: 10.1002/2017GL072514. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2017GL072514. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2017GL072514.

Xylar S. Asay-Davis, Stephen L. Cornford, Gaël Durand, Benjamin K. Galton-Fenzi, Rupert M. Gladstone, G. Hilmar Gudmundsson, Tore Hattermann, David M. Holland, Denise Holland, Paul R. Holland, Daniel F. Martin, Pierre Mathiot, Frank Pattyn, and Hélène Seroussi. Experimental design for three interrelated marine ice sheet and ocean model intercomparison projects: MISMIP v. 3 (MISMIP +), ISOMIP v. 2 (ISOMIP +) and MISOMIP v. 1 (MISOMIP1). *Geoscientific Model Development*, 9(7):2471–2497, July 2016. ISSN 1991-959X. doi: 10.5194/gmd-9-2471-2016. URL https://gmd.copernicus.org/articles/9/2471/2016/. Publisher: Copernicus GmbH.

Uri M. Ascher, Steven J. Ruuth, and Brian T. R. Wetton. Implicit-Explicit Methods for Time-Dependent Partial Differential Equations. *SIAM Journal on Numerical Analysis*, 32 (3):797–823, June 1995. ISSN 0036-1429. doi: 10.1137/0732037. URL https://epubs.siam.org/doi/10.1137/0732037. Publisher: Society for Industrial and Applied Mathematics.

Alexandros Avdis, Adam S. Candy, Jon Hill, Stephan C. Kramer, and Matthew D. Piggott. Efficient unstructured mesh generation for marine renewable energy applications. *Renewable Energy*, 116:842–856, February 2018. ISSN 0960-1481. doi: 10.1016/j.renene.2017.09.058. URL https://www.sciencedirect.com/science/article/pii/S0960148117309205.

Satish Balay, William D. Gropp, Lois Curfman McInnes, and Barry F. Smith. Efficient Management of Parallelism in Object Oriented Numerical Software Libraries. In E. Arge, A. M. Bruaset, and H. P. Langtangen, editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press, 1997.

Jeremy N Bassis and Y Ma. Evolution of basal crevasses links ice shelf stability to ocean forcing. *Earth and Planetary Science Letters*, 409:203–211, 2015.

Aike Beckmann, Hartmut H Hellmer, and Ralph Timmermann. A numerical model of the weddell sea: Large-scale circulation and water mass distribution. *Journal of Geophysical Research: Oceans*, 104(C10):23375–23391, 1999.

Carolyn Branecky Begeman, Xylar Asay-Davis, and Luke Van Roekel. Ice-shelf ocean boundary layer dynamics from large-eddy simulations. *The Cryosphere*, 16(1):277–295, January 2022. ISSN 1994-0416. doi: 10.5194/tc-16-277-2022. URL `https://tc.copernicus.org/articles/16/277/2022/`. Publisher: Copernicus GmbH.

Douglas I Benn, Adrian Luckman, Jan A Åström, Anna J Crawford, Stephen L Cornford, Suzanne L Bevan, Thomas Zwinger, Rupert Gladstone, Karen Alley, Erin Pettit, et al. Rapid fragmentation of thwaites eastern ice shelf. *The Cryosphere*, 16(6):2545–2564, 2022.

Gheorghe-Teodor Bercea, Andrew T. T. McRae, David A. Ham, Lawrence Mitchell, Florian Rathgeber, Luigi Nardi, Fabio Luporini, and Paul H. J. Kelly. A structure-exploiting numbering algorithm for finite elements on extruded meshes, and its performance evaluation in Firedrake. *Geoscientific Model Development*, 9(10):3803–3815, October 2016. ISSN 1991-959X. doi: 10.5194/gmd-9-3803-2016. URL `https://gmd.copernicus.org/articles/9/3803/2016/`. Publisher: Copernicus GmbH.

Rainer Bleck, Claes Rooth, Dingming Hu, and Linda T Smith. Salinity-driven thermocline transients in a wind-and thermohaline-forced isopycnic coordinate model of the north atlantic. *Journal of Physical Oceanography*, 22(12):1486–1505, 1992.

Alexandre Joel Chorin. The numerical solution of the Navier-Stokes equations for an incompressible fluid. *Bulletin of the American Mathematical Society*, 73(6):928–931, 1967. ISSN 0002-9904, 1936-881X. doi: 10.1090/S0002-9904-1967-11853-6. URL `https://www.ams.org/bull/1967-73-06/S0002-9904-1967-11853-6/`.

C. J. Cotter and D. A. Ham. Numerical wave propagation for the triangular P1DG–P2 finite element pair. *Journal of Computational Physics*, 230(8):2806–2820, April 2011. ISSN 0021-

9991. doi: 10.1016/j.jcp.2010.12.024. URL https://www.sciencedirect.com/science/article/pii/S0021999110006984.

Colin J. Cotter, David A. Ham, and Christopher C. Pain. A mixed discontinuous/continuous finite element pair for shallow-water ocean modelling. *Ocean Modelling*, 26(1):86–90, January 2009a. ISSN 1463-5003. doi: 10.1016/j.ocemod.2008.09.002. URL https://www.sciencedirect.com/science/article/pii/S1463500308001479.

Colin J. Cotter, David A. Ham, Christopher C. Pain, and Sebastian Reich. LBB stability of a mixed Galerkin finite element pair for fluid flow simulations. *Journal of Computational Physics*, 228(2):336–348, February 2009b. ISSN 0021-9991. doi: 10.1016/j.jcp.2008.09.014. URL https://www.sciencedirect.com/science/article/pii/S0021999108004816.

