Viscoelastic Modelling of Crustal Deformation

James D. P. Moore, M.Sci. (Dunelm)

Thesis submitted to the University of Oxford
for the degree of Doctor of Philosophy
in
Geology and Mineralogy



Department of Earth Sciences
and St Cross College,
University of Oxford

Michaelmas, 2014

Supervised by Prof. B. Parsons

Declaration

The contents of this thesis are all my own work, except where otherwise stated. The views and opinions expressed herein are mine and not necessarily those of any other person or body unless so attributed.

Citation: Moore, J. D. P. (2014), Viscoelastic Modelling of Crustal Deformation, D.Phil. Thesis, University of Oxford, Department of Earth Sciences, Oxford, UK.

Keywords: Earthquake, Defomation, Postseismic, Faulting, Numerical Modelling, Santorini, Viscoelasticity, Rheology.

This thesis has been typeset using LaTeX 2_ε and references were compiled using BibTeX. Many of the figures and calculations were produced using Wolfram Mathematica, *Wolfram Research* (2012).

Copyright ©, by James Moore, 2014.
All rights reserved.

No part of the material protected by this copyright notice may be reproduced or utilised in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage and retrieval system, without written permission from the copyright owner.

Printed in the United Kingdom.

Abstract

Viscoelastic Modelling of Crustal Deformation

James Moore

St Cross College, Oxford

Michaelmas Term, 2014

Deformation in continents is not restricted to narrow bands but is spread over great distances within their interiors. A number of lines of evidence, including the distribution of earthquakes, reveal that the strength of different continental regions varies markedly. While it is relatively easy to qualitatively map out these variations, little progress has been made in quantifying the range of strength in the continents and identifying the physical mechanisms that control these variations. I investigate crustal deformation associated with the earthquake cycle, inflation of magma chambers beneath volcanoes, and changes in surface loads. Results of these models has important implications for our understanding of large-scale continental deformation and mountain building, in addition to both seismic and volcanic hazard assessment.

Novel analytic solutions for simple shear with depth-dependent linear and non-linear viscoelastic rheologies are derived, in addition to analytical solutions for imposed harmonic tractions and displacements on an elastic layer over a Maxwell viscoelastic half space. Notably this model exhibits an effective relaxation timescale with wavenumber (k) and elastic lid thickness (d_e) dependence, $t_R = \eta/\mu[1 + \coth(kd_e)]$, a result which elegantly explains observations of past earthquakes where short wavelength deformation has been observed to decay more rapidly than longer wavelengths.

The long term structural form of distributed shear beneath a strike-slip fault is investigated to determine the relative importance of physical mechanisms which may localise shear. For a depth-dependent viscosity, $\eta = \eta_0 \exp(-z/z_0)$, I find a shear zone develops with half-width $\delta_w \sim \sqrt{z_0}$ for small z_0. Including a non-linear stress-strain rate relation ($\dot{\epsilon} \propto \sigma^n$) scales δ_w by $1/\sqrt{n}$, comparable to length scales in thin viscous sheet calculations used for large-scale continental deformation calculations over geologic time scales. In contrast to previous work, this model shows for the first time

that the primary control on the shear zone width is the depth-dependence of viscosity, instead of thermomechanical coupling or a non-linear rheology.

When examining post-seismic transient deformation, a purely non-linear rheology is found to rapidly cease deforming due to the implausibly high viscosities away from the fault, with deformation confined to a region half that of the depth of faulting unless I include far field loading. A full earthquake cycle model with non-linear rheology reproduces the zone of weakness observed in the viscous shear zone model. A new model is also proposed that allows for a combination rheology with dislocation and diffusion creep mechanisms. Initial analytical calculations indicate that the viscosity will then behave linearly, excepting a narrow region of reduced viscosity surrounding the fault tip.

InSAR and GPS measurements from Santorini during a recent phase of unrest led to the opportunity to model a system containing both seismic and volcanic activity. A new model for a shallow magma chamber at Santorini is proposed and demonstrates that two separate pulses of increasing pressure to a chamber, surrounded by a viscoelastic shell of constant viscosity, account for the observed cumulative volume change. The derived shell viscosity and ratio of outer-inner radius match those reported by other studies and are consistent with observations of metamorphic aureoles. Derived shut-off times for the pressure pulses match seismicity observations and suggest that any effective increase in pressure to the shallow magma chamber ceased in early 2012.

These results suggest that combining depth-dependent viscosity and multiple creep mechanisms in a single model will reproduce observations seen across the seismic cycle and may be extended to further our understanding of magmatic processes. Previous models have often invoked heterogeneities in sub-fault properties to explain localisation of deformation at various stages of the earthquake cycle. These results reveal that localisation arises naturally from the physical processes in operation.

Extended Abstract

Viscoelastic Modelling of Crustal Deformation

James Moore

St Cross College, Oxford

Michaelmas Term, 2014

Deformation in continents is not restricted to narrow bands but is spread over great distances within their interiors. A number of lines of evidence, including the distribution of earthquakes, reveal that the strength of different continental regions varies markedly. While it is relatively easy to qualitatively map out these variations, little progress has been made in quantifying the range of strength in the continents and identifying the physical mechanisms that control these variations. I investigate crustal deformation associated with the earthquake cycle, inflation of magma chambers beneath volcanoes, and changes in surface loads. Results of these models have important implications for our understanding of large-scale continental deformation and mountain building, in addition to both seismic and volcanic hazard assessment.

3D finite element viscoelastic deformation in *oregano_ve* and viscoelastic relaxation of imposed harmonic tractions and displacements

I conduct modifications to the 3D finite element program *oregano_ve*, including periodic boundary conditions to simulate anti-plane strain conditions. This allows me to build on simple analytical models, from the literature and my own derivations, with numerical calculations to investigate problems that cannot be solved analytically. In the process of modifying the code I develop novel benchmarks for simple shear of a viscoelastic block with depth dependent viscosity and a non-linear rheology. Subsequently, I develop a model applicable to a Fourier decomposition of an imposed stress field, with the deformation and relaxation timescale associated with each mode. Notably this model exhibits an effective relaxation timescale with wavenumber (k) and elastic lid thickness (d_e) dependance, $t_R = \eta/\mu[1 + \coth(kd_e)]$, i.e. a linear Maxwell rheology is capable of exhibiting multiple relaxation timescales dependant on the wavelength of imposed deformation or stress.

Scaling of viscous shear zones with depth dependant viscosity and power law stress-strain dependence

One of the unresolved questions concerning fault deformation is the degree and cause of localisation of shear at depth beneath a fault. Geologic observations of exhumed shear zones indicate that whilst the motion is no longer planar, it can still be localised near the down-dip extension of the fault; however, the degree of localisation is uncertain. I employ simple analytic and numerical models to investigate the structural form of distributed shear beneath a strike-slip fault, and the relative importance of the physical mechanisms that have the potential to localise a shear zone.

For a purely depth dependent viscosity, $\eta = \eta_0 \exp(-z/z_0)$, I find that a shear zone develops with a half-width $\delta_w \sim \sqrt{z_0}$ for small z_0 at the base of the layer, where lengths are non-dimensionalised by the layer thickness (d km). Including a non-linear stress-strain rate relation ($\dot{\epsilon} \propto \sigma^n$) scales δ_w by $1/\sqrt{n}$, comparable to deformation length scales in thin viscous sheet calculations. I find that the primary control on the shear zone width is the depth dependence of viscosity that arises from the temperature dependence of viscosity and the increase in temperature with depth. As this relationship is exponential, scaling relations give a half-width that scales approximately as

$$\delta_w \approx T_{\frac{1}{2}} \sqrt{\frac{Rd}{nQ\beta}} \text{ km},$$

where $T_{\frac{1}{2}}$ (K) is the temperature at the midpoint of the layer, R (J/mol K) the gas constant, Q (J/mol) the activation energy, and β (K/km) the geothermal gradient. This relation agrees with the numerical results for the parameter range consistent with continental rheologies to within 2-5% and gives shear zone half-widths from 2-6 km.

The inclusion of shear-stress heating reduces δ_w by only an additional 5-25%, depending on the initial width of the shear zone. Whilst the width of the shear zone may not decrease significantly, local temperature increases from shear-stress heating range from 50-300°C resulting in a reduction in viscosities beneath the fault of several orders of magnitude and a concomitant reduction in the stresses needed to drive the motion.

Viscoelastic post-seismic relaxation with a non-linear stress strain-rate relationship

I investigate the post-seismic phase of the earthquake cycle, primarily considering apparent localised weakening in the immediate post-seismic deformation field. I concentrate on the effective viscosity reduction generated by a power-law stress strain-rate

relationship and investigate the effect of two kinds of boundary conditions, free-slip and driving velocities, on post-seismic motions for both single and repeated ruptures. The calculations were performed using the 3D finite element code *oregano_ve* with periodic boundary conditions in the along-strike direction to simulate anti-plane conditions. Ruptures broke the elastic lid (thickness d) and drove displacements in a viscoelastic layer (thickness $20d$) with a non-linear rheology. For the case of a single rupture with free-slip boundary conditions, the post-seismic displacements are half that of the linear case when $n = 2$, and considerably lower for higher values of n. This is due to the very low strain-rates across the domain generating viscosities up to several orders of magnitude higher than the equivalent linear viscosity and suppressing deformation. I find that to ensure deformation commensurate with geodetic observations I need to enforce far field loading. The case of repeated ruptures with far field loading leads to the development of a narrow zone of weakness in agreement with the results of the viscous shear zone models.

Viscoelastic modelling of subsidence and unrest at Santorini Volcano, Greece

Analysis of long-term deformation trends is key to an improved understanding of sub-volcanic processes. InSAR and GPS measurements from Santorini volcano are analysed over an extended period incorporating both a phase of quiescence (1993-2010) and a phase of unrest (January 2011-April 2012) characterised by caldera-wide inflation and seismicity. Mean InSAR velocity maps from 1993-2010 indicate ~ 6 mm/yr of subsidence over the southern half of the intra-caldera island Nea Kameni. I demonstrate that this subsidence may result from a combination of thermal contraction of the 1866-1870 lava flows and load-induced relaxation of the substrate. Including a novel dataset of historic bathymetric measurements allows us to quantify the load due to the 1866-1870

flows and the observed static offsets between the historic 1848 map and 2006 swath bathymetry in conjunction with the InSAR maps to estimate the elastic lid thickness of 1km and viscosity 6.3×10^{15} Pa s.

For the period of unrest, a joint inversion technique is used to convert both InSAR measurements and GPS observations into a time series of sub-surface volume change. The optimal source location is consistent with previous studies, situated north of Nea Kameni at a depth of $\sim$ 4 km. However, the time series reveals two distinct pressure pulses that may be explained by separate batches of melt being delivered to the shallow chamber. The first pulse corresponds to a volume change (ΔV) within the shallow magma chamber of $11.56 \pm 0.14 \times 10^6 \mathrm{m}^3$ and the second pulse has a ΔV of $9.73 \pm 0.10 \times 10^6 \mathrm{m}^3$. The distinctive rollover observed in the volume change curve at the end of each pulse combined with the lag response identified between the cessation in earthquakes within the caldera and the exponential decline in the rate of sub-surface volume change (several months later) is indicative of stress relaxation through viscoelastic processes. I propose a new model for the shallow magma chamber at Santorini volcano and demonstrate that two separate pulses of increasing pressure to a shallow magma chamber, surrounded by a viscoelastic shell, can account for the observed variation in cumulative volume change throughout the period of unrest. The derived shell viscosity (4.7×10^{12} Pa s) and ratio of the outer-inner radius (1.73) are within the expected range reported in other studies. The derived shut-off times for the pressure pulses provide a good fit to seismicity observations and suggest that any effective increase in pressure to the shallow magma chamber ceased at the end of January 2012.

Acknowledgements

I would firstly like to thank my supervisors, Barry Parsons and Philip England, for their guidance throughout my DPhil. I would also like to thank Michelle Parks and Tony Watts for many useful discussions and collaboration working on Santorini. I am extremely grateful to the Burdett-Coutts committee for a maintenance grant which allowed the completion of this DPhil, and the Natural Environmental Research Council who funded my studentship.

I would especially like to thank Greg Houseman and Tadashi Yamasaki for all their help and support thought my DPhil. Discussions with them have lead to many of the avenues I explored in this thesis. I would also like to thank my examiners, Tim Wright and Richard Katz, for a stimulating viva, and for the discussions that have followed.

The COMET group has made the department a great place to work and kept me sane from time to time, so I would like to thank all of them and everyone else who helped to make this thesis a reality, be it by providing tea or proof reading: Eleanor Ainscoe, Ruth Amey, Ivana Barisin, Julia Barrott, Marcus Bell, Emily Carson, Susi Ebmeier, John Elliot, Laura Gregory, Miriam Gordan, Will Hutchinson, David Mackenzie, Fergus McNab, Tim Middleton, Rebecca Morgan, Victoria Nockles, Francesca Silverii, Al Sloan, Richard Walker, Caroline Walters, Richard Walters, and Yu Zhou.

I would finally like to thank my parents, Timothy and Elizabeth Moore, for their unwavering support and encouragement.

Contents

Declaration i

Abstract iii

Extended Abstract v

Acknowledgments xi

Contents xvi

List of Figures xxi

List of Tables xxiii

1 Introduction **1**

1.1 The Earthquake cycle 5

1.1.1 Elastic dislocation theory 6

1.2 Viscoelasticity 11

1.2.1 Constitutive laws 11

1.2.2 The correspondence principle 13

1.3 Deformation of a spherical magma chamber 14

1.4 Overview of thesis 18

2 3D finite element viscoelastic deformation in *oregano_ve* and viscoelastic relaxation of imposed harmonic tractions and displacements **21**

2.1 Introduction to *oregano_ve* 21

2.2 Modifications 25

2.2.1 Code Modifications 25
2.2.2 List of files and subroutines modified 26
2.2.3 Input and output options 27
2.2.4 Modification of Parallelisation Routines for Periodic Boundary Conditions in Z 28
2.3 Benchmarking 30
2.4 Viscoelastic relaxation on a strike slip fault 31
2.5 Simple shear, constant tangental stress 34
2.5.1 Power law rheology 35
2.5.2 Test results from *oregano_ve* 37
2.6 Relaxation of harmonic tractions and displacements 39
2.6.1 Elastic half-space 40
2.6.2 Elastic layer over elastic half-space 41
2.6.3 Elastic layer over viscoelastic half-space with constant shear moduli (μ) 44
2.6.4 Elastic layer over viscoelastic half-space with contrasting shear moduli (μ_e and μ_{ve}) 48
2.6.5 Elastic layer with zero shear traction at base 49
2.6.6 Viscoelastic half-space 50
2.6.7 Analysis of solution to elastic layer over viscoelastic half-space . 52
2.6.8 Analytic solution for periodic displacement condition at the surface 54
2.6.9 Discussion 59
2.7 Conclusions 61

3 Scaling of viscous shear zones with depth dependant viscosity and power law stress-strain dependence 63

3.1 Introduction 63
3.2 Model 65

3.3 Linear ductile shear zones 68
3.3.1 Uniform properties 68
3.3.2 Exponentially depth dependent viscosity 68
3.3.3 Arrhenius temperature dependence 73
3.4 Non-linear ductile shear zones 80
3.4.1 Uniform physical properties 80
3.4.2 Exponentially depth dependent viscosity 84
3.4.3 Arrhenius temperature dependence 86
3.5 Shear-stress heating and thermomechanical coupling 87
3.5.1 Arrhenius temperature dependence 88
3.5.2 Non-linear rheology with Arrhenius temperature dependence . . 91
3.6 Discussion 95
3.7 Conclusions 102

4 Viscoelastic post-seismic relaxation with a non-linear stress strain-rate relationship 103

4.1 Introduction 103
4.2 Model 104
4.3 Free slip condition on far-field boundary 106
4.4 Loading conditions on far-field boundary 114
4.5 Earthquake cycle 121
4.6 Earthquake cycle with driving boundary conditions 124
4.7 Combined power law and linear rheology 127
4.8 Comparison to results from *Yamasaki et al.* (2014) 128
4.9 Discussion 134