Louis-Alexandre Couston, Eric Hester, Benjamin Favier, John R. Taylor, Paul R. Holland, and Adrian Jenkins. Topography generation by melting and freezing in a turbulent shear flow. *Journal of Fluid Mechanics*, 911, March 2021. ISSN 0022-1120, 1469-7645. doi: 10.1017/jfm.2020.1064. URL https://www.cambridge.org/core/journals/journal-of-fluid-mechanics/article/topography-generation-by-melting-and-freezing-in-a-turbulent-shear-flow/E142533DC7BA22F48167C209C86CBA85. Publisher: Cambridge University Press.

Kurt M Cuffey and William Stanley Bryce Paterson. *The physics of glaciers*. Academic Press, 2010.

Véronique Dansereau, Patrick Heimbach, and Martin Losch. Simulation of subice shelf melt rates in a general circulation model: Velocity-dependent transfer and the role of friction. *Journal of Geophysical Research: Oceans*, 119(3):1765–1790, 2014. ISSN 2169-9291. doi: 10.1002/2013JC008846. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2013JC008846. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2013JC008846.

Peter ED Davis, Keith W Nicholls, David M Holland, Britney E Schmidt, Peter Washam, Kiya L Riverman, Robert J Arthern, Irena Vaňková, Clare Eayrs, James A Smith, et al.

Suppressed basal melting in the eastern thwaites glacier grounding zone. *Nature*, 614(7948): 479–485, 2023.

Robert M DeConto and David Pollard. Contribution of antarctica to past and future sea-level rise. *Nature*, 531(7596):591–597, 2016.

Michael S. Dinniman, Xylar S. Asay-Davis, Benjamin K. Galton-Fenzi, Paul R. Holland, Adrian Jenkins, and Ralph Timmermann. Modeling Ice Shelf/Ocean Interaction in Antarctica: A REVIEW. *Oceanography*, 29(4):144–153, 2016. ISSN 1042-8275. URL `https://www.jstor.org/stable/24862289`. Publisher: Oceanography Society.

Christine F Dow, Won Sang Lee, Jamin S Greenbaum, Chad A Greene, Donald D Blankenship, Kristin Poinar, Alexander L Forrest, Duncan A Young, and Christopher J Zappa. Basal channels drive active surface hydrology and transverse ice shelf fracture. *Science Advances*, 4(6):eaao7212, 2018.

Pierre Dutrieux, Craig Stewart, Adrian Jenkins, Keith W Nicholls, Hugh FJ Corr, Eric Rignot, and Konrad Steffen. Basal terraces on melting ice shelves. *Geophysical Research Letters*, 41 (15):5506–5513, 2014.

Howard C. Elman, David J. Silvester, and Andrew J. Wathen. *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*. Oxford University Press, 2014. ISBN 978-0-19-967880-8. Google-Books-ID: 9zSTAwAAQBAJ.

Yekaterina Epshteyn and Béatrice Rivière. Estimation of penalty parameters for symmetric interior penalty Galerkin methods. *Journal of Computational and Applied Mathematics*, 206(2):843–872, September 2007. ISSN 0377-0427. doi: 10.1016/j.cam.2006.08.029. URL `https://www.sciencedirect.com/science/article/pii/S0377042706005279`.

Ronald M. Errico. What Is an Adjoint Model? *Bulletin of the American Meteorological Society*, 78(11):2577–2592, November 1997. ISSN 0003-0007, 1520-0477. doi: 10.1175/1520-0477(1997)078⟨2577:WIAAM⟩2.0.CO;2. Publisher: American Meteorological Society Section: Bulletin of the American Meteorological Society.

P. E. Farrell, M. D. Piggott, G. J. Gorman, D. A. Ham, C. R. Wilson, and T. M. Bond. Automated continuous verification for numerical simulation. *Geoscientific Model Development*, 4(2):435–449, May 2011. ISSN 1991-959X. doi: 10.5194/gmd-4-435-2011. URL https://gmd.copernicus.org/articles/4/435/2011/. Publisher: Copernicus GmbH.

P. E. Farrell, D. A. Ham, S. W. Funke, and M. E. Rognes. Automated Derivation of the Adjoint of High-Level Transient Finite Element Programs. *SIAM Journal on Scientific Computing*, 35(4):C369–C393, January 2013. ISSN 1064-8275. doi: 10.1137/120873558. URL https://epubs.siam.org/doi/10.1137/120873558. Publisher: Society for Industrial and Applied Mathematics.

Patrick E Farrell. Mathematical background: Adjoints and their applications¶. URL http://www.dolfin-adjoint.org/en/latest/documentation/maths/index.html#dolfin-adjoint-mathematical-background.

Lionel Favier, Nicolas C. Jourdain, Adrian Jenkins, Nacho Merino, Gaël Durand, Olivier Gagliardini, Fabien Gillet-Chaulet, and Pierre Mathiot. Assessment of sub-shelf melting parameterisations using the ocean–ice-sheet coupled model NEMO(v3.6)–Elmer/Ice(v8.3). *Geoscientific Model Development*, 12(6):2255–2283, June 2019. ISSN 1991-959X. doi: 10.5194/gmd-12-2255-2019. URL https://gmd.copernicus.org/articles/12/2255/2019/. Publisher: Copernicus GmbH.