5 Viscoelastic modelling of subsidence and unrest at Santorini Volcano, Greece 137

5.1 Introduction 137
5.2 Slow subsidence from 1993-2010 140
5.2.1 Thermal contraction 140
5.2.2 Loading from historic lava flows and viscoelastic relaxation of the substrate 143
5.3 Historical bathymetric charts of Santorini 147
5.3.1 Introduction 147
5.3.2 Collapse and the viscoelastic behaviour of the crust and mantle 151
5.4 Caldera-wide uplift and viscoelastic modelling 160
5.4.1 Introduction, joint inversion etc 160
5.4.2 Viscoelastic Modelling 163
5.5 Conclusions 174

6 Discussion and Conclusions **177**
6.1 Further Work 182

Bibliography **184**

A *oregano_ve* code **193**
A.1 `parconjgr.f` 193
A.2 `conjgr3.f` 205

B Mathematica codes for chapter 3 **223**

C *oregano_ve* job files **227**

D Mathematica codes for plotting post-seismic models **235**

E Mathematica code for Chapter 5 **245**

List of Figures

1.1 Schematic illustrating the different kinds of physical model used to explain crustal deformation 3
1.2 Time scales of observation for crustal deformation 4
1.3 Screw dislocation and method of images 7
1.4 Schematic of earthquake cycle phases 8
1.5 Co-seismic elastic dislocation 10
1.6 Post-seismic elastic dislocation 10
1.7 Inter-seismic elastic dislocation 10
1.8 The most common linear viscoelastic materials, *Bell* (2014) 12
1.9 Geometry of a spherical magma chamber 15

2.1 oregano_ve basic mesh unit. 24
2.2 oregano_ve modified subroutines. 26
2.3 oregano_ve parallelisation of solution domain 28
2.4 Post-seismic relaxation for elastic lid over viscoelastic half-space. 33
2.5 Tracked nodes for viscoelastic benchmark on simple shear 37
2.6 oregano_ve depth dependent viscosity benchmarks 38
2.7 Periodic stress/displacement on surface 39
2.8 Soution for elastic half-space. 42
2.9 Surface traction condition: constant shear modulus, variable wavelength, total displacement. 47

2.10 Plot for $w = 8$ of surface traction condition 48
2.11 Total displacement at $w = 8$ for contrasting shear modulus 50
2.12 Fully relaxed solution with contrasting shear modulus 51
2.13 Time constant t_R for constant shear modulus as a function of wavelength over elastic lid thickness, shear stress boundary condition 53
2.14 Time constant t_R for constant shear modulus as a function of wavelength over elastic lid thickness, displacement condition 56
2.15 Elastic layer over viscoelastic half space, imposed surface displacement condition 57
2.16 oregano_ve simulations for imposed surface displacement condition . . . 58
2.17 oregano_ve simulations for imposed surface displacement condition, log normalised 58

3.1 Shear zone model schematic 66
3.2 Constant viscosity analytical solution 69
3.3 Sensitivity of solution to aspect ratio 69
3.4 Velocity fields for exponentially depth dependent viscosity 73
3.5 Velocity profiles at $z = 1$ for exponentially depth dependent viscosity . 74
3.6 Shear zone half-width for DDV and Arrhenius law 75
3.7 Absolute deviation of velocity field for Arrhenius law approximate v. numerical solutions 77
3.8 Velocity fields for Arrhenius law 78
3.9 Log plot of Arrhenius viscosities 79
3.10 Velocity fields for non-linear uniform properties 81
3.11 Shear zone scaling for non-linear uniform properties 82
3.12 Shear-zone width for non-linear rheology with depth dependent viscosity 85
3.13 Temperature fields relative to steady state geothermal gradient 92
3.14 Grid of velocity fields for all mechanisms 94

3.15 Continental shear zone half-widths 96
3.16 Predicted shear zone half widths compared to numerical simulations . . 97
3.17 Viscosity fields for each physical mechanism 99

4.1 Post-seismic model schematic 105
4.2 Viscosity structures for $n = 2, 3, 4, 5$ at $t = 0$ following a co-seismic rupture 108
4.3 Postseismic displacements at 50 Maxwell times (for equivalent linear rheology) 110
4.4 Postseismic displacements at 100 Maxwell times (for equivalent linear rheology) 110
4.5 Post-seismic displacements for $n = 1, 2, 3, 4, 5$, single rupture with free slip boundary conditions 111
4.6 Temporal evolution of post-seismic displacements at $x = 1$ 112
4.7 Temporal evolution of post-seismic displacements at $x = 5$ 112
4.8 Viscosity structures for $n = 2, 3, 4, 5$, single rupture with free slip boundary conditions 113
4.9 Postseismic displacements at 5 Maxwell times (for equivalent linear rheology) 116
4.10 Postseismic displacements at 10 Maxwell times (for equivalent linear rheology) 116
4.11 Post-seismic displacements for $n = 1, 2, 3, 4, 5$, single rupture with driving boundary conditions 117
4.12 Post-seismic displacements for $n = 1, 2, 3, 4, 5$, single rupture with driving boundary conditions, double width calculation 118
4.13 Temporal evolution of post-seismic displacements at $x = 1$ 119
4.14 Temporal evolution of post-seismic displacements at $x = 4$ 119
4.15 Viscosity structures for $n = 2, 3, 4, 5$, single rupture with driving boundary conditions 120

4.16 Post-seismic displacements for $n = 1, 2, 3, 4, 5$, repeated ruptures with free slip boundary conditions 122
4.17 Viscosity structures for $n = 2, 3, 4, 5$, repeated ruptures with free slip boundary conditions 123
4.18 Post-seismic displacements for $n = 2, 3, 4, 5$, repeated ruptures with driving boundary conditions 125
4.19 Viscosity structures for $n = 2, 3, 4, 5$, repeated ruptures with driving boundary conditions 126
4.20 Viscosity variation in the viscoelastic layer 130
4.21 Post-seismic weakening factor 131
4.22 $log_{10}(\eta/\eta_0)$ for combined rheology 132
4.23 Viscosity variation for combined rheology 133

5.1 Map of Santorini 139
5.2 InSAR showing period of slow subsidence 141
5.3 Merged bathymetry and LiDAR grid with InSAR 142
5.4 Viscoelastic model for 100m thick lava flow emplaced in 1866 145
5.5 Viscoelastic model for 100m thick lava flow emplaced in 1866 parameters 146
5.6 Graves' 1848 Bathymetry Map 148
5.7 Digitised version of Graves' 1848 Bathymetry Map 149
5.8 Plot of number of points and spacing used to generate the synthetic model. 150
5.9 Map of the differences in seafloor depth between observed and synthetic bathymetry. 151
5.10 Map of the bathymetry, geology and differences in seafloor depth in the Kameni Islands region. 152
5.11 Viscoelastic Green's functions for displacement at 140 years of a point load 155
5.12 Displacements at 140 years for $\mu = 5$ GPa, $r = 1150$m, $h = 110$m. . . . 156

5.13 Bathymetry map of the Kameni Islands region showing the predicted subsidence and uplift at the present day 157

5.14 Models for temperature and viscosity structure in the vicinity of a magma chamber with radius and depth of 2 and 5 km respectively. (a) Temperature structure. Contours=100° C interval. Surface temperature = 20° C. Bottom temperature ∼ 650° C. (b) Viscosity structure assuming a wet quartz rheology. Contour interval = one order of magnitude. Surface viscosity 10^{35} Pa s. 158

5.15 Depth integrated viscosity 159

5.16 Plot of the maximum subsidence vs. year 160

5.17 Temporal evolution of volume change 162

5.18 Mogi source location 163

5.19 Viscoelastic shell model 164

5.20 Comparison between the viscoelastic relaxation model, joint inversion and cumulative seismic moment release 171

5.21 Trade-off curve for viscoelastic model 173

List of Tables

2.1 Least-squares fit for dislocation in elastic lid over viscoelastic half-space. 33

3.1 Viscosity structures 67

4.1 Best fit parameters from *Yamasaki et al.* (2014) 129

4.2 Parameter sets from combined rheologies to match *Yamasaki et al.* (2014) 131

5.1 Cylindrical load parameters 144

5.2 Pressure functions 169

5.3 Initial guesses for viscoelastic shell model minimisation 172

5.4 Viscoelastic shell best fit parameters 172

Chapter 1

Introduction

Plate tectonics, the theory that the Earth's surface consists of a few rigid fragments that deform only in narrow bands around their edges, has been accepted for 40 years as an accurate description of the behaviour of the oceanic parts of the Earth's surface. It is equally clear that continental regions do not behave like oceanic plates. Deformation in continents is not restricted to narrow bands but is spread over great distances within their interiors. A number of lines of evidence, including the distribution of earthquakes, reveal that the strength of different continental regions varies markedly. While it is relatively easy to qualitatively map out these variations, little progress has been made in quantifying the range of strength in the continents and identifying the physical mechanisms that control these variations. I investigate crustal deformation associated with the earthquake cycle, inflation of magma chambers beneath volcanoes, and changes in surface loads. Results of these models have important implications for our understanding of large-scale continental deformation and mountain building, in addition to both seismic and volcanic hazard assessment.

The Earth is unique among the terrestrial bodies of our solar system, not just because it supports life, but also because it appears to be the only one to experience plate tectonics with spreading ridges and subduction zones. This is unlikely to be coincidental - plate tectonics drives the recycling and renewal of surface water vital for

life. Tectonic boundary regions are also lines of weakness, favouring the deposition of economically important minerals. Even the influence of climate change can be observed in the motions of the Earth's surface, with deglaciation driving continental uplift and sea level change. The fundamental challenge of tectonics is to understand the rules governing the spatial and temporal distribution of crustal deformation and to explain the unique status of plate tectonics on Earth.

Crustal deformation timescales range from fractions of a second during an earthquake, to major mountain building events over tens of millions of years. Equally, length scales range from the diameter of a single crystal (μm-cm), to the length of faults (thousands of km). Laboratory experimental methods can only sample a small fraction of this range of scales (μm-cm and seconds-hours) *Bürgmann and Dresen* (2008), though their results must necessarily inform development of my models. To study deformation on geological spatial scales, we may examine problems where the crust is subjected to known forces. The Earth itself becomes the laboratory where surface deformation driven by processes such as earthquakes, volcanoes, mountain building, and deglaciation provide a window into the rheology (relationship between stress and strain) of the crust and upper mantle. Results of existing models vary with the timescale of observation and a spatial bias arises due to the lack of fault zones or volcanoes in stable areas of continental crust. Typically, these problems have been modelled independently as elastic (a solid capable of returning to its original shape after deformation), viscous (able to flow as a liquid with resistance), viscoelastic (having a combination of both properties), or inviscid fluids (flowing without resistance). Depending on the process studied, the parameters obtained and the physical model used varies widely (Figure 1.1).

Over the past two decades, extensive measurements of continental deformation occurring in different stages of the earthquake cycle and volcanic systems have become possible. These include GPS measurements over networks surrounding active faults

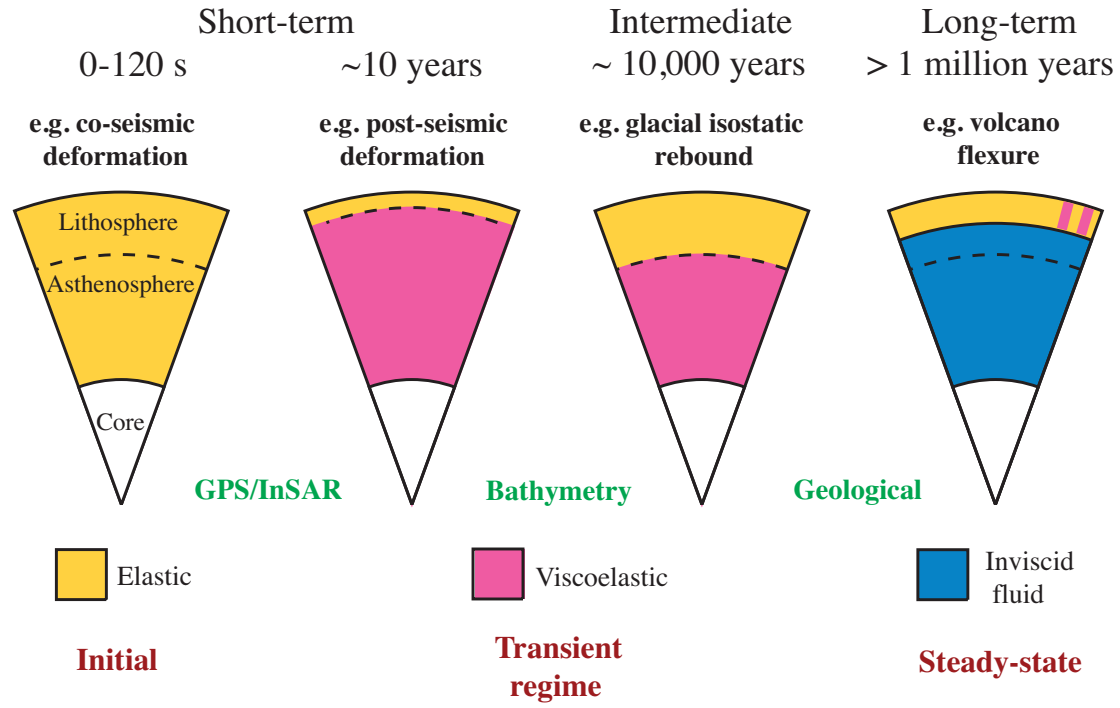


Figure 1.1: Schematic illustrating the different kinds of physical model used to explain the labelled processes on short-term, intermediate, and long-term timescales. Note that viscoelastic processes include a wide range of mechanisms with both viscous and elastic properties. Light blues text indicates the most commonly used rheological model at each timescale, dark red text indicates the nature of the response, and dark green text indicates available observations. Figure modified from *Watts et al.* (2013).

and volcanoes, and satellite radar observations (InSAR) which can be used to monitor deformation rates down to millimetres per year. These techniques can also be applied to deformation driven by climate change and human intervention. Recently rediscovered Royal Navy charts (*Watts et al.* (2015)) and modern swath bathymetry allow investigation of oceanic crustal deformation on novel timescales. Figure 1.2 illustrates the range of timescales that available observations span.

Other important questions I intent to address in this Thesis include the distribution of shear at depth? We observer faults as quasi-planar structures at the Earth's surface but geologic observations of exhumed shear zones (*Poirier* (1980), *White et al.* (1980), *Rutter* (1999), and *Norris and Cooper* (2003)) indicate that whilst the motion

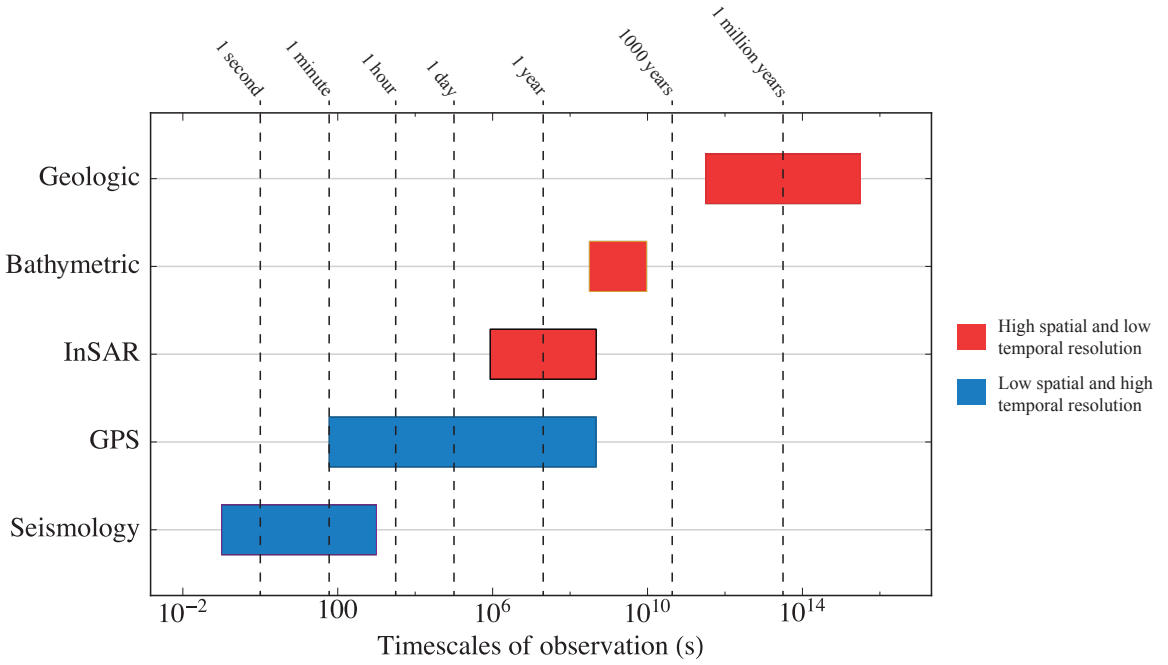


Figure 1.2: Time scales of observation for crustal deformation, from the passage of seismic waves through the crust over seconds to geologic observations. Historical bathymetric charts have only been recently rediscovered (*Watts et al.* (2015)) and cover a novel timescale of observation.

is no longer planar, it can still be localised near the down-dip extension of the fault; however, the degree of localisation is still uncertain. The degree of localisation of shear at depth and the effective viscosity structure beneath the fault has important implications for the interpretation of geodetic observations of surface deformation throughout the earthquake cycle. If shear is localised at depth, then the region of applicability for elastic half-space models of fault creep at depth, *Savage and Burford* (1973), is greater than previously thought. A localised shear zone would allow inter-seismic strain accumulation resembling the well-known arctangent distribution of surface displacements. The results we present on the localisation of shear at depth complements previous work by *Hetland and Hager* (2005, 2006), *Barbot et al.* (2008) and others, where they develop analytical solutions for viscoelastic earthquake cycle models, by allowing for the development of a weak zone beneath a fault.

This thesis will primarily take a theoretical approach towards continental deforma-

tion, examining the different physical mechanisms that interact in the continents to produce zones of distributed shear over millions of years and post-seismic transients in days to years. Additionally, deformation due to crustal loading from volcanic eruptions will be explored.

1.1 The Earthquake cycle

The elastic rebound hypothesis proposed by *Reid* (1910) was derived from a comparison between triangulation surveys in the San Fransisco Bay area taken over the period 1851-1892 and in 1907 following the San Fransisco Earthquake of 1906. The hypothesis envisions the accumulation of elastic strain driven by the relative motions of neighbouring portions of the Earth's crust. Once the stress which caused this strain exceeds the friction on the fault, the fault is able to slip (an earthquake) releasing the stored elastic potential energy by allowing the strained rock to 'catch up' with the far field displacements, and return to the start of the process. This forms the basis of our current understanding of what has come to be known as the earthquake cycle. The modern concept is generally considered to consist of three main phases: an inter-seismic strain accumulation stage, a co-seismic rupture event, and post-seismic transient deformation. During the post-seismic phase, deformation rates are elevated for a period ranging from days to years after the rupture before returning to the background inter-seismic rate. The current model also takes into account the variation in rheological parameters with depth. It is now believed that the top 15km (for continental crust) behaves more or less like an elastic solid (as in the elastic rebound hypothesis *Reid* (1910)) with the material beneath exhibiting viscoelastic behaviour. Viscoelasticity is therefore frequently used to explain the post-seismic transients with their associated time scales.

1.1.1 Elastic dislocation theory

As I will be focussing on transcurrent, or strike-slip, deformation in this thesis I shall begin by considering the application of elastic dislocation theory to the simple model of an infinitely long strike slip fault, where slip is uniform with depth. For this case of anti-plane strain (where we use ϵ for strain), there are only two non-zero strain components (x_1 is horizontal and normal to the fault plane, x_2 is vertical, and u_3 is displacement in and out of the plane)

$$\epsilon_{13} = \frac{1}{2}\frac{\partial u_3}{\partial x_1} \text{ and } \epsilon_{23} = \frac{1}{2}\frac{\partial u_3}{\partial x_2}. \tag{1.1}$$

Applying Hooke's law for a homogeneous, isotropic elastic material (with stress σ, shear modulus μ, and lame parameter λ),

$$\sigma_{ij} = 2\mu\epsilon_{ij} + \lambda\epsilon_{kk}\delta_{ij}, \tag{1.2}$$

and solving the force balance equation (neglecting inertial terms)

$$\partial\sigma_{ij}/\partial x_j = 0 \tag{1.3}$$

is equivalent to solving Laplace's equation for the displacement field

$$\nabla^2 u_3(x_1, x_2) = 0. \tag{1.4}$$

The solution of Laplace's equation in a full space for a dislocation line with fixed offset s, also known as a screw dislocation (*Segall* (2010)), is shown in Figure 1.3(a) and has

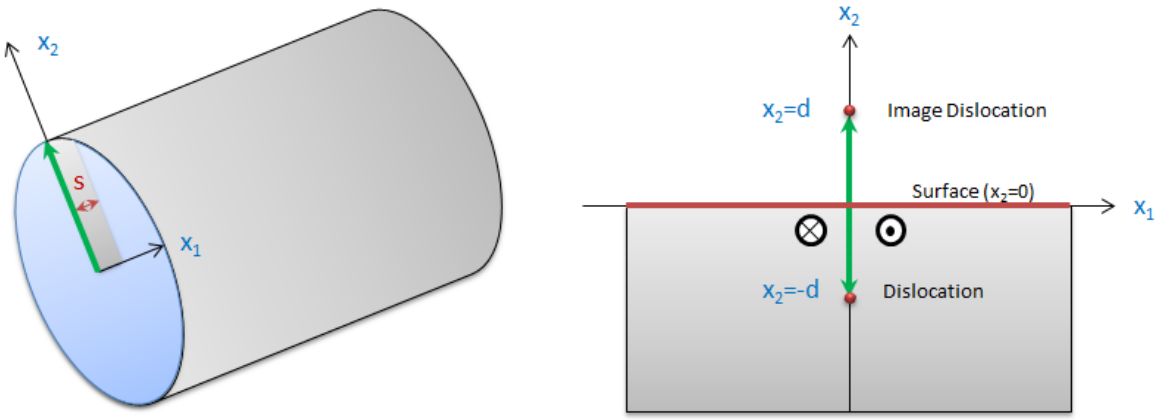


Figure 1.3: (a) Screw dislocation along x_3 with fixed offset s, where x_3 is parallel to s. (b) Dislocation at $x_2 = -d$ and image dislocation at $x_2 = d$, where x_3 is perpendicular to the page.

displacement u_3 and stresses, σ,

$$u_3 = -\frac{s}{2\pi} \arctan\left(\frac{x_1}{x_2}\right), \tag{1.5a}$$

$$\sigma_{13} = \mu \frac{\partial u_3}{\partial x_1} = -\frac{s\mu}{2\pi} \frac{x_2}{x_1^2 + x_2^2}, \tag{1.5b}$$

$$\sigma_{23} = \mu \frac{\partial u_3}{\partial x_2} = -\frac{s\mu}{2\pi} \frac{x_1}{x_1^2 + x_2^2}. \tag{1.5c}$$

Shifting the dislocation pole from the origin to a depth $-d$ and employing the method of images, where an oppositely signed dislocation is included at $+d$, Figure 1.3(b), produces a solution where the shear tractions are zero at the $x = 0$ surface

$$u_3 = -\frac{s}{2\pi} \left[\arctan\left(\frac{x_1}{x_2 + d}\right) - \arctan\left(\frac{x_1}{x_2 - d}\right)\right]. \tag{1.6}$$

Physically this solution would correspond to uniform slip on the interval $-d$ to d. The earthquake cycle can then be divided up into three phases corresponding to slip at different depths, illustrated in Figure 1.4. Equation (1.6) would then give the coseismic displacements illustrated in Figure 1.5. Including the contribution from a second dislocation (and it's image) allows us to construct a solution for uniform slip between

two depths (i.e. a fault which doesn't reach the surface)

$$u_3 = -\frac{s}{2\pi}\left[\arctan\left(\frac{x_1}{x_2+d_1}\right) - \arctan\left(\frac{x_1}{x_2-d_1}\right) - \arctan\left(\frac{x_1}{x_2+d_2}\right) + \arctan\left(\frac{x_1}{x_2-d_2}\right)\right]. \tag{1.7}$$

Slip between two depths d_1 and d_2 could then be interpreted as one explanation for the post-seismic phase of the earthquake cycle, where the stress change introduced by the coseismic displacements drives slip over a finite interval at depth, as shown in Figure 1.6.

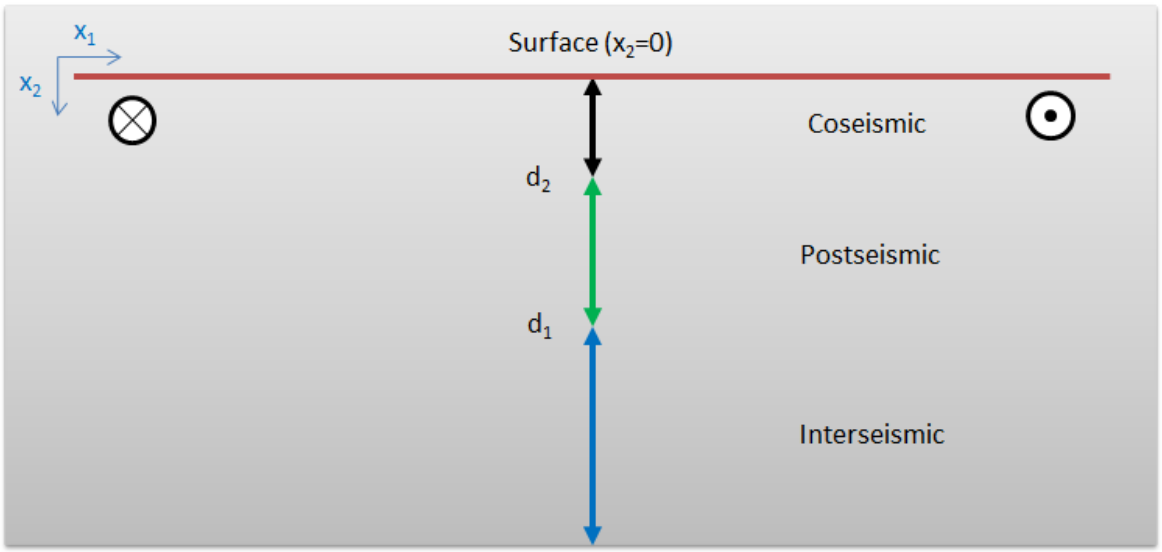


Figure 1.4: Earthquake cycle phases, displacements and shear strains on the surface $x_2 = 0$ given by equation (1.8). During the inter-seismic phase, slip occurs below the depth $x_2 = d_1$, the co-seismic rupture occurs between $d_2 < x_2 < 0$ and the cycle is completed by slip at intermediate depth between $d_1 < x_2 < d_2$.

If we take the appropriate limits to equation (1.7) we can calculate surface displacements (equation (1.8)) for coseismic ($d_2 = 0$, Figure 1.5), postseismic (Figure 1.6), and interseismic ($d_1 \to \infty$, Figure 1.7) phases. Schematically, these contributions

are represented by Figure 1.4.

$$\text{Interseismic displacements:} \quad u_3(x_2 = 0) = \frac{s}{\pi} \arctan\left(\frac{x_1}{d_1}\right) \tag{1.8a}$$

$$\text{Interseismic shear strain:} \quad \epsilon_{13} = \frac{s}{2\pi d_1}\left[1 + \left(\frac{x_1}{d_1}\right)^2\right]^{-1} \tag{1.8b}$$

$$\text{Coseismic displacements:} \quad u_3(x_2 = 0) = \frac{s}{\pi} \arctan\left(\frac{d_2}{x_1}\right) \tag{1.8c}$$

$$\text{Coseismic shear strain:} \quad \epsilon_{13} = \frac{-s}{2\pi d_2}\left[1 + \left(\frac{x_1}{d_2}\right)^2\right]^{-1} \tag{1.8d}$$

$$\text{Postseismic displacements:} \quad u_3(x_2 = 0) = \frac{-s}{\pi}\left[\arctan\left(\frac{x_1}{d_1}\right) - \arctan\left(\frac{x_1}{d_2}\right)\right] \tag{1.8e}$$

$$\text{Postseismic shear strain:} \quad \epsilon_{13} = \frac{s}{2\pi}\left[\frac{d_2}{d_2^2 + x_1^2} - \frac{d_1}{d_1^2 + x_1^2}\right] \tag{1.8f}$$

Summing the contributions to the surface displacements for all three phases at the surface (equations (1.8)) produces a step function

$$\frac{s}{2}\text{sign}(x_1), \tag{1.9}$$

and thus recovers the expected long term block like motion. Equations (1.8) are plotted in Figures 1.5, 1.6 and 1.7 parts (b).

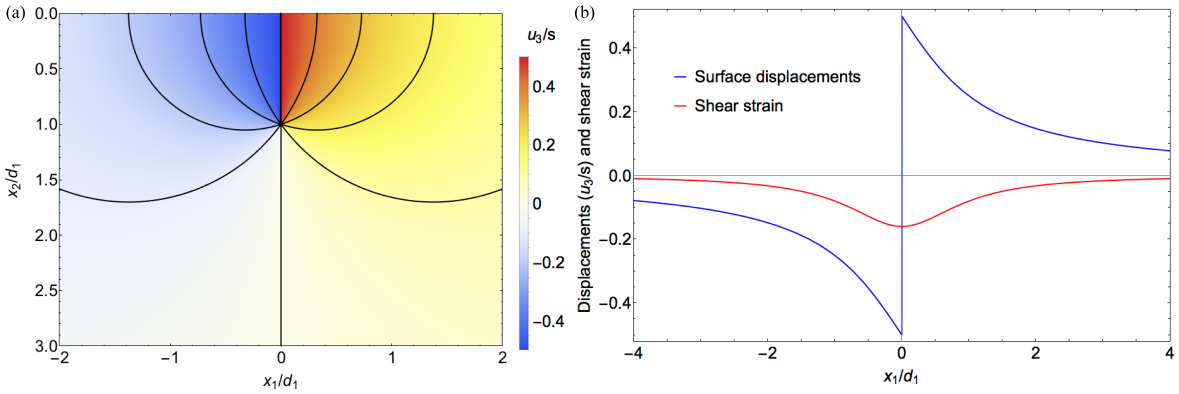


Figure 1.5: Co-seismic phase on an infinitely long strike slip fault with fault slip to depth 3km and displacement non-dimensionalised as u_3/s: (a) Displacements, (b) Surface displacements and strain rate.

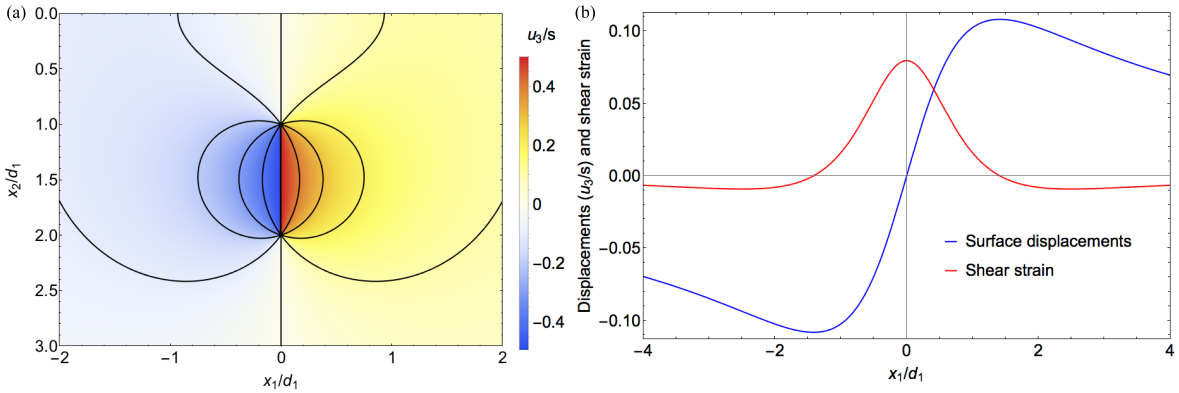


Figure 1.6: Post-seismic phase on an infinitely long strike slip fault with fault slipping from 3km to 6km and displacement non-dimensionalised as u_3/s: (a) Displacements, (b) Surface displacements and strain rate.