P. Fretwell, H. D. Pritchard, D. G. Vaughan, J. L. Bamber, N. E. Barrand, R. Bell, C. Bianchi, R. G. Bingham, D. D. Blankenship, G. Casassa, G. Catania, D. Callens, H. Conway, A. J. Cook, H. F. J. Corr, D. Damaske, V. Damm, F. Ferraccioli, R. Forsberg, S. Fujita, Y. Gim, P. Gogineni, J. A. Griggs, R. C. A. Hindmarsh, P. Holmlund, J. W. Holt, R. W. Jacobel, A. Jenkins, W. Jokat, T. Jordan, E. C. King, J. Kohler, W. Krabill, M. Riger-Kusk, K. A. Langley, G. Leitchenkov, C. Leuschen, B. P. Luyendyk, K. Matsuoka, J. Mouginot, F. O. Nitsche, Y. Nogi, O. A. Nost, S. V. Popov, E. Rignot, D. M. Rippin, A. Rivera, J. Roberts, N. Ross, M. J. Siegert, A. M. Smith, D. Steinhage, M. Studinger, B. Sun, B. K. Tinto, B. C. Welch, D. Wilson, D. A. Young, C. Xiangbin, and A. Zirizzotti. Bedmap2: improved ice bed, surface and thickness datasets for Antarctica. *The Cryosphere*, 7(1):375–393, February

2013. ISSN 1994-0416. doi: 10.5194/tc-7-375-2013. URL https://tc.copernicus.org/articles/7/375/2013/. Publisher: Copernicus GmbH.

Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, 2009. ISSN 1097-0207. doi: 10.1002/nme.2579. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2579. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2579.

Thomas H. Gibson, Lawrence Mitchell, David A. Ham, and Colin J. Cotter. Slate: extending Firedrake's domain-specific abstraction to hybridized solvers for geoscience and beyond. *Geoscientific Model Development*, 13(2):735–761, February 2020. ISSN 1991-959X. doi: 10.5194/gmd-13-735-2020. URL https://gmd.copernicus.org/articles/13/735/2020/. Publisher: Copernicus GmbH.

D. N. Goldberg, N. Gourmelen, S. Kimura, R. Millan, and K. Snow. How Accurately Should We Model Ice Shelf Melt Rates? *Geophysical Research Letters*, 46(1):189–199, 2019. ISSN 1944-8007. doi: 10.1029/2018GL080383. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2018GL080383. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2018GL080383.

D. N. Goldberg, T. A. Smith, S. H. K. Narayanan, P. Heimbach, and M. Morlighem. Bathymetric Influences on Antarctic Ice-Shelf Melt Rates. *Journal of Geophysical Research: Oceans*, 125(11):e2020JC016370, 2020. ISSN 2169-9291. doi: 10.1029/2020JC016370. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2020JC016370. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2020JC016370.

JP Gosink and TE Osterkamp. Measurements and analyses of velocity profiles and frazil ice-crystal rise velocities during periods of frazil-ice formation in rivers. *Annals of Glaciology*, 4: 79–84, 1983.

JS Greenbaum, DD Blankenship, DA Young, TG Richter, JL Roberts, ARA Aitken, B Legresy,

DM Schroeder, RC Warner, TD Van Ommen, et al. Ocean access to a cavity beneath totten glacier in east antarctica. *Nature Geoscience*, 8(4):294–298, 2015.

Stephen M. Griffies, Ronald C. Pacanowski, and Robert W. Hallberg. Spurious Diapycnal Mixing Associated with Advection in a z-Coordinate Ocean Model. *Monthly Weather Review*, 128(3):538–564, March 2000. ISSN 1520-0493, 0027-0644. doi: 10.1175/1520-0493(2000)128⟨0538:SDMAWA⟩2.0.CO;2. Publisher: American Meteorological Society Section: Monthly Weather Review.

David E. Gwyther, Kazuya Kusahara, Xylar S. Asay-Davis, Michael S. Dinniman, and Benjamin K. Galton-Fenzi. Vertical processes and resolution impact ice shelf basal melting: A multi-model study. *Ocean Modelling*, 147:101569, March 2020. ISSN 1463-5003. doi: 10.1016/j.ocemod.2020.101569. URL https://www.sciencedirect.com/science/article/pii/S1463500319301854.

Robert L Haney. On the pressure gradient force over steep topography in sigma coordinate ocean models. 1991.

Tore Hattermann, Lars Henrik Smedsrud, Ole Anders Nøst, Jonathan M Lilly, and Ben K Galton-Fenzi. Eddy-resolving simulations of the fimbul ice shelf cavity circulation: Basal melting and exchange with open ocean. *Ocean Modelling*, 82:28–44, 2014.

Patrick Heimbach and Martin Losch. Adjoint sensitivities of sub-ice-shelf melt rates to ocean circulation under the Pine Island Ice Shelf, West Antarctica. *Annals of Glaciology*, 53(60):59–69, 2012. ISSN 0260-3055, 1727-5644. doi: 10.3189/2012/AoG60A025. URL https://www.cambridge.org/core/journals/annals-of-glaciology/article/adjoint-sensitivities-of-subiceshelf-melt-rates-to-ocean-circulation-under-the-pine-i54DC1B26E790FBE29F2C9262B4BF6DAB. Publisher: Cambridge University Press.

H. H. Hellmer and D. J. Olbers. A two-dimensional model for the thermohaline circulation under an ice shelf. *Antarctic Science*, 1(4):325–336, December 1989. ISSN 1365-2079, 0954-1020. doi: 10.1017/S0954102089000490. URL https://www.cambridge.org/core/journals/antarctic-science/article/abs/

twodimensional-model-for-the-thermohaline-circulation-under-an-ice-shelf/DCB7E3D1C510375CD52CBBD6C42A3993. Publisher: Cambridge University Press.