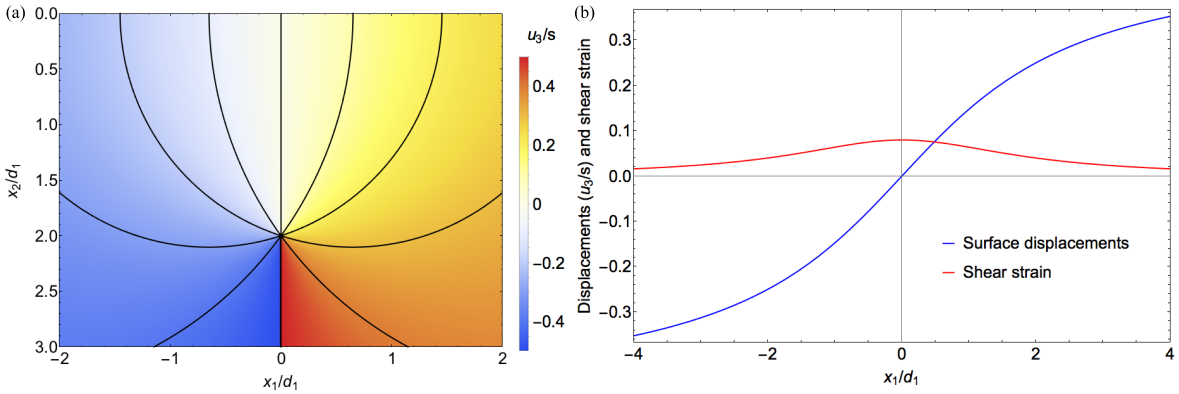


Figure 1.7: Inter-seismic phase on an infinitely long strike slip fault with fault locked to depth 6km and displacement non-dimensionalised as u_3/s: (a) Displacements, (b) Surface displacements and strain rate.

1.2 Viscoelasticity

Observations of crustal deformation have revealed that elasticity is only one part of a complex interlinked system of physical processes governing the redistribution of stress. An elastic medium is capable of returning to its original state after deformation, whilst a viscous media is able to flow as a fluid and does not return to its original state. Geodetic measurements and laboratory experiments have shown that rocks deform both elastically and viscously on different timescales and under different stress regimes. A medium which has both these properties is viscoelastic, though the interaction of elastic and viscous components of strain are not necessarily straightforward. In the cases I am interested in the deformation is governed by the force balance equation (1.3), where I neglect body forces and inertial terms, and a relation between stress, strain and rates thereof, known as a constitutive relation, appropriate to the media in question.

1.2.1 Constitutive laws

Whilst there are a suite of constitutive equations, there are two end member cases; elastic,

$$\sigma = 2\mu\epsilon, \tag{1.10}$$

and viscous,

$$\sigma = 2\eta\dot{\epsilon}, \tag{1.11}$$

which may be combined in several ways to produce a viscoelastic media, illustrated in Figure 1.8.

The linear viscoelastic material most often used to model crustal processes is the Maxwell material. A Maxwell material behaves as an elastic solid over short time scales and as a viscous fluid at long time scales. This is constructed by assuming the total

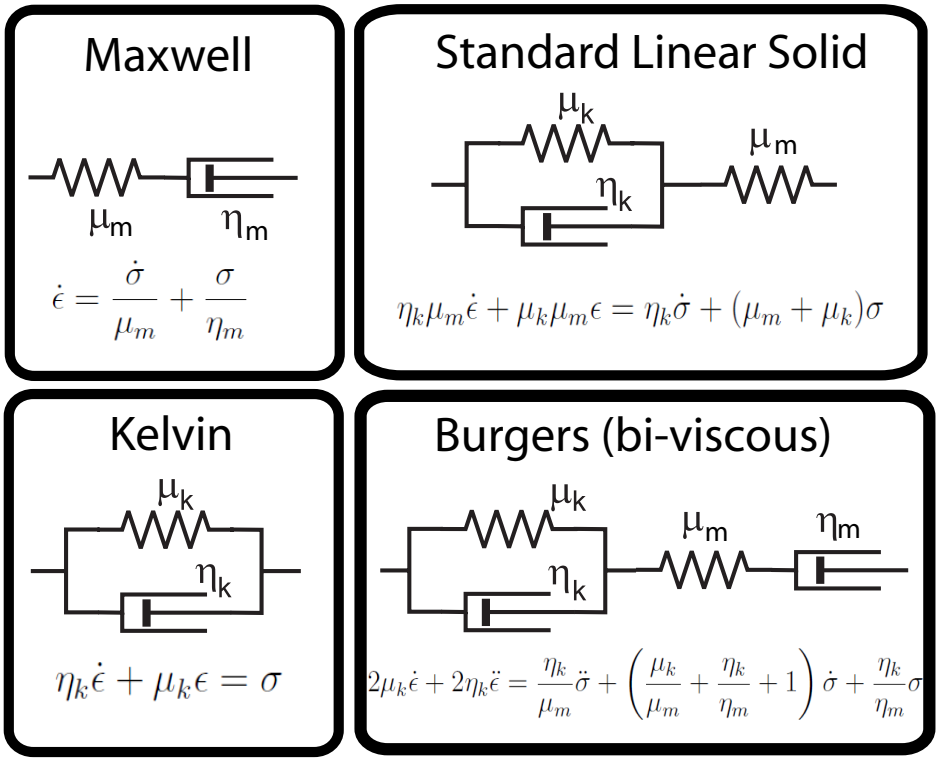


Figure 1.8: The most common linear viscoelastic materials, *Bell* (2014)

strain rate is the sum of the viscous strain rates and elastic strain rates,

$$\dot{\epsilon}_{ve} = \dot{\epsilon}_e + \dot{\epsilon}_v, \tag{1.12a}$$

$$= \frac{\dot{\sigma}}{2\mu} + \frac{\sigma}{2\eta}, \tag{1.12b}$$

which in tensor form is

$$\dot{\sigma}_{ij} + \frac{\mu}{\eta}\left(\sigma_{ij} - \frac{\sigma_{kk}}{3}\delta_{ij}\right) = 2\mu\dot{\epsilon}_{ij} + \lambda\dot{\epsilon}_{kk}\delta_{ij}. \tag{1.13}$$

I will primarily be dealing with a Maxwell material in this thesis for the linear rheologies, though it is important to note that the Burgers rheology (Figure 1.8) is sometimes invoked to explain post-seismic transients with multiple timescales of relaxation (*Pollitz* (2005)). Other explanations for observed transient deformation include a Maxwell rheology with depth dependent viscosity (e.g. *Yamasaki and Houseman*

(2012a)) and non-linear rheologies (e.g. *Takeuchi and Fialko* (2013)). The first increase in complexity is a depth dependent Maxwell rheology, which obeys the constitutive equation (1.13) with $\eta(x, y, z)$ in the most general case. Further to this, I will investigate the effect of non-linear rheologies where $\dot{\epsilon} = A\sigma^n$, which may be modelled as an extension to the Maxwell model with an effective viscosity

$$\eta_{nl} = \frac{1}{B} J_2^{\frac{1}{n}-1}, \tag{1.14}$$

where J_2 is the second invariant of the strain-rate tensor,

$$J_2 = 2^{-\frac{1}{2}} \sqrt{\dot{\epsilon}_{ij}\dot{\epsilon}_{ij}}. \tag{1.15}$$

1.2.2 The correspondence principle

If we look at the constitutive equation for a Maxwell viscoelastic rheology, Equation (1.12b) can be rewritten as

$$\frac{\partial}{\partial t}\epsilon = \frac{1}{2}\left(\frac{1}{\mu}\frac{\partial}{\partial t} + \frac{1}{\eta}\right)\sigma. \tag{1.16}$$

If we then take the Laplace transform,

$$\mathcal{L}\{f(t)\} = \bar{f}(s) = \int_0^\infty f(t)e^{-st}\,dt, \tag{1.17}$$

with standard result

$$\mathcal{L}\left\{\frac{\partial f(t)}{\partial t}\right\} = s\bar{f}(s) \tag{1.18}$$

equation 1.16 is of the form

$$\bar{\sigma} = 2\bar{\mu}\bar{\epsilon}, \tag{1.19}$$

where

$$\bar{\mu}(s) = \frac{s}{\mu^{-1}s + \eta^{-1}}. \tag{1.20}$$

Equation 1.19 resembles the constitutive relation for an elastic media; if we know the solution to an elastic problem, we can therefore calculate the solution to the corresponding Maxwell viscoelastic solution by substituting $\mu \to \bar{\mu}(s)$ in equation 1.20 and performing the inverse Laplace transform

$$f(t) = \frac{1}{2\pi i} \int_{\gamma - i\infty}^{\gamma + i\infty} \bar{f}(s) e^{st}\, ds. \tag{1.21}$$

The correspondence principle assumes that η is constant, so cannot be used in its current form for depth dependent problems, though it can be used for layered rheologies. Unfortunately, viscosity is dependent on temperature and this restricts the usefulness of the correspondence principle for whole crustal models; though approximations can be made using layered rheologies.

1.3 Deformation of a spherical magma chamber

Another source of crustal deformation is that due to magmatic plumbing systems, the simplest case of which is elastic deformation driven by a pressure change in a spherical magma chamber of radius a at a depth d in a uniform elastic half-space, illustrated in Figure 1.9. This was first derived by *Mogi* (1958) and is reproduced from *Segall* (2010). I require both spherical coordinates (r, θ, ϕ), centred on the magma chamber, and cylindrical coordinates (ρ, θ, z), centred a distance d above the chamber on the free surface. The boundary conditions include traction free on the $z = 0$ surface, constant pressure p within the chamber and negligible shear stresses on the chamber,

$$\sigma_{iz} = 0 \qquad \text{on} \qquad z = 0, \tag{1.22a}$$

$$\sigma_{rr} = -p \qquad \text{on} \qquad r = a, \tag{1.22b}$$

$$\sigma_{r\theta} = \sigma_{r\phi} = 0 \qquad \text{on} \qquad r = a. \tag{1.22c}$$

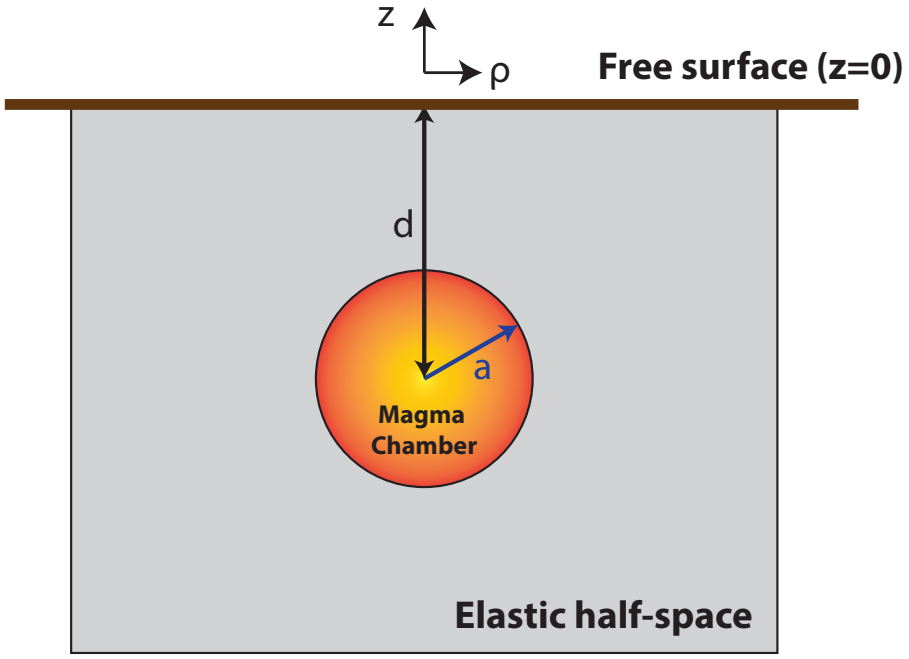


Figure 1.9: A spherical magma chamber with radius a in an elastic half-space at a depth d beneath the free surface.

We begin by calculating the solution to a pressurised sphere in a full-space, ignoring boundary condition (1.22a). In this case the force balance equation (1.3) reduces to

$$\frac{\partial \sigma_{rr}}{\partial r} + \frac{1}{r}(2\sigma_{rr} - \sigma_{\theta\theta} - \sigma_{\phi\phi}) = 0, \tag{1.23}$$

which may be combined with Hooke's law (equation (1.2)), radial strain $\epsilon_{rr} = \partial u_r / \partial r$, and the hoop stresses $\sigma_{\theta\theta} = \sigma_{\phi\phi} = u_r / r$, to give

$$\frac{d}{dr}\left(\frac{du_r}{dr} + 2\frac{u_r}{r}\right) = 0. \tag{1.24}$$

This may be integrated once, then pre-multiplied by a factor $e^{\Phi(r)}$,

$$e^{\Phi(r)}\frac{du_r}{dr} + \frac{2u_r}{r}e^{\Phi(r)} = Ae^{\Phi(r)}, \tag{1.25}$$

and with the substitution $y = u_r e^{\Phi(r)}$, we find

$$\frac{dy}{dr} + \left(\frac{2}{r} - \frac{d\Phi}{dr}\right) u_r e^{\Phi(r)} = A e^{\Phi(r)}. \tag{1.26}$$

As Φ is an arbitrary function, if we require that it satisfies

$$\frac{2}{r} - \frac{d\Phi}{dr} = 0 \tag{1.27}$$

then the solutions

$$\Phi = 2\log(r), \tag{1.28a}$$

and

$$u_r = A\frac{r}{3} + \frac{B}{r^2}, \tag{1.28b}$$

are obtained. Requiring the displacements to decay away from the chamber provides $A = 0$ and the boundary conditions 1.22b give

$$\sigma_{rr} = 2\mu\epsilon_{rr} = \frac{-4\mu B}{r^3}, \tag{1.29}$$

such that $B = pa^3/4\mu$. The displacements and stresses are therefore given by

$$u_r = \frac{pa^3}{4\mu r^2}, \tag{1.30a}$$

$$\sigma_{rr} = \frac{-pa^3}{r^3}, \tag{1.30b}$$

$$\sigma_{\theta\theta} = \sigma_{\phi\phi} = 2\mu\frac{u_r}{r} = \frac{pa^3}{2r^3}. \tag{1.30c}$$

This solution violates the surface boundary condition (1.22a), but we may follow the method of *McTigue* (1987) and iteratively apply corrections to cancel out the surface stresses and the boundary conditions on the chamber, equation (1.22b). Evaluating equation (1.30a) on the free surface in a cylindrical coordinate system gives radial and

vertical displacements

$$u_z(\rho, z=0) = \frac{pa^3}{4\mu}\frac{d}{(\rho^2+d^2)^3/2}, \tag{1.31a}$$

$$u_\rho(\rho, z=0) = \frac{pa^3}{4\mu}\frac{\rho}{(\rho^2+d^2)^3/2}. \tag{1.31b}$$

The stresses transform as

$$\sigma_{\rho z} = (\sigma_{rr} - \sigma_{\theta\theta})\sin(\theta)\cos(\theta), \tag{1.32a}$$

$$\sigma_{zz} = \sigma_{rr}\cos^2(\theta) + \sigma_{\theta\theta}\sin^2(\theta), \tag{1.32b}$$

giving

$$\sigma_{\rho z} = -\frac{pa^3}{2}\left(\frac{3d^2}{(\rho^2+d^2)^5/2} - \frac{1}{(\rho^2+d^2)^{3/2}}\right), \tag{1.33a}$$

$$\sigma_{zz} = -\frac{3pa^3}{2}\frac{\rho d}{(\rho^2+d^2)^{5/2}}. \tag{1.33b}$$

If we apply equal and opposite surface tractions on $z = 0$ we find additional displacements (with ν Poisson's ratio),

$$u_z(\rho, z=0) = \frac{(3-4\nu)pa^3}{4\mu}\frac{d}{(\rho^2+d^2)^{3/2}}, \tag{1.34a}$$

$$u_\rho(\rho, z=0) = \frac{(3-4\nu)pa^3}{4\mu}\frac{\rho}{(\rho^2+d^2)^{3/2}}, \tag{1.34b}$$

which are added to equations (1.31) to produce the total surface displacement

$$u_z(\rho, z=0) = \frac{(1-\nu)pa^3}{\mu}\frac{d}{(\rho^2+d^2)^{3/2}}, \tag{1.35a}$$

$$u_\rho(\rho, z=0) = \frac{(1-\nu)pa^3}{\mu}\frac{\rho}{(\rho^2+d^2)^{3/2}}. \tag{1.35b}$$

Whilst this solution now satisfies the boundary condition (1.22a), it violates (1.22b) on the chamber walls by order $(a/d)^3$. As d is in general much greater than a, this source of error is often neglected for real problems. Note that the solution is unable to resolve

p and a independently as they appear only together as pa^3, thus at this level of approximation it is not possible to estimate the size of the chamber and pressure change independently. A higher order approximation, presented in *McTigue* (1987) theoretically allows us to separate pressure and chamber radius; however, the corrections are of order $(a/d)^3$ and are not generally resolvable using current geodetic methods. Instead, we can rewrite equations (1.35) in terms of a volume change (ΔV),

$$u_z(\rho, z=0) = \frac{(1-\nu)\Delta V}{\pi} \frac{d}{(\rho^2+d^2)^{3/2}}, \tag{1.36a}$$

$$u_\rho(\rho, z=0) = \frac{(1-\nu)\Delta V}{\pi} \frac{\rho}{(\rho^2+d^2)^{3/2}}. \tag{1.36b}$$

Geodetic measurements of radial and vertical displacements of volcanic unrest may be modelled by equations (1.36) to infer source location and volume change. I.E. The uplift is independent of the magma chamber radius for a given influx of magma ΔV.

1.4 Overview of thesis

This thesis is concerned with the application of rheological models to explain crustal deformation, utilising approaches combining Maxwell viscoelasticity, depth dependent viscosity, and non-linear stress strain-rate constitutive relationships.

Chapter 2 outlines modifications made to the 3D finite element program *oregano_ve* to include periodic boundary conditions to simulate anti-plane strain conditions. This will allow me to build upon simple analytical models, both from the literature and my own derivations, using numerical calculations to investigate problems that cannot be solved analytically. In the process of modifying the code, I develop novel benchmarks for simple shear of a viscoelastic block with depth dependent viscosity and a non-linear rheology. Finally, Chapter 2 develops a model applicable to a Fourier decomposition of an imposed stress field, and the deformation and relaxation timescale associated with each mode.

Chapter 3 provides a detailed analysis of viscous shear zones, where the earthquake cycle is time averaged to a block model in an elastic lid driving viscous flow at depth. I draw comparisons between the different physical mechanisms: depth dependent viscosity, non-linear rheologies, and thermomechanical coupling. There I find that the key control on the scaling of the width of the shear zone is the depth dependence of viscosity, with secondary contributions from a non-linear rheology and thermomechanical coupling. Whilst thermomechanical coupling is a relatively minor control on the velocity field, it provides a strong control on the viscosity structure beneath the fault.

Chapter 4 investigates the post-seismic phase of the earthquake cycle, primarily concerned with apparent localised weakening in the immediate post-seismic deformation field. I concentrate on the effective viscosity reduction generated by a power-law stress strain-rate relationship, and use *oregano_ve* with periodic boundary conditions to simulate an infinitely long strike-slip fault. To ensure deformation commensurate with geodetic observations, I must also include a far field loading condition. Finally, implementing repeated ruptures on a fault with far field loading reproduces the viscosity structures seen in Chapter 3.

Chapter 5 covers a detailed analysis of deformation at Santorini volcano, Greece, between 1848 and 2012. It includes historic bathymetric charts and modern swath bathymetry showing up to 100m of subsidence between 1848 and 2006 which I model as viscoelastic relaxation of the substratum. It also includes InSAR measurements between 1993 and 2010 revealing a slow subsidence signal located at the centre of the displacements driven by the load identified from the historic charts. These two data sets allow me to find a best fitting elastic lid thicknesses and viscosity of the lower crust. Finally, I investigate a period of unrest between 2011 and 2012, which I model as two separate pulses of material being supplied to a shallow magma chamber, with viscoelastic relaxation of a thermal aureole surrounding the chamber driving continuing

deformation following cessation of activity.

Chapter 2

3D finite element viscoelastic deformation in *oregano_ve* and viscoelastic relaxation of imposed harmonic tractions and displacements

2.1 Introduction to *oregano_ve*

Oregano_ve is a finite deformation program which computes the distribution of strain rates and displacements in a 3D non-Newtonian viscoelastic material within a volume subjected to arbitrary surface tractions and displacement rates, and a body force caused by gravity acting on a variable distribution of density. The solution region is spanned by a finite element mesh of tetrahedral elements on which displacement rates and pressures are represented by nodal values. In this version of the program, the solution domain is fixed in time and displacement fields are specified with respect to this fixed reference frame. In the viscous only version of the program (*basil*), the mesh is advected at every time step to obtain the Lagrangian-frame finite deformation field.

The program is primarily written in Fortran with elements of C code being used to link to the Open MPI (Message Passing Interface) library, allowing for parallelisation of the problem. Conceptually, the solution domain is sliced into layers perpendicular to the z-direction, thus to allow the most efficient parallelisation the z-direction should be chosen as one of the long dimensions of the solution domain, and the gravity vector may be defined to act in any of the x, y, or z directions.

The solution domain can be subject to a combination of displacement (Dirichlet) and traction (Neumann) boundary conditions, corresponding to enforced displacements or surface tractions. My contribution to the project is to extend the boundary conditions that can be handled to include periodic boundary conditions, so I can numerically reproduce anti-plane problems. I have only implemented this in the z-direction for two reasons. Firstly, the modifications required to introduce periodic boundary conditions in the x and y directions differ significantly to that of the z direction due to the manner in which the code has been parallelised. Periodicity in x and y requires significant modification to the matrix assembly routines in the program, whilst periodicity in the z-direction was achieved by modifying the parallelisation routines. Secondly, I only require periodic boundary conditions in one direction to numerically approximate an infinitely long strike slip fault; periodicity in the other boundaries would allow investigations of fault networks, which is beyond the scope of this thesis.

The program is organised into 5 major sections:

- Load input file - READONS in vsbcons.f

  Initially, *oregano_ve* takes an input file containing information concerning the geometry and specifics of the problem to be calculated. This file specifies dimensions for the solution domain (a rectangular box), the number of nodes for the finite element mesh, boundary conditions, material regions, faults and time series

data. This section has been modified to recognise 'PERIODIC ' as a command line flag, with the expectation of X, Y or Z to follow. Due to the construction of the mesh, it is recommended that the number of mesh nodes along the z-direction, NZ, is an integer multiple of double the number of MPI processes (*processes*).
$NZ = 2 \times N \times processes$, $N \in \mathbb{Z}$

- Mesh Generation

  The finite element mesh is constructed using tetrahedrons defined by 4 vertex nodes at the corners, with an additional 6 nodes located on the midpoints of the edges of the tetrahedron. The velocity field is interpolated quadratically between the nodes, whereas the pressure field is only linearly interpolated (this is of less importance for the anti-plane problems I will be investigating, where the pressure field should be zero everywhere). The meshing routine begins by defining a triangle, projecting it into a 3D pentahedron and then subdividing it to form three tetrahedra with additional midpoint nodes (Figure 2.1). This means that there are two basic units constituting the mesh, corresponding to a reflection of the pentahedral unit of three tetrahedra on one of the side faces. To tessellate the volume coherently, with matching vertex nodes on the bounding faces of adjacent units, the whole volume should be divided up into pairs of these units. This is also true when matching boundaries between MPI processes.

- Matrix Assembly

  The governing equations to be solved are the force balance equation in a viscous medium (neglecting inertia),

$$\partial \sigma_{ij} / \partial x_j = -\rho g_i, \tag{2.1}$$

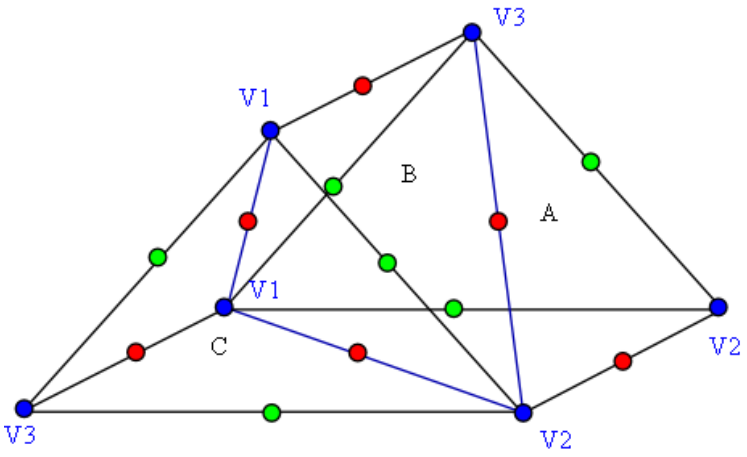


Figure 2.1: Basic mesh unit. Original vertices in black with blue vertex nodes (Vx) and midpoint nodes in green, additional vertices in blue with midpoint nodes in red. Individual tetrahedra labelled A to C.

a Maxwell constitutive relation,

$$\dot{\sigma}_{ij} + \frac{\mu}{\eta}\left(\sigma_{ij} - \frac{\sigma_{kk}}{3}\delta_{ij}\right) = 2\mu\dot{\epsilon}_{ij} + \lambda\dot{\epsilon}_{ij}\delta_{ij}, \tag{2.2}$$

and the conservation of mass,

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{2.3}$$

These equations are discretised into a set of linear equations for the nodal values using the Galerkin method, in which force balance and conservation of mass are enforced with respect to a weighted volume integral. This results in a matrix equation ($\mathbf{Ax} = \mathbf{b}$) for the solution field where the symmetric stiffness matrix $\mathbf{A}$ contains entries corresponding to the volume integrals of the quadratic interpolating functions. The integration is done using Gaussian quadrature with 11 sample points, including the centroid of the mesh tetrahedron and an interior point close to each of the nodes.

- Matrix Solution

The matrix equation is preconditioned in the routine PRECON3 using a diagonal

preconditioner and then solved using the method of conjugate gradients, in the routine CONJGR3. It is in these routines that many of the Message Passing Interface (MPI) calls are located, to enable the solutions to be matched at the boundaries of the computational slices.

- Time-stepping

  oregano_ve is discretised in time using finite difference methods with adaptive time-stepping. There are implementations of both explicit and implicit solvers.

2.2 Modifications

My modifications to *oregano_ve* are motivated by two problems: an infinitely long strike slip fault and an infinite number of parallel faults. Numerically approximating an infinitely-long strike slip fault would allow me to calculate solutions to problems where analytical solutions for simple structures can be derived, such that these analytical results can inform my interpretation of numerical calculations for more complex rheologies. The aim of my modification was to include periodic boundary conditions; if the properties in the direction of the periodic boundary were constant along the periodic axis this would allow users of *oregano_ve* to treat one dimension as infinite, sometimes described as a 2.5D problem. If the properties in the direction of the periodic boundary were not constant, then it would constitute a repeating set of properties.

2.2.1 Code Modifications

There were three major components required to complete the modifications to allow for periodicity in the z-direction.

- Modify the routines with MPI commands (CONJGR3, PRECON3, ADDMATVEC3 and ADDPROD) to implement periodicity in the z-direction.

- Pass a flag (PBC) down from the main routine (OREGANO) to allow for periodicity to be optional.
- Modify the input routine (READONS) to recognise 'PERIODIC Z' at the command line and set the flag appropriately.
- Modify the output routine (SAVE_ARRAY) to save the periodic boundary condition flag.

The flow of the periodic boundary condition flag is shown in Figure 2.2, along with all the modified subroutines.

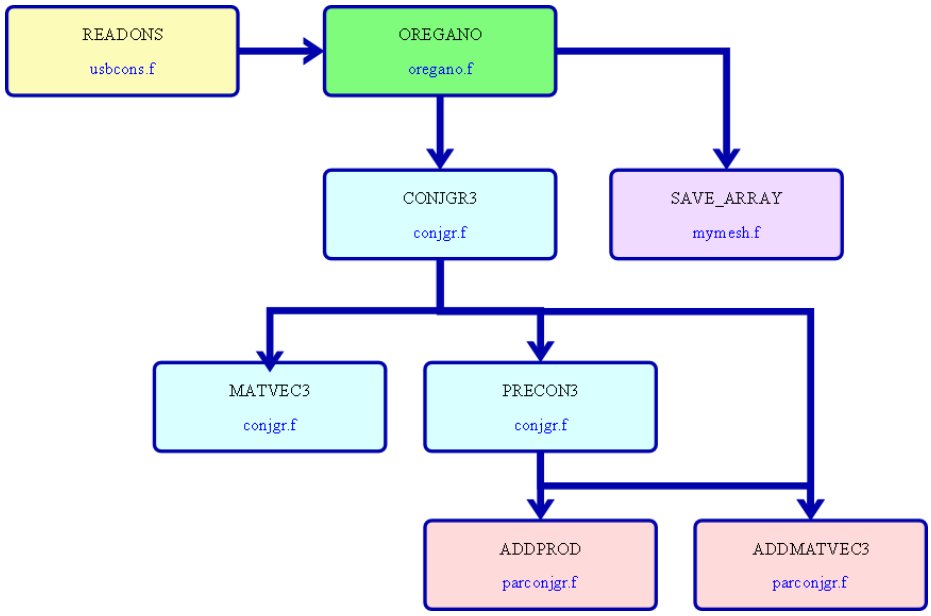


Figure 2.2: Flow chart of modified subroutines. Arrows indicate the direction flag for periodic boundary condition (PBC) to be passed.

2.2.2 List of files and subroutines modified

- `oregano.f`
  - OREGANO, lines: 42, 312, 317, 730, 980, 1123, 1248, 1326

- `vsbcons.f`
  - READONS, lines: 8, 12, 58 to 100
- `mymesh.f`
  - SAVE_ARRAY, lines: 2074, 2096, 2180
- `conjgr3.f`
  - CONJGR3, extensive alterations
  - PRECON3, extensive alterations
- `parconjgr.f`
  - ADDMATVEC3, extensive alterations
  - ADDPROD, extensive alterations

Full listings of these subroutines may be found in Appendix A.

2.2.3 Input and output options

To make the periodic boundary condition an option available in the input file, it was necessary to introduce an integer flag, PBC, which is passed through the program so the subroutines in `conjgr3.f` and `parconjgr.f` know to apply the appropriate boundary conditions.

The flag is declared on line 42 of `oregano.f`, and initialised to 0 in line 312. The other occurrences in `oregano.f` are passing and returning the flag to the other subroutines in which it is used. Although it is defined as an integer, the program treats it as a 3 digit binary number between 000 and 111, with the digits corresponding to XYZ periodic boundary conditions respectively (i.e. 101 would imply periodic boundary conditions on both the X and Z boundaries).

The input routine (READONS in `vsbcons.f`) recognises the word PERIODIC and then expects the next word to be X, Y or Z. The integer flag PBC is then correspondingly increased by 100, 10 or 1 (for X, Y and Z), with appropriate error handling to ensure it remains in the correct limits. At present the code ignores the first two digits, corresponding to X or Y periodicity as this has not been implemented and only checks to see if it ends in a 0 or 1 for Z periodicity.

2.2.4 Modification of Parallelisation Routines for Periodic Boundary Conditions in Z

The problem is parallelised by slicing the solution domain into layers perpendicular to the z-direction; a schematic with two MPI processes is shown in Figure 2.3. The fortran files `conjgr3.f` and `parconjgr.f` contain all of the MPI calls for matching the boundaries between the slices of the solution domain that have been distributed to the computational units. The original code has been modified so that not only are the surfaces matched between adjacent computational units to ensure continuity, but if the PBC flag ends in a 1 then the two boundary surfaces at z_{max} and z_{min} will be matched between the appropriate MPI processes.

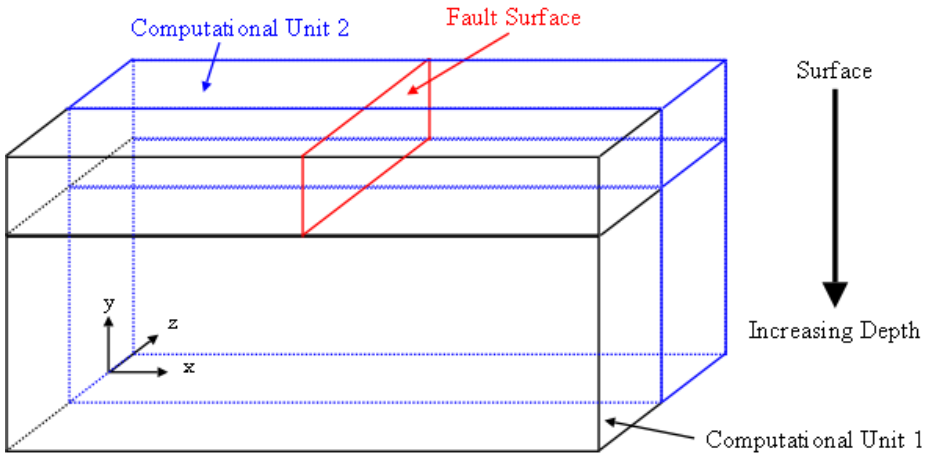


Figure 2.3: Schematic of parallelisation of solution domain with two MPI processes (black and blue respectively) and a fault (in red).