Van Emden Henson and Ulrike Meier Yang. BoomerAMG: A parallel algebraic multigrid solver and preconditioner. *Applied Numerical Mathematics*, 41(1):155–177, April 2002. ISSN 0168-9274. doi: 10.1016/S0168-9274(01)00115-5. URL https://www.sciencedirect.com/science/article/pii/S0168927401001155.

Hannah Rebecca Hiester. *The application of adaptive mesh techniques to numerical simulations of gravity current flows*. PhD thesis, Imperial College London, 2012.

Koen Hillewaert. *Development of the discontinuous Galerkin method for high-resolution, large scale CFD and acoustics in industrial geometries*. PhD thesis, University of Liege, February 2013.

David M. Holland and Adrian Jenkins. Modeling Thermodynamic Ice–Ocean Interactions at the Base of an Ice Shelf. *Journal of Physical Oceanography*, 29(8):1787–1800, August 1999. ISSN 0022-3670, 1520-0485. doi: 10.1175/1520-0485(1999)029⟨1787:MTIOIA⟩2.0.CO;2. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

David M Holland and Adrian Jenkins. Adaptation of an isopycnic coordinate ocean model for the study of circulation beneath ice shelves. *Monthly Weather Review*, 129(8):1905–1927, 2001.

Paul R. Holland. A model of tidally dominated ocean processes near ice shelf grounding lines. *Journal of Geophysical Research: Oceans*, 113(C11), 2008. ISSN 2156-2202. doi: 10.1029/2007JC004576. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2007JC004576. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2007JC004576.

SS Jacobs, HH Helmer, CSM Doake, A Jenkins, and RM Frolich. Melting of ice shelves and the mass balance of antarctica. *Journal of Glaciology*, 38(130):375–387, 1992.

Adrian Jenkins. A Simple Model of the Ice Shelf–Ocean Boundary Layer and Current. *Journal of Physical Oceanography*, 46(6):1785–1803, June 2016. ISSN 0022-3670, 1520-0485. doi:

10.1175/JPO-D-15-0194.1. URL https://journals.ametsoc.org/view/journals/phoc/46/6/jpo-d-15-0194.1.xml. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

Adrian Jenkins. Shear, Stability, and Mixing within the Ice Shelf–Ocean Boundary Current. *Journal of Physical Oceanography*, 51(7):2129–2148, July 2021. ISSN 0022-3670, 1520-0485. doi: 10.1175/JPO-D-20-0096.1. URL https://journals.ametsoc.org/view/journals/phoc/51/7/JPO-D-20-0096.1.xml. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

Adrian Jenkins and Andreas Bombosch. Modeling the effects of frazil ice crystals on the dynamics and thermodynamics of Ice Shelf Water plumes. *Journal of Geophysical Research: Oceans*, 100(C4):6967–6981, 1995. ISSN 2156-2202. doi: 10.1029/94JC03227. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/94JC03227. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/94JC03227.

Adrian Jenkins and David Holland. Melting of floating ice and sea level rise. *Geophysical Research Letters*, 34(16), 2007.

Adrian Jenkins, Hartmut H. Hellmer, and David M. Holland. The Role of Meltwater Advection in the Formulation of Conservative Boundary Conditions at an Ice–Ocean Interface. *Journal of Physical Oceanography*, 31(1):285–296, January 2001. ISSN 0022-3670, 1520-0485. doi: 10.1175/1520-0485(2001)031⟨0285:TROMAI⟩2.0.CO;2. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

Adrian Jenkins, Keith W. Nicholls, and Hugh F. J. Corr. Observation and Parameterization of Ablation at the Base of Ronne Ice Shelf, Antarctica. *Journal of Physical Oceanography*, 40(10):2298–2312, October 2010. ISSN 0022-3670, 1520-0485. doi: 10.1175/2010JPO4317.1. URL https://journals.ametsoc.org/view/journals/phoc/40/10/2010jpo4317.1.xml. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

James R. Jordan, Paul R. Holland, Adrian Jenkins, Matthew D. Piggott, and Satoshi Kimura. Modeling ice-ocean interaction in ice-shelf crevasses. *Journal of Geophysical Re-*

search: Oceans, 119(2):995–1008, 2014. ISSN 2169-9291. doi: 10.1002/2013JC009208. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2013JC009208. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2013JC009208.

James R. Jordan, Paul R. Holland, Dan Goldberg, Kate Snow, Robert Arthern, Jean-Michel Campin, Patrick Heimbach, and Adrian Jenkins. Ocean-Forced Ice-Shelf Thinning in a Synchronously Coupled Ice-Ocean Model. *Journal of Geophysical Research: Oceans*, 123(2):864–882, 2018. ISSN 2169-9291. doi: 10.1002/2017JC013251. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2017JC013251. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2017JC013251.

Eugenia Kalnay. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2003. ISBN 978-0-521-79179-3. Google-Books-ID: zx_BakP2I5gC.

Ala Khazendar and Adrian Jenkins. A model of marine ice formation within Antarctic ice shelf rifts. *Journal of Geophysical Research: Oceans*, 108(C7), 2003. ISSN 2156-2202. doi: 10.1029/2002JC001673. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2002JC001673. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2002JC001673.