Sample code from `parconjgr.f` is shown below demonstrating how the routine identifies the different surfaces and passes appropriate information between processors.

```
24
25 C     INITIALISE SU and SL
26 C        store contributions to SU from this processor's
27 C        boundary nodes. Top surface.
28          DO I=1,NSEND
29             SU(I)=RES(ISU(I))
30          ENDDO
31       IF (RANK.NE.NPROCS) THEN
32 C        if not top processor send values of RES to processor above
33          CALL MPI_ISEND(SU,NSEND,MPI_DOUBLE_PRECISION,
34      .           RANK,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
35       ENDIF
36       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
37 C     check flag matches pbc in z, (1,11,101 or 111)
38          IF (RANK.EQ.NPROCS) THEN
39 C           if top processor send values of RES to bottom processor
40          CALL MPI_ISEND(SU,NSEND,MPI_DOUBLE_PRECISION,
41      .           0,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
42          ENDIF
43       ENDIF
44
45       IF (RANK.NE.1) THEN
46 C        if not bottom processor receive values of RES from processor below
47          CALL MPI_RECV(SL,NSEND,MPI_DOUBLE_PRECISION,
48      .          RANK-2,0,MPI_COMM_WORLD,STAT1,ERCODE)
49
50 C        add received values from processor below to
```

```
C        the RES vector
         DO I=1,NSEND
            RES(ISL(I))=RES(ISL(I))+SL(I)
         ENDDO
      ENDIF
      IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
C     check flag matches pbc in z, (1,11,101 or 111)
         IF (RANK.EQ.1) THEN
C           if bottom processor receive values of RES from top processor
         CALL MPI_RECV(SL,NSEND,MPI_DOUBLE_PRECISION,
     .          NPROCS-1,0,MPI_COMM_WORLD,STAT1,ERCODE)
C           add received values from processor below to
C           the RES vector
            DO I=1,NSEND
               RES(ISL(I))=RES(ISL(I))+SL(I)
            ENDDO
         ENDIF

      ENDIF
      IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
         CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
      ENDIF
      IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN
         IF(RANK.NE.NPROCS)CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
      ENDIF
```

2.3 Benchmarking

Now that the code has been modified, it must be compared to benchmarks to check that it is operating within the expected parameters. There are a number of benchmark

tests that are considered in the following sections:

- Model for viscoelastic relaxation following a rupture on a strike slip fault, *Nur and Mavko* (1974).
- Novel solution for simple shear under a constant tangental stress with depth dependence viscosity and non-linear stress strain-rate relation.
- Imposed periodic stresses/displacements at the surface, as in Figure 2.7.

2.4 Viscoelastic relaxation following a rupture on a strike slip fault, after *Nur and Mavko* (1974)

The surface displacements for an infinitely long strike-slip fault in an elastic elayer over a viscoelastic half-space can be calculated using the solution to an elastic layer over an elastic half-space with contrasting shear modulus. Then the correspondence principle can be used to to obtain the solution to the corresponding viscoelastic problem from the elastic solution. The solution to this problem was first published by *Nur and Mavko* (1974), where all motions are assumed anti-plane,

$$u_z(x,t) = \frac{\Delta u}{\pi}\left\{\arctan\left(\frac{D}{x}\right) + \sum_{n=1}^{\infty}\left[1 - e^{-t/t_R}\sum_{m=1}^{n}\frac{(t/t_R)^{n-m}}{(n-m)!}\right]F_n(x,D,H)\right\}, \quad (2.4a)$$

$$F_n(x,D,H) = \arctan\left(\frac{2xD}{x^2 + (2nH)^2 - D^2}\right), \quad (2.4b)$$

where slip on the fault is given by Δu, the depth of dislocation D, thickness of elastic lid H, and the relaxation time $t_R = 2\eta/\mu$ (N.B. this definition differs from my usual convention by a factor of 2, the literature is inconsistent on this point). The coordinate system remains as illustrated in Figure 2.3, evaluated at the surface ($y = y_{max}$).

This can be used to test both the periodic boundary condition in *oregano_ve* and the identification of mesh nodes on the fault. Firstly the equations are non-dimensionalised by slip (Δu) for displacements, and elastic lid thickness (H) for dimensions. A box of

dimensions $60 \times 40 \times 1$ was used, with a node spacing of 0.5 in the vertical axis (to better constrain slip depth) and 1 in the horizontal axis. These horizontal and vertical dimensions were chosen to approximate a half-space as they are both much greater than the slip depth. The simulations were run for 100 time steps, corresponding to $t \in [0, 4.167]$ Maxwell times. The fault was located at $x = 30, 39 \leq y \leq 40$ with an elastic lid thickness of 1 and the fault ruptures the elastic lid ($D = H$). . Movement in the x-direction was prevented by setting $u_x = 0$ on $x \in \{0, 60\}$, but these boundaries allowed to slide ($\sigma_{xz} = \sigma_{yz} = 0$) so that the deformation lies in the $y - z$ plane on those boundaries. On $z \in \{0, 1\}$, it is assumed that the solution is periodic in the z-direction. Twenty surface data points are tracked either side of the fault, ignoring those closest to the boundary as they are likely to be subject to boundary effects where shear stresses are assumed to be zero. The other parameters are:

$$\begin{aligned} &\Delta u = 2.00, \quad \eta_e = 1.0 \times 10^{20}, \quad \eta_{ve} = 1.0, \quad \mu = 1.0, \\ &\nu = 0.25, \qquad D = 1.0, \qquad H = 1.0. \end{aligned}$$

The model is non-dimensional with a total slip of 2 units, although it is specified in the *oregano_ve* jobfile as 1, because *oregano_ve* uses a split node technique and each side of the fault will move by 1. The viscosities and shear moduli are also non-dimensionalised by the viscosity and shear modulus in the half-space. With these parameters the relaxation timescale is $t_R = 2$.

Following the conclusion of the numerical calculation, the results are inverted to recover the initial parameters using a non-linear least squares solver in Mathematica 9.0 (*Wolfram Research* (2012)). Equation (2.4) has 4 free parameters (H, D, Δu, and t_R), and two variables (x and t), so to fit this model to the numerical simulation I first eliminate two parameters. The $t = 0$ elastic solution should match

$$u_z(x) = \frac{\Delta u}{\pi} \arctan\left(\frac{D}{x}\right), \tag{2.5}$$

and I chose to fix $\Delta u = 2$ and solve for D. Once this has been determined, the remaining 4000 data points output from the numerical simulation are used to solve for

H and t_R. Table 2.1 shows the expected and derived results, as well as the standard error associated with the fit.

Parameter	Input	Fit	Standard Error
Δu	2.0	2.0	N/A
D	1.0	1.0973	0.0019
H	1.0	1.1079	0.0004
t_R	2.0	2.0464	0.0009

Table 2.1: Least-squares fit for dislocation in elastic lid over viscoelastic half-space. Numerical model from *oregano_ve* and fit to equations 2.4 achieved using a non-linear least-squares routine in Mathematica 9.0 (*Wolfram Research* (2012)).

These results are in agreement with that predicted given the node spacing on the finite element mesh is 0.5. This fit can be seen in Figure 2.4.

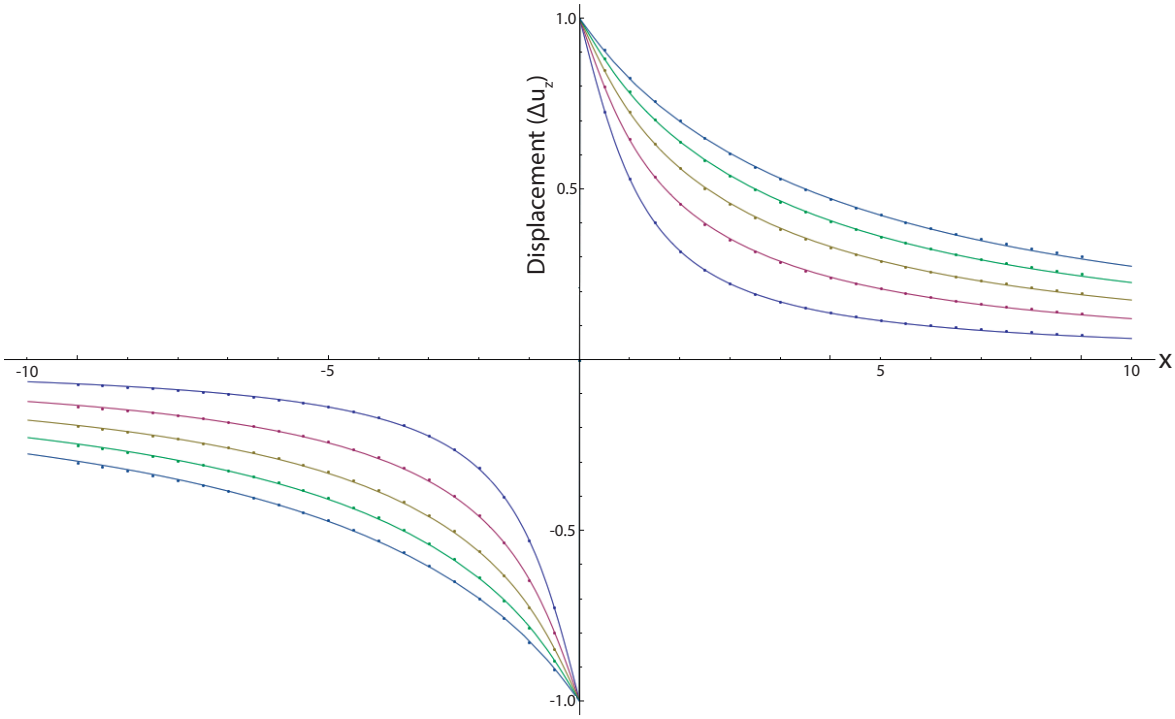


Figure 2.4: Fit for dislocation in elastic lid over viscoelastic half-space, after Nur and Mavko (1974). Data points from *oregano_ve* simulation with analytical solution as solid line. Each curve represents 1 Maxwell time.

2.5 Simple shear, constant tangental stress, oregano viscosity rules

This test requires periodicity in one dimension, and also investigates the viscosity structures available in *oregano_ve*. These are structures that I will be investigating in more detail in Chapters 3 and 4

This model is a rectangular block of dimensions $a \times b \times c$, where I impose a dimensionless stress in the z-direction: $\sigma_{yz} = \tau_0$ on the surface $y = b$, and set displacement $u_z = 0$ on the opposite side ($y = 0$). On both of these surfaces I also set $u_y = 0$ and $\sigma_{xz} = 0$. Movement in the x-direction is prevented by setting $u_x = 0$ on $x \in \{0, a\}$ but these boundaries are allowed to slide ($\sigma_{xz} = \sigma_{xy} = 0$) so that the deformation is purely 2D in the $y - z$ plane. On $z \in \{0, c\}$, assume that the solution is periodic in the z-direction. I assume a viscoelastic rheology with depth-dependent viscosity as defined by oregano rules

$$\text{VR} = 0 \qquad \eta = \eta_0, \tag{2.6a}$$

$$\text{VR} = 1 \qquad \eta = \eta_0 + \eta_1 y, \tag{2.6b}$$

$$\text{VR} = 2 \qquad \eta = \eta_0 \exp(\eta_1 y), \tag{2.6c}$$

$$\text{VR} = 3 \qquad \eta = \eta_0/(1.0 - \eta_1 y). \tag{2.6d}$$

First, solve for the time-zero elastic solution, which is identical in all cases,

$$\tau_0 = 2\mu\epsilon^e_{yz} = 0. \tag{2.7}$$

Then, using $2\mu = \frac{E}{(1+\nu)}$,

$$\epsilon^e_{yz} = \frac{\tau_0}{2\mu} = (1+\nu)\frac{\tau_0}{E}. \tag{2.8}$$

There is no displacement in the y-direction, so z-direction elastic displacement on $y = b$ is

$$u_z = 2b(1+\nu)\frac{\tau_0}{E}. \tag{2.9}$$

In this simple shear problem, there are no displacements in the x or y directions, and the z-direction displacements (u_z) are constant in x and z, so the following components of elastic strain are zero:

$$\epsilon_{xx} = \epsilon_{yy} = \epsilon_{zz} = \epsilon_{xy} = \epsilon_{xz} = 0. \tag{2.10}$$

The corresponding components of elastic stress must also be zero,

$$\sigma_{xx} = \sigma_{yy} = \sigma_{zz} = \sigma_{xy} = \sigma_{xz} = 0. \tag{2.11}$$

Thus both dilatation and pressure are zero, and the viscous strain rate follows directly from inverting the maxwell constitutive relation, equation, (1.13),

$$\dot{\epsilon}^v_{yz} = \left(\frac{\mu}{\eta}\right)\epsilon^e_{yz} = \left(\frac{\mu}{E}\right)(1+\nu)\frac{\tau_0}{\eta} = \frac{\tau_0}{2\eta}. \tag{2.12}$$

From this we can substitute in the viscosities, equations (2.6), and integrate to calculate the viscous displacement rate at a point y, where $0 < y < b$,

$$\text{VR} = 0 \qquad \dot{u}_z = \frac{\tau_0}{\eta_0}y, \tag{2.13a}$$

$$\text{VR} = 1 \qquad \dot{u}_z = \frac{\tau_0}{\eta_1}\log\left(1 + \frac{\eta_1}{\eta_0}y\right), \tag{2.13b}$$

$$\text{VR} = 2 \qquad \dot{u}_z = \frac{\tau_0}{\eta_0\eta_1}(1 - \exp\left(-\eta_1 y\right)), \tag{2.13c}$$

$$\text{VR} = 3 \qquad \dot{u}_z = \frac{\tau_0}{\eta_0}\left(y - \frac{\eta_1}{2}y^2\right). \tag{2.13d}$$

The solutions for both elastic strain and viscous strain-rate are constant in time and space as the driving shear stresses are constant for $t > 0$.

2.5.1 Power law rheology

The benchmark simple shear under constant tangental stress can also be extended to allow for a power law rheology. Assuming the same conditions as before, but with a

viscoelastic power law rheology with uniform properties, I find an effective viscosity

$$\eta = \eta_0(\sigma'_{ij}\sigma'_{ij})^{\frac{1-n}{2}}. \tag{2.14}$$

The time-zero elastic solution is as equation (2.9) and both dilatation and pressure are again zero so the viscous strain rate follows directly from equation (2.12). I then need the effective viscosity from (2.14), with only σ_{yz} or σ_{zy} non zero,

$$\eta = \eta_0(\sigma'_{yz}\sigma'_{yz} + \sigma'_{zy}\sigma'_{zy})^{\frac{1-n}{2}}, \tag{2.15a}$$

$$\eta = \eta_0(2\sigma'_{yz}\sigma'_{yz})^{\frac{1-n}{2}}. \tag{2.15b}$$

Deviatoric stress is dependent upon the elastic strain as

$$\sigma'_{yz} = 2\mu\epsilon^e_{yz}. \tag{2.16}$$

Substituting equation (2.16) into equation (2.15) gives

$$\eta = \eta_0(8\mu^2(\epsilon^e_{yz})^2)^{\frac{1-n}{2}}, \tag{2.17}$$

and substituting equation (2.8) into equation (2.17) gives

$$\eta = \eta_0 2^{\frac{1-n}{2}}\tau_0^{1-n}. \tag{2.18}$$

Returning to the viscous strain rate of equation (2.12) we now have

$$\dot{\epsilon}^v_{yz} = \frac{\tau_0}{2\eta} = \frac{\tau_0}{2\eta_0}2^{\frac{n-1}{2}}\tau_0^{n-1} = 2^{\frac{1-n}{2}}\frac{\tau_0^n}{2\eta_0}. \tag{2.19}$$

From this we can calculate the viscous displacement rate on $y = b$,

$$\dot{u}_z = 2^{\frac{1-n}{2}}\frac{\tau_0^n}{\eta_0}b. \tag{2.20}$$

As before, the solutions for both elastic strain and viscous strain-rate are constant in time and space as the driving shear stresses are constant for $t > 0$.