Satoshi Kimura, Adam S. Candy, Paul R. Holland, Matthew D. Piggott, and Adrian Jenkins. Adaptation of an unstructured-mesh, finite-element ocean model to the simulation of ocean circulation beneath ice shelves. *Ocean Modelling*, 67:39–51, July 2013. ISSN 1463-5003. doi: 10.1016/j.ocemod.2013.03.004. URL https://www.sciencedirect.com/science/article/pii/S1463500313000449.

Satoshi Kimura, Adrian Jenkins, Heather Regan, Paul R. Holland, Karen M. Assmann, Daniel B. Whitt, Melchoir Van Wessem, Willem Jan van de Berg, Carleen H. Reijmer, and Pierre Dutrieux. Oceanographic Controls on the Variability of Ice-Shelf Basal Melting and Circulation of Glacial Meltwater in the Amundsen Sea Embayment, Antarctica. *Journal of Geophysical Research: Oceans*, 122(12):10131–10155, 2017. ISSN 2169-9291. doi: 10.1002/2017JC012926. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2017JC012926. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2017JC012926.

Robert C. Kirby, Anders Logg, Marie E. Rognes, and Andy R. Terrel. Common and unusual finite elements. In Anders Logg, Kent-Andre Mardal, and Garth Wells, editors, *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*, Lecture Notes in Computational Science and Engineering, pages 95–119. Springer, Berlin, Heidelberg, 2012. ISBN 978-3-642-23099-8. doi: 10.1007/978-3-642-23099-8$_$3. URL https://doi.org/10.1007/978-3-642-23099-8_3.

S. C. Kramer, C. J. Cotter, and C. C. Pain. Solving the Poisson equation on small aspect ratio domains using unstructured meshes. *Ocean Modelling*, 35(3):253–263, January 2010. ISSN 1463-5003. doi: 10.1016/j.ocemod.2010.08.001. URL https://www.sciencedirect.com/science/article/pii/S1463500310001241.

Dmitri Kuzmin. A vertex-based hierarchical slope limiter for p-adaptive discontinuous Galerkin methods. *Journal of Computational and Applied Mathematics*, 233(12):3077–3085, April 2010. ISSN 0377-0427. doi: 10.1016/j.cam.2009.05.028. URL https://www.sciencedirect.com/science/article/pii/S0377042709003318.

Dmitri Kuzmin, Otto Mierka, and Stefan Turek. On the implementation of the κ-ε turbulence model in incompressible flow solvers based on a finite element discretisation. *International Journal of Computing Science and Mathematics*, 1(2-4):193–206, 2007.

Tuomas Kärnä, Stephan C. Kramer, Lawrence Mitchell, David A. Ham, Matthew D. Piggott, and António M. Baptista. Thetis coastal ocean model: discontinuous Galerkin discretization for the three-dimensional hydrostatic equations. *Geoscientific Model Development*, 11(11):4359–4382, October 2018. ISSN 1991-959X. doi: 10.5194/gmd-11-4359-2018. URL https://gmd.copernicus.org/articles/11/4359/2018/. Publisher: Copernicus GmbH.

E. L. Lewis and R. G. Perkin. Ice pumps and their rates. *Journal of Geophysical Research: Oceans*, 91(C10):11756–11762, 1986. ISSN 2156-2202. doi: 10.1029/JC091iC10p11756. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/JC091iC10p11756. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/JC091iC10p11756.

M. Losch. Modeling ice shelf cavities in a z coordinate ocean general circulation

model. *Journal of Geophysical Research: Oceans*, 113(C8), 2008. ISSN 2156-2202. doi: 10.1029/2007JC004368. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2007JC004368. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2007JC004368.

John Marshall, Alistair Adcroft, Chris Hill, Lev Perelman, and Curt Heisey. A finite-volume, incompressible Navier Stokes model for studies of the ocean on parallel computers. *Journal of Geophysical Research: Oceans*, 102(C3):5753–5766, 1997. ISSN 2156-2202. doi: 10.1029/96JC02775. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/96JC02775. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/96JC02775.

Miles McPhee. Similarity for the Ice/Ocean Boundary Layer. In Miles McPhee, editor, *Air-Ice-Ocean Interaction: Turbulent Ocean Boundary Layer Exchange Processes*, pages 65–86. Springer, New York, NY, 2008. ISBN 978-0-387-78335-2. doi: 10.1007/978-0-387-78335-2$_$4.

A. T. T. McRae, G.-T. Bercea, L. Mitchell, D. A. Ham, and C. J. Cotter. Automated Generation and Symbolic Manipulation of Tensor Product Finite Elements. *SIAM Journal on Scientific Computing*, 38(5):S25–S47, January 2016. ISSN 1064-8275. doi: 10.1137/15M1021167. URL https://epubs.siam.org/doi/10.1137/15M1021167. Publisher: Society for Industrial and Applied Mathematics.

John Hainsworth Mercer. West antarctic ice sheet and co2 greenhouse effect: a threat of disaster. *Nature*, 271(5643):321–325, 1978.

Aaron Meurer, Christopher P. Smith, Mateusz Paprocki, Ondřej Čertík, Sergey B. Kirpichev, Matthew Rocklin, AMiT Kumar, Sergiu Ivanov, Jason K. Moore, Sartaj Singh, Thilina Rathnayake, Sean Vig, Brian E. Granger, Richard P. Muller, Francesco Bonazzi, Harsh Gupta, Shivam Vats, Fredrik Johansson, Fabian Pedregosa, Matthew J. Curry, Andy R. Terrel, Stěpán Roucka, Ashutosh Saboo, Isuru Fernando, Sumith Kulal, Robert Cimrman, and Anthony Scopatz. SymPy: symbolic computing in Python. *PeerJ Computer Science*, 3: e103, January 2017. ISSN 2376-5992. doi: 10.7717/peerj-cs.103. URL https://peerj.com/articles/cs-103. Publisher: PeerJ Inc.