2.5.2 Test results from *oregano_ve*

Test results were obtained using *oregano_ve* version 3.0. In all cases, a box of dimensions $1 \times 1 \times 1$ was used, node spacing of 0.1 and periodic boundary condition in z. The simulations were run for 500 time steps, corresponding to $t \in [0, 20]$. A dimensionless stress $\sigma_{yz} = \tau_0 = 1$ was applied on the surface $y = 1.0$. At every time step the displacement in the z-direction (u_z) is tracked on 10 points from $y = 0$ to $y = 1$, as shown in Figure 2.5. Multiple z locations are tracked (0.0, 0.5 and 1.0) with matching

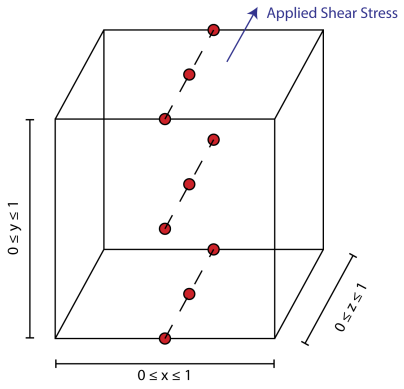


Figure 2.5: Tracked nodes on solution domain highlighted in red. Other nodes are at a spacing of 0.1, not shown for clarity.

y as I would expect all data points with matching y-coordinates to register identical displacements. This will determine whether the code implementing the periodicity correctly. A range of parameters were tested; for brevity I shall discuss results with the following parameters.

$$\tau_0 = 1.00 \text{ Pa} \quad \eta_0 = 1.00 \text{ Pa s} \quad \eta_1 = 1.00 \text{ Pa s} \quad E = 2.50 \text{ Pa} \quad \nu = 0.25$$

Integrating $\dot{u}_z$ from equations (2.13) with respect to t, using equation (2.9) as the initial condition and substituting in the above constants gives the expected displacements for

$t > 0$, $x \in [0, 1]$, and $y \in [0, 1]$,

$$VR = 0 \qquad u_z = y(1 + t), \tag{2.21a}$$

$$VR = 1 \qquad u_z = y + \log(1 + y)\, t, \tag{2.21b}$$

$$VR = 2 \qquad u_z = y + (1 - \exp(-y))t, \tag{2.21c}$$

$$VR = 3 \qquad u_z = y + \left(y - \frac{y^2}{2}\right) t. \tag{2.21d}$$

These functions are plotted against numerical data in Figure 2.6.

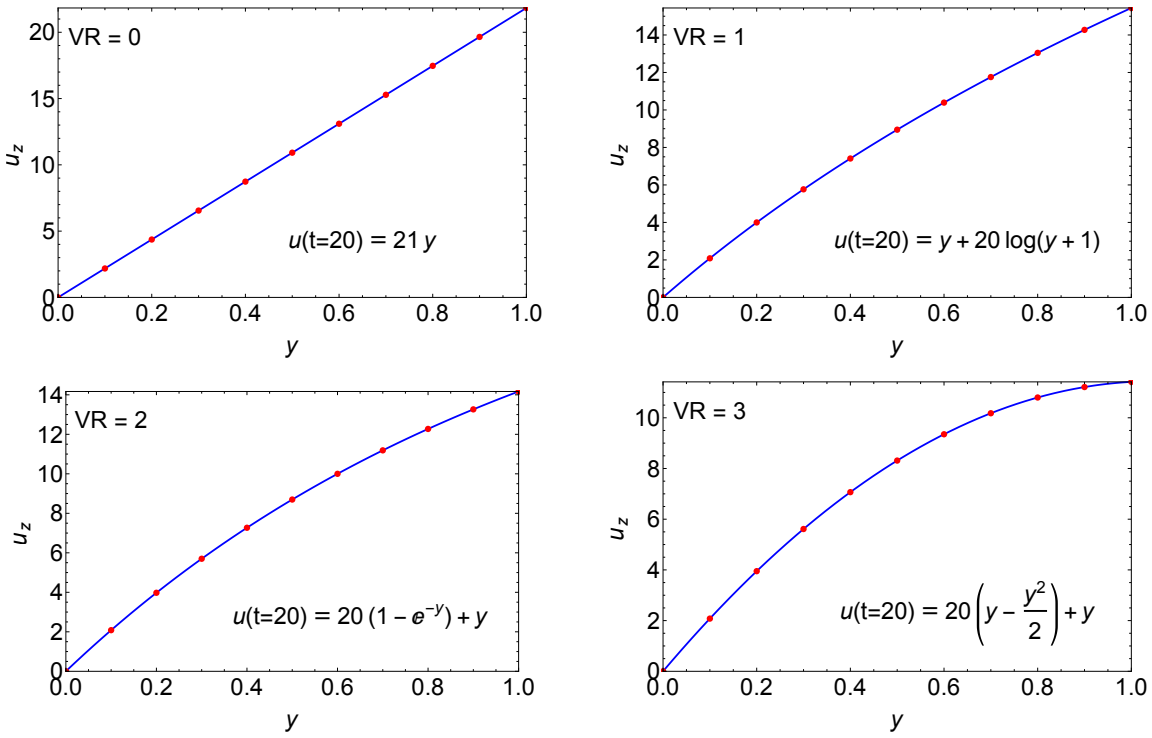


Figure 2.6: Fit at 20 Maxwell times for different *oregano_ve* viscosity rules. Red dots are numerical calculations performed in *oregano_ve* and analytical solutions (equations 2.21) plotted in blue.

2.6 Viscoelastic relaxation of imposed harmonic tractions and displacements

In this section, I examine the effects of a harmonic shear stress or displacement condition, with wavenumber k, applied at the surface of a 2D plane where all motions are assumed to be anti-plane. The coordinate system defined as y positive upwards, x horizontal and z perpendicular to plane, as shown in Figure 2.7. The surface boundary

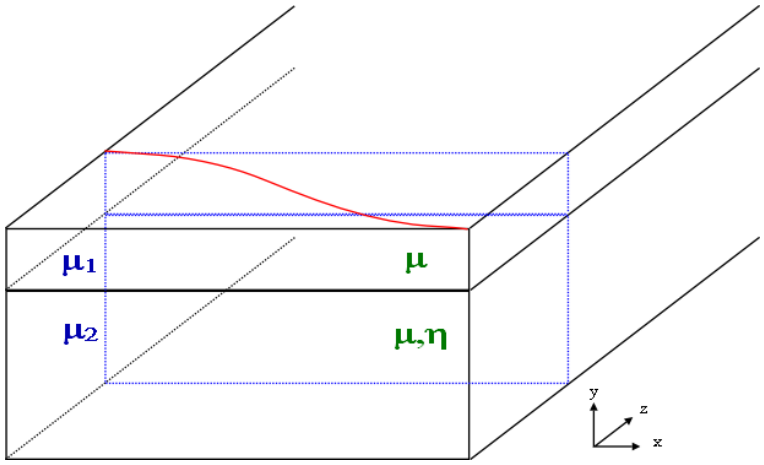


Figure 2.7: Periodic stress/displacement in the z-direction is imposed on the top surface of the domain, whilst the side boundaries are allowed to freely slide.

conditions are given by

$$\text{Shear stress surface condition:} \qquad \sigma_{yz}(x,d) = A\sin(kx), \tag{2.22a}$$

$$\text{Displacement surface condition:} \qquad u_{+}(x,d) = A\sin(kx), \tag{2.22b}$$

whilst the remaining boundary conditions are given by

$$u(0, y) = 0, \tag{2.23a}$$

$$u(w, y) = 0, \tag{2.23b}$$

$$\lim_{y \to -\infty} [u(x, y)] \to 0, \tag{2.23c}$$

$$u_+(x, 0) = u_-(x, 0), \tag{2.23d}$$

$$\mu_+ \left.\frac{\partial u_+}{\partial y}\right|_{y=0} = \mu_- \left.\frac{\partial u_-}{\partial y}\right|_{y=0}. \tag{2.23e}$$

The domain to be solved is divided into two, an elastic lid $y \in [0, d]$ and a viscoelastic half-space $y \in (-\infty, 0]$. The problem is initially solved as two elastic regions with contrasting shear moduli, μ_+ and μ_-, which entails solving Laplace's equation with continuity conditions of displacement and shear stresses on the interface $y = 0$. The correspondence principle is then utilised and the inverse Laplace transform calculated to derive the time-dependent viscoelastic solution.

2.6.1 Elastic half-space

The general result for antiplane elastic problems is the displacement field $[0, 0, u(x, y)]^T$ which must satisfy Laplace's equation,

$$\nabla^2 u(x, y) = 0. \tag{2.24}$$

I apply a periodic stress condition on the surface $y = 0$ for a region $0 < x < w$, where $k = \frac{2\pi}{w}$,

$$\sigma_{yz}(x, 0) = A \sin(kx). \tag{2.25}$$

Anti-plane elastic stresses and strains, shear modulus μ,

$$\sigma_{ij} = \mu \epsilon_{ij}, \tag{2.26a}$$

$$\epsilon_{yz} = \frac{1}{2} \frac{\partial u}{\partial y}. \tag{2.26b}$$

Combining equations (2.26) and (2.25) gives the surface boundary condition

$$\left.\frac{\partial u}{\partial y}\right|_{y=0} = \frac{A}{\mu}\sin(kx)\,, \tag{2.27}$$

with the remaining boundary conditions

$$u(0,y) = 0, \tag{2.28a}$$

$$u(w,y) = 0, \tag{2.28b}$$

$$\lim_{y\to-\infty}[u(x,y)] \to 0. \tag{2.28c}$$

If I assume the solution to equation (2.24) is of the form $u(x,y) = X(x)Y(y)$, then subject to boundary conditions (2.27 and 2.28) the solution is given by

$$u(x,y) = \frac{A}{k\mu}e^{ky}\sin(kx)\,, \tag{2.29}$$

which is plotted in Figure 2.8

2.6.2 Elastic layer over elastic half-space

The domain is divided into an elastic layer ($0 < y < d$), with shear modulus μ_+ and thickness d over an elastic half-space ($y < 0$) with shear modulus μ_-. Equation (2.24) must be satisfied in both regions,

$$\nabla^2 u_+(x,y) = 0, \qquad 0 < y < d, \tag{2.30a}$$

$$\nabla^2 u_-(x,y) = 0, \qquad y < 0, \tag{2.30b}$$

with boundary conditions from equations (2.28) and

$$\left.\frac{\partial u}{\partial y}\right|_{y=d} = \frac{A}{\mu_+}\sin(kx)\,. \tag{2.31}$$

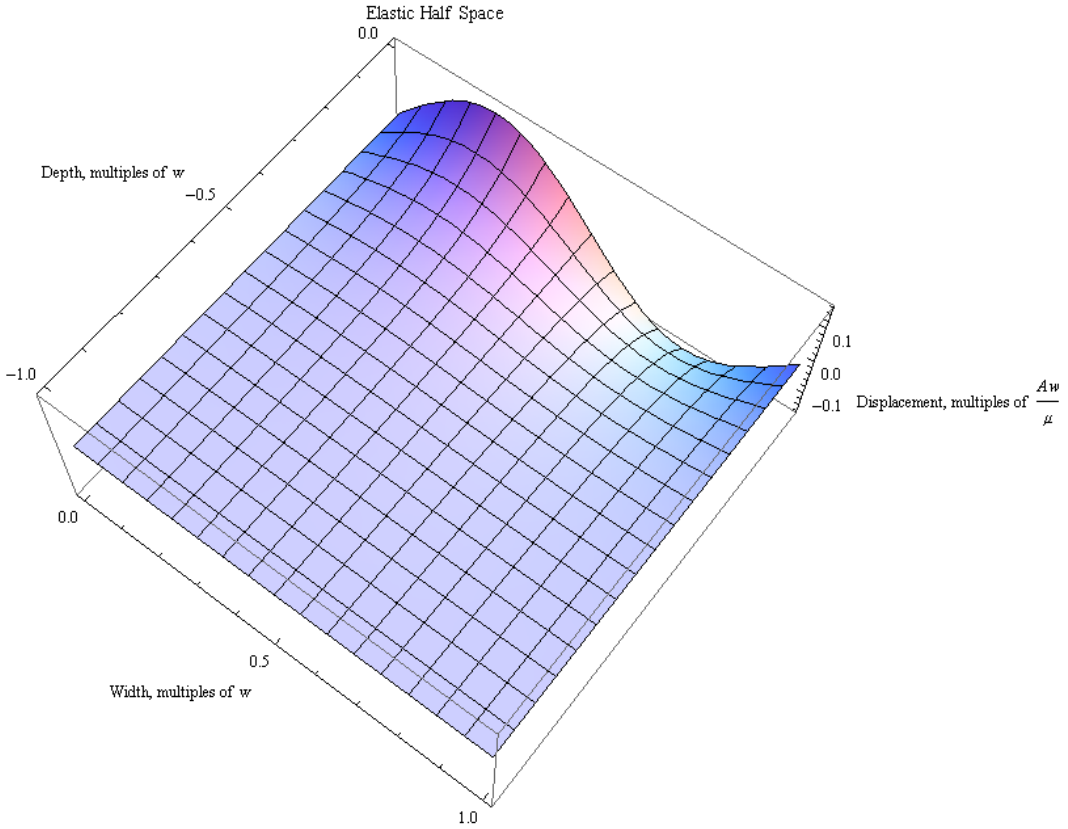


Figure 2.8: Soution for elastic half-space.

The two solutions must also give continuous displacements and shear stresses ($\sigma_{yz+} = \sigma_{yz-}$) across the interface ($y = 0$), resulting in matching conditions

$$u_+(x,0) = u_-(x,0), \tag{2.32a}$$

$$\mu_+ \left.\frac{\partial u_+}{\partial y}\right|_{y=0} = \mu_- \left.\frac{\partial u_-}{\partial y}\right|_{y=0}. \tag{2.32b}$$

General solutions to equation (2.30) are given below, again assuming the solution is separable ($u(x,y) = X(x)Y(y)$), with constants $\alpha, \beta, \gamma, \delta$ to match boundary and interface conditions

$$u_+ = \sin(kx)\left(\alpha e^{ky} + \beta e^{-ky}\right), \tag{2.33a}$$

$$u_- = \sin(kx)\left(\gamma e^{ky} + \delta e^{-ky}\right). \tag{2.33b}$$

Substituting equations (2.33) into the boundary conditions given in equations (2.28 and 2.31) and interface conditions (2.32) produces the matching equations

$$\delta = 0, \tag{2.34a}$$

$$\alpha + \beta = \gamma, \tag{2.34b}$$

$$\gamma\mu_- + \beta\mu_+ - \alpha\mu_+ = 0, \tag{2.34c}$$

$$\alpha e^{dk} - \beta e^{-dk} = \frac{A}{k\mu_+}. \tag{2.34d}$$

The coeffiecients that satisfy equations (2.34) are

$$\alpha = \frac{A(\mu_- + \mu_+)}{2k\mu_+(\mu_- \cosh(kd) + \mu_+ \sinh(kd))}, \tag{2.35a}$$

$$\beta = \frac{A(-\mu_- + \mu_+)}{2k\mu_+(\mu_- \cosh(kd) + \mu_+ \sinh(kd))}, \tag{2.35b}$$

$$\gamma = \frac{A}{k(\mu_- \cosh(kd) + \mu_+ \sinh(kd))}. \tag{2.35c}$$

In the limiting case $\mu_- \to \mu_+$, it should reduce to equation (2.29) from Section 2.6.1,

$$\lim_{\mu_- \to \mu_+} [\alpha] \to \frac{Ae^{-dk}}{k\mu_+}, \tag{2.36a}$$

$$\lim_{\mu_- \to \mu_+} [\beta] \to 0, \tag{2.36b}$$

$$\lim_{\mu_- \to \mu_+} [\gamma] \to \frac{Ae^{-dk}}{k\mu_+}. \tag{2.36c}$$

This corresponds to equation (2.29) with a shift $u(x, y + d)$, which is commensurate with the change in the surface boundary condition (2.27 and 2.31) from $y = 0$ to $y = d$. Substituting the coefficients (2.35) back in to the general solutions (2.33) gives us the solutions in the two domains as

$$u_+(x, y) = \frac{A\sin(kx)(\mu_- \sinh(ky) + \mu_+ \cosh(ky))}{k\mu_+(\mu_- \cosh(kd) + \mu_+ \sinh(kd))} \qquad 0 < y < d, \tag{2.37a}$$

$$u_-(x, y) = \frac{Ae^{ky}\sin(kx)}{k(\mu_+ \sinh(kd) + \mu_- \cosh(kd))} \qquad y < 0. \tag{2.37b}$$

2.6.3 Elastic layer over viscoelastic half-space with constant shear moduli (μ)

The solution for the elastic layer over the elastic half-space, equations (2.37), can be utilised to solve the problem of an elastic lid over a viscoelastic half-space using the correspondence principle. Firstly, the surface condition is modified to allow for time dependence,

$$\sigma_{yz}(x,t) = A\sin\left(\frac{2\pi x}{w}\right)H(t). \tag{2.38}$$

Where $H(t)$ is the Heaviside or Step function,

$$H(t) = \begin{cases} 0 & t < 0, \\ 1 & t > 0. \end{cases} \tag{2.39}$$

In both regions, I am now working in the Laplace transform domain, so two modifications are required. Firstly the solutions from equations (2.37) must be multiplied by the Laplace transform of the Heaviside function $(1/s)$, where I will follow the convention of using s as the Laplace transform variable, and then the shear modulus of the lower layer (μ_-) must be replaced with

$$\mu_- \to \frac{s}{\mu^{-1}s + \eta^{-1}}. \tag{2.40}$$

At the same time, I rename $\mu_+ \to \mu$, so that μ corresponds to the shear modulus in both the elastic layer and the viscoelastic half-space ($\mu = \mu_e = \mu_{ve}$). This gives the solution in the Laplace transform domain

$$\mathcal{L}[u_e(x,y,t)] = \frac{A\sin(kx)}{sk\mu}\left[\frac{\mu\cosh(ky) + \frac{s}{\mu^{-1}s+\eta^{-1}}\sinh(ky)}{\mu\sinh(kd) + \frac{s}{\mu^{-1}s+\eta^{-1}}\cosh(kd)}\right] \quad 0 < y < d, \tag{2.41a}$$

$$\mathcal{L}[u_{ve}(x,y,t)] = \frac{Ae^{ky}\sin(kx)}{sk}\left[\mu\sinh(kd) + \frac{s}{\mu^{-1}s+\eta^{-1}}\cosh(kd)\right]^{-1} \quad y < 0. \tag{2.41b}$$

I proceed with the analysis of the viscoelastic half-space u_{ve}, with the substitution $\psi = \frac{A}{k}e^{ky}\sin(kx)$ to simplify the resulting expression;

$$\begin{aligned} u_{ve}(x,y,t) &= \psi\mathcal{L}^{-1}\left\{\frac{1}{s}\left[\mu\sinh(kd) + \frac{s}{\mu^{-1}s+\eta^{-1}}\cosh(kd)\right]^{-1}\right\}, \\ &= \psi\mathcal{L}^{-1}\left\{\frac{1}{s}\left(\frac{\mu^{-1}s+\eta^{-1}}{se^{kd}+\mu\eta^{-1}\sinh(kd)}\right)\right\}, \\ &= \psi e^{-kd}\left(\frac{1}{\mu}\mathcal{L}^{-1}\left\{\frac{1}{s+\mu\eta^{-1}e^{-kd}\sinh(kd)}\right\}\right. \\ &\qquad \left. + \frac{1}{\eta}\mathcal{L}^{-1}\left\{\frac{1}{s^2+s\mu\eta^{-1}e^{-kd}\sinh(kd)}\right\}\right). \end{aligned} \tag{2.42}$$

At this stage, some standard results for the inverse Laplace transform can now be utilised,

$$\mathcal{L}^{-1}\left\{\frac{1}{s+a}\right\} = e^{-at}H(t), \tag{2.43a}$$

$$\mathcal{L}^{-1}\left\{\frac{1}{s^2+as}\right\} = \frac{1}{a}(1-e^{-at})H(t). \tag{2.43b}$$

Inspection of equation (2.42) reveals that the same coefficient a is applicable to both inverse transforms,

$$a = \frac{\mu}{\eta}e^{-kd}\sinh(kd). \tag{2.44}$$

Making the substitution from equation (2.44) and using the standard results from equations (2.43) we can perform the inverse Laplace transformations in (2.42),

$$u_{ve}(x,y,t) = H(t)\psi e^{-kd}\left[\frac{1}{\mu}e^{-at} + \frac{1}{\eta a}(1-e^{-at})\right], \tag{2.45a}$$

$$= H(t)\psi e^{-kd}\left[\frac{1}{\eta a} + e^{-at}\left(\frac{1}{\mu} - \frac{1}{\eta a}\right)\right]. \tag{2.45b}$$

Substituting back in for a and ψ, using $\frac{1}{a} = \frac{\eta}{\mu}(1 + \coth(kd))$ gives

$$\begin{aligned} u_{ve}(x,y,t) &= \frac{1}{k}H(t)Ae^{k(y-d)}\sin(kx)\left[\frac{1}{\mu}(1+\coth(kd)) + \frac{e^{-\frac{t}{t_R}}}{\mu}(1-1-\coth(kd))\right], \\ &= \frac{1}{\mu k}H(t)Ae^{k(y-d)}\sin(kx)\left[(1+\coth(kd)) - \coth(kd)e^{-\frac{t}{t_R}}\right], \end{aligned} \tag{2.46}$$

where the decay constant (t_R) is given by

$$t_R = \frac{\eta}{\mu}(1 + \coth(kd)). \tag{2.47}$$

The analysis proceeds similarly for u_e, this time substituting $\omega = \frac{A}{k\mu}\sin(kx)$;

$$\begin{aligned} u_e(x,y,t) &= \omega\mathcal{L}^{-1}\left\{\frac{1}{s}\left(\frac{\mu(\mu^{-1}s+\eta^{-1})\cosh(ky) + s\sinh(ky)}{\mu(\mu^{-1}s+\eta^{-1})\sinh(kd) + s\cosh(kd)}\right)\right\}, \\ &= \omega\left[e^{k(y-d)}\mathcal{L}^{-1}\left\{\frac{1}{s+\mu\eta^{-1}e^{-kd}\sinh(kd)})\right\}\right. \\ &\qquad \left. + \frac{\mu}{\eta}e^{-kd}\cosh(ky)\mathcal{L}^{-1}\left\{\frac{1}{s^2+s\mu\eta^{-1}e^{-kd}\sinh(kd)}\right\}\right]. \end{aligned} \tag{2.48}$$

This requires the same substitution as for u_{ve}, given by equation (2.44), so we can then use the standard results from equations (2.43) to perform the inverse Laplace transformations in equation (2.48),

$$u_e(x,y,t) = \omega\left[e^{k(y-d)}e^{-at}H(t) + \frac{\mu e^{-kd}\cosh(ky)}{\eta a}(1-e^{-at})H(t)\right]. \tag{2.49}$$

Substituting back in for a and ω, again with $\frac{1}{a} = \frac{\eta}{\mu}(1+\coth(kd))$,

$$\begin{aligned} u_e(x,y,t) &= \frac{H(t)A\sin(kx)}{k\mu}\left[e^{k(y-d)}e^{-\frac{t}{t_R}} + e^{-kd}\cosh(ky)(1+\coth(kd))(1-e^{-\frac{t}{t_R}})\right], \\ u_e(x,y,t) &= \frac{H(t)Ae^{-kd}\sin(kx)}{k\mu}\left(\cosh(ky)(1+\coth(kd))\right. \\ &\qquad \left. - \cosh(k(d-y))\,\mathrm{csch}(kd)e^{-\frac{t}{t_R}}\right), \end{aligned} \tag{2.50}$$

where the decay constant, t_R, is also given by (2.47). The complete solution is then

$$u_e(x,y,t) = \frac{H(t)Ae^{-kd}\sin(kx)}{k\mu}\Bigg(\cosh(ky)(1+\coth(kd)) - \cosh(k(d-y))\operatorname{csch}(kd)e^{-\frac{t}{t_R}}\Bigg), \qquad 0<y<d, \tag{2.51a}$$

$$u_{ve}(x,y,t) = \frac{H(t)Ae^{k(y-d)}\sin(kx)}{k\mu}\left[(1+\coth(kd)) - \coth(kd)e^{-\frac{t}{t_R}}\right], y<0, \tag{2.51b}$$

$$t_R = \frac{\eta}{\mu}(1+\coth(kd)), \tag{2.51c}$$

which is plotted in Figures 2.9 and 2.10.

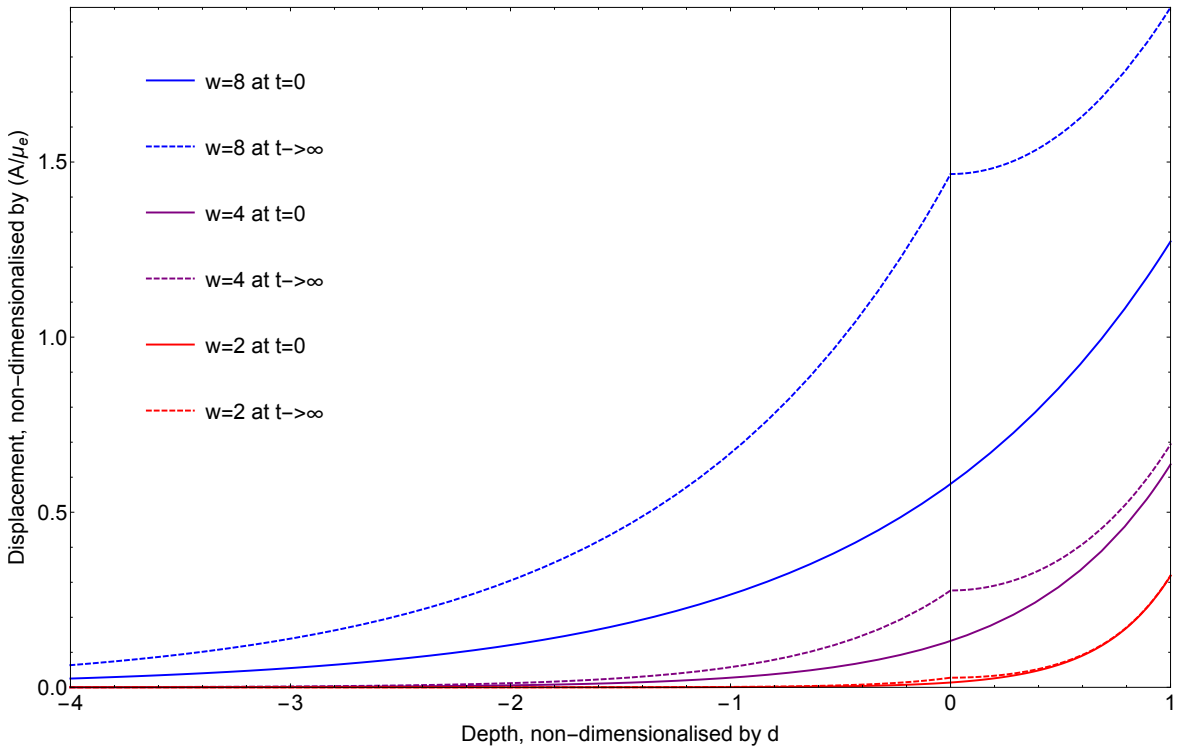


Figure 2.9: Soution for elastic layer over viscoelastic half space, surface traction condition, taken as vertical slice at $x = \frac{w}{4}$ for maximum displacement. Constant shear modulus, variable wavelength, total displacement.

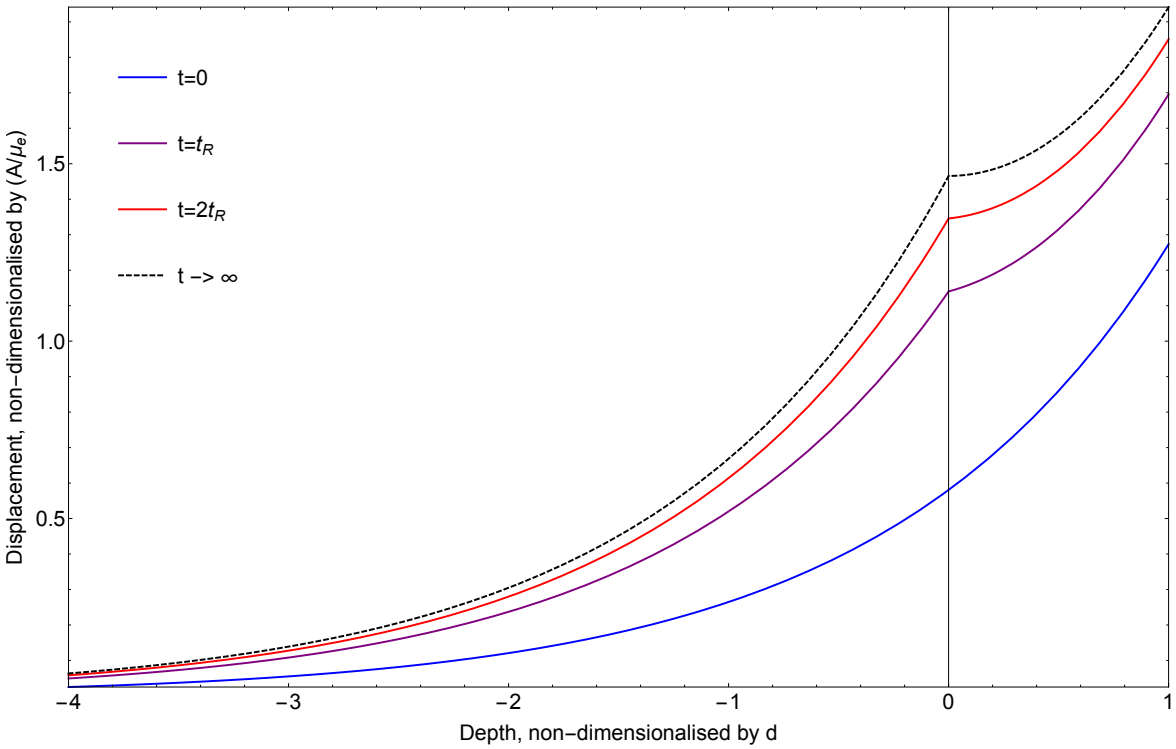


Figure 2.10: Soution for elastic layer over viscoelastic half space with $w = 8$, taken as vertical slice at $x = 2$ for maximum displacement. Constant shear modulus, total displacement.

2.6.4 Elastic layer over viscoelastic half-space with contrasting shear moduli (μ_e and μ_{ve})

The problem can be further extended to that of an elastic layer over a viscoelastic half-space with contrasting shear moduli. I begin with the same modification to the surface condition as in Section 2.38, except now we replace the shear modulus of the lower layer (μ_-) with

$$\mu_- = \frac{s}{\mu_{ve}^{-1} s + \eta^{-1}}. \tag{2.52}$$

At the same time we rename $\mu_+ \to \mu_e$, so μ_e corresponds to the shear modulus in the elastic layer and μ_{ve} the shear modulus in the viscoelastic half-space, which gives us

the solution in the Laplace transform domain as

$$\mathcal{L}[u_e(x,y,t)] = \frac{A\sin(kx)(s\eta\mu_{ve}\sinh(ky) + \mu_e(s\eta+\mu_{ve})\cosh(ky)}{sk\mu_e(s\eta\mu_{ve}\cosh(kd) + \mu_e(s\eta+\mu_{ve})\sinh(kd))},\ 0<y<d, \quad (2.53a)$$

$$\mathcal{L}[u_{ve}(x,y,t)] = \frac{2A\sin(kx)e^{k(d+y)}(s\eta+\mu_{ve})}{sk(s\eta\mu_{ve}\cosh(kd) + \mu_e(s\eta+\mu_{ve})\sinh(kd))},\ y<0. \quad (2.53b)$$

Taking the inverse Laplace transformation of equation (2.53), using the standard results from equations (2.43).

$$u_e(x,y,t) = \frac{A\mu_{ve}\sin(kx)}{k\mu_e(\mu_e\sinh(kd)+\mu_{ve}\cosh(kd))}\left(\cosh(ky)\left(\frac{\mu_e}{\mu_{ve}}+\coth(kd)\right) - \right.$$
$$\left. - \