Leo Middleton, Catherine A. Vreugdenhil, Paul R. Holland, and John R. Taylor. Numerical Simulations of Melt-Driven Double-Diffusive Fluxes in a Turbulent Boundary Layer beneath an Ice Shelf. *Journal of Physical Oceanography*, 51(2):403–418, January 2021. ISSN 0022-3670, 1520-0485. doi: 10.1175/JPO-D-20-0114.1. URL `https://journals.ametsoc.org/view/journals/phoc/51/2/jpo-d-20-0114.1.xml`. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

P. Milillo, E. Rignot, P. Rizzoli, B. Scheuchl, J. Mouginot, J. Bueso-Bello, and P. Prats-Iraola. Heterogeneous retreat and ice melt of Thwaites Glacier, West Antarctica. *Science Advances*, 5(1):eaau3433, 2019. doi: 10.1126/sciadv.aau3433. URL `https://www.science.org/doi/full/10.1126/sciadv.aau3433`. Publisher: American Association for the Advancement of Science.

MEA Millennium ecosystem assessment. *Ecosystems and human well-being*, volume 1. Island press Washington, DC, 2005.

FJ Millero. Freezing point of sea water. *eighth report of the Joint Panel of Oceanographic Tables and Standards, appendix*, 6:29–31, 1978.

Thomas Millgate, Paul R Holland, Adrian Jenkins, and Helen L Johnson. The effect of basal channels on oceanic ice-shelf melting. *Journal of Geophysical Research: Oceans*, 118(12): 6951–6964, 2013.

Sebastian K. Mitusch, Simon W. Funke, and Jørgen S. Dokken. dolfin-adjoint 2018.1: automated adjoints for FEniCS and Firedrake. *Journal of Open Source Software*, 4(38):1292, June 2019. ISSN 2475-9066. doi: 10.21105/joss.01292. URL `https://joss.theoj.org/papers/10.21105/joss.01292`.

Mathieu Morlighem, Daniel Goldberg, Thiago Dias dos Santos, Jane Lee, and Max Sagebaum. Mapping the Sensitivity of the Amundsen Sea Embayment to Changes in External Forcings Using Automatic Differentiation. *Geophysical Research Letters*, 48(23):e2021GL095440, 2021. ISSN 1944-8007. doi: 10.1029/2021GL095440.

URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2021GL095440. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021GL095440.

J. Mouginot, E. Rignot, and B. Scheuchl. Sustained increase in ice discharge from the Amundsen Sea Embayment, West Antarctica, from 1973 to 2013. *Geophysical Research Letters*, 41(5):1576–1584, 2014. ISSN 1944-8007. doi: 10.1002/2013GL059069. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2013GL059069. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2013GL059069.

Y. Nakayama, D. Menemenlis, M. Schodlok, and E. Rignot. Amundsen and Bellingshausen Seas simulation with optimized ocean, sea ice, and thermodynamic ice shelf model parameters. *Journal of Geophysical Research: Oceans*, 122(8):6180–6195, 2017. ISSN 2169-9291. doi: 10.1002/2016JC012538. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2016JC012538. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2016JC012538.

Yoshihiro Nakayama, Cilan Cai, and Helene Seroussi. Impact of Subglacial Freshwater Discharge on Pine Island Ice Shelf. *Geophysical Research Letters*, 48 (18):e2021GL093923, 2021a. ISSN 1944-8007. doi: 10.1029/2021GL093923. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2021GL093923. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021GL093923.

Yoshihiro Nakayama, Dimitris Menemenlis, Ou Wang, Hong Zhang, Ian Fenty, and An T. Nguyen. Development of adjoint-based ocean state estimation for the Amundsen and Bellingshausen seas and ice shelf cavities using MITgcm–ECCO (66j). *Geoscientific Model Development*, 14(8):4909–4924, August 2021b. ISSN 1991-959X. doi: 10.5194/gmd-14-4909-2021. URL https://gmd.copernicus.org/articles/14/4909/2021/. Publisher: Copernicus GmbH.

Kaitlin A. Naughten, Jan De Rydt, Sebastian H. R. Rosier, Adrian Jenkins, Paul R. Holland, and Jeff K. Ridley. Two-timescale response of a large Antarctic ice shelf to climate change. *Nature Communications*, 12(1):1991, March 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-22259-0. URL https://www.nature.com/articles/s41467-021-22259-0. Number: 1 Publisher: Nature Publishing Group.

Reuben W. Nixon-Hill, Daniel Shapero, Colin J. Cotter, and David A. Ham. Consistent point data assimilation in firedrake and icepack, 2023.

O Orheim, JO Hagen, S Østerhus, and AC Saetrang. Glaciologic and oceanographic studies on Fimbulisen during NARE 1989/90. *Filchner-Ronne Ice Shelf Programme Rep*, 4:120–139, 1990. Alfred-Wegener Inst. for Polar and Mar. Res., Bremerhaven, Germany.

Joseph Pedlosky. *Geophysical fluid dynamics*. Springer Science & Business Media, 2013.

M. D. Piggott, G. J. Gorman, C. C. Pain, P. A. Allison, A. S. Candy, B. T. Martin, and M. R. Wells. A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes. *International Journal for Numerical Methods in Fluids*, 56(8):1003–1015, 2008a. ISSN 1097-0363. doi: 10.1002/fld.1663. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1663`. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/fld.1663.

Matthew D Piggott, Christopher C Pain, Gerard J Gorman, David P Marshall, and Peter D Killworth. Unstructured adaptive meshes for ocean modeling. *Ocean modeling in an eddying regime*, 177:383–408, 2008b.

MD Piggott, PE Farrell, CR Wilson, GJ Gorman, and CC Pain. Anisotropic mesh adaptivity for multi-scale ocean modelling. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1907):4591–4611, 2009.

QGIS Development Team. QGIS Geographic Information System, 2022. URL `https://www.qgis.org`.

Florian Rathgeber, David A. Ham, Lawrence Mitchell, Michael Lange, Fabio Luporini, Andrew T. T. Mcrae, Gheorghe-Teodor Bercea, Graham R. Markall, and Paul H. J. Kelly. Firedrake: Automating the Finite Element Method by Composing Abstractions. *ACM Transactions on Mathematical Software*, 43(3):24:1–24:27, December 2016. ISSN 0098-3500. doi: 10.1145/2998441. URL `https://doi.org/10.1145/2998441`.

E. Rignot, S. Jacobs, J. Mouginot, and B. Scheuchl. Ice-Shelf Melting Around Antarctica. *Science*, 341(6143):266–270, July 2013. doi: 10.1126/science.1235798. URL `https://www.`

science.org/doi/full/10.1126/science.1235798. Publisher: American Association for the Advancement of Science.

Todd Ringler, Mark Petersen, Robert L. Higdon, Doug Jacobsen, Philip W. Jones, and Mathew Maltrud. A multi-resolution approach to global ocean modeling. *Ocean Modelling*, 69:211–232, September 2013. ISSN 1463-5003. doi: 10.1016/j.ocemod.2013.04.010. URL https://www.sciencedirect.com/science/article/pii/S1463500313000760.

Patrick J. Roache. *Verification and validation in computational science and engineering*, volume 895. Hermosa, Albuquerque, New Mexico, 1998.

Kambiz Salari and Patrick Knupp. Code Verification by the Method of Manufactured Solutions. Technical Report SAND2000-1444, Sandia National Lab. (SNL-NM), Albuquerque, NM (United States); Sandia National Lab. (SNL-CA), Livermore, CA (United States), June 2000. URL https://www.osti.gov/biblio/759450/.

T. A. Scambos, R. E. Bell, R. B. Alley, S. Anandakrishnan, D. H. Bromwich, K. Brunt, K. Christianson, T. Creyts, S. B. Das, R. DeConto, P. Dutrieux, H. A. Fricker, D. Holland, J. MacGregor, B. Medley, J. P. Nicolas, D. Pollard, M. R. Siegfried, A. M. Smith, E. J. Steig, L. D. Trusel, D. G. Vaughan, and P. L. Yager. How much, how fast?: A science review and outlook for research on the instability of Antarctica's Thwaites Glacier in the 21st century. *Global and Planetary Change*, 153:16–34, June 2017. ISSN 0921-8181. doi: 10.1016/j.gloplacha.2017.04.008. URL https://www.sciencedirect.com/science/article/pii/S092181811630491X.

Britney E Schmidt, Peter Washam, Peter ED Davis, Keith W Nicholls, David M Holland, Justin D Lawrence, Kiya L Riverman, James A Smith, A Spears, DJG Dichek, et al. Heterogeneous melting near the thwaites glacier grounding line. *Nature*, 614(7948):471–478, 2023.

Frank Schnaase and Ralph Timmermann. Representation of shallow grounding zones in an ice shelf-ocean model with terrain-following coordinates. *Ocean Modelling*, 144:101487, 2019.

William I Scott, Stephan C Kramer, Paul R Holland, Keith W Nicholls, Martin J Siegert, and Matthew D Piggott. Towards a fully unstructured ocean model for ice shelf cavity envi-

ronments: Model development and verification using the firedrake finite element framework. *Ocean Modelling*, 182:102178, 2023.

OV Sergienko. Basal channels on ice shelves. *Journal of Geophysical Research: Earth Surface*, 118(3):1342–1355, 2013.

H. Seroussi, Y. Nakayama, E. Larour, D. Menemenlis, M. Morlighem, E. Rignot, and A. Khazendar. Continued retreat of Thwaites Glacier, West Antarctica, controlled by bed topography and ocean circulation. *Geophysical Research Letters*, 44(12):6191–6199, 2017. ISSN 1944-8007. doi: 10.1002/2017GL072910. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/2017GL072910`. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2017GL072910.

Daniel R. Shapero, Jessica A. Badgeley, Andrew O. Hoffman, and Ian R. Joughin. icepack: a new glacier flow modeling package in Python, version 1.0. *Geoscientific Model Development*, 14(7):4593–4616, July 2021. ISSN 1991-959X. doi: 10.5194/gmd-14-4593-2021. URL `https://gmd.copernicus.org/articles/14/4593/2021/`. Publisher: Copernicus GmbH.

Alexander F Shchepetkin and James C McWilliams. The regional oceanic modeling system (roms): a split-explicit, free-surface, topography-following-coordinate oceanic model. *Ocean modelling*, 9(4):347–404, 2005.

Andrew Shepherd, Helen Amanda Fricker, and Sinead Louise Farrell. Trends and connections across the Antarctic cryosphere. *Nature*, 558(7709):223–232, June 2018. ISSN 1476-4687. doi: 10.1038/s41586-018-0171-6. URL `https://www.nature.com/articles/s41586-018-0171-6`. Number: 7709 Publisher: Nature Publishing Group.

Andrew John Sole, Douglas Watson Fraser Mair, Peter W Nienow, ID Bartholomew, MA King, Matt J Burke, and Ian Joughin. Seasonal speedup of a greenland marine-terminating outlet glacier forced by surface melt–induced changes in subglacial hydrology. *Journal of Geophysical Research: Earth Surface*, 116(F3), 2011.

Yuhe Song and Dale Haidvogel. A semi-implicit ocean circulation model using a generalized

topography-following coordinate system. *Journal of Computational Physics*, 115(1):228–244, 1994.

Gilbert Strang. *Computational Science and Engineering*, volume 791. Wellesley-Cambridge Press Wellesley, 2007.

R. Timmermann, Q. Wang, and H. H. Hellmer. Ice-shelf basal melting in a global finite-element sea-ice/ice-shelf/ocean model. *Annals of Glaciology*, 53(60):303–314, 2012. ISSN 0260-3055, 1727-5644. doi: 10.3189/2012AoG60A156. URL https://www.cambridge.org/core/journals/annals-of-glaciology/article/iceshelf-basal-melting-in-a-global-finiteelement-seaiceiceshelfocean-model/B8724B766A10BD37FBB6B3F6C70F9089. Publisher: Cambridge University Press.

Lloyd N. Trefethen and David Bau III. *Numerical Linear Algebra*. SIAM, June 1997. ISBN 978-0-89871-361-9. Google-Books-ID: 4Mou5YpRD_kC.

Jean Utke, Uwe Naumann, Mike Fagan, Nathan Tallent, Michelle Strout, Patrick Heimbach, Chris Hill, and Carl Wunsch. OpenAD/F: A Modular Open-Source Tool for Automatic Differentiation of Fortran Codes. *ACM Transactions on Mathematical Software*, 34(4):18:1–18:36, July 2008. ISSN 0098-3500. doi: 10.1145/1377596.1377598. URL https://doi.org/10.1145/1377596.1377598.

Geoffrey K. Vallis. *Atmospheric and Oceanic Fluid Dynamics*. Cambridge University Press, June 2017. ISBN 978-1-107-06550-5. Google-Books-ID: BTgoDwAAQBAJ.

David G Vaughan, Hugh FJ Corr, Robert A Bindschadler, Pierre Dutrieux, G Hilmar Gudmundsson, Adrian Jenkins, Thomas Newman, Patricia Vornberger, and Duncan J Wingham. Subglacial melt channels and fracture in the floating part of pine island glacier, antarctica. *Journal of Geophysical Research: Earth Surface*, 117(F3), 2012.

Catherine A. Vreugdenhil and John R. Taylor. Stratification Effects in the Turbulent Boundary Layer beneath a Melting Ice Shelf: Insights from Resolved Large-Eddy Simulations. *Journal of Physical Oceanography*, 49(7):1905–1925, July 2019. ISSN 0022-3670, 1520-0485. doi: 10.1175/JPO-D-18-0252.1. URL https://journals.ametsoc.org/view/journals/phoc/

49/7/jpo-d-18-0252.1.xml. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

Catherine A. Vreugdenhil, John R. Taylor, Peter E. D. Davis, Keith W. Nicholls, Paul R. Holland, and Adrian Jenkins. The Ocean Boundary Layer beneath Larsen C Ice Shelf: Insights from Large-Eddy Simulations with a Near-Wall Model. *Journal of Physical Oceanography*, 52(8):1903–1926, August 2022. ISSN 0022-3670, 1520-0485. doi: 10.1175/JPO-D-21-0166.1. URL https://journals.ametsoc.org/view/journals/phoc/52/8/JPO-D-21-0166.1.xml. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

Joseph G. Wallwork, Nicolas Barral, Stephan C. Kramer, David A. Ham, and Matthew D. Piggott. Goal-oriented error estimation and mesh adaptation for shallow water modelling. *SN Applied Sciences*, 2(6):1053, May 2020. ISSN 2523-3971. doi: 10.1007/s42452-020-2745-9. URL https://doi.org/10.1007/s42452-020-2745-9.

Joseph Gregory Wallwork. Mesh adaptation and adjoint methods for finite element coastal ocean modelling. 2021.

Ray H. Watkins, Jeremy N. Bassis, and M. D. Thouless. Roughness of Ice Shelves Is Correlated With Basal Melt Rates. *Geophysical Research Letters*, 48(21):e2021GL094743, 2021. ISSN 1944-8007. doi: 10.1029/2021GL094743. URL https://onlinelibrary.wiley.com/doi/abs/10.1029/2021GL094743. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021GL094743.

Benjamin Adam Yeager. Moving mesh finite element modeling of ocean circulation beneath ice shelves. October 2018. doi: 10.25560/73697. URL http://spiral.imperial.ac.uk/handle/10044/1/73697. Accepted: 2019-10-01T15:36:11Z Publisher: Imperial College London.

Qin Zhou and Tore Hattermann. Modeling ice shelf cavities in the unstructured-grid, Finite Volume Community Ocean Model: Implementation and effects of resolving small-scale topography. *Ocean Modelling*, 146:101536, February 2020. ISSN 1463-5003. doi: 10.

1016/j.ocemod.2019.101536. URL https://www.sciencedirect.com/science/article/pii/S1463500319301738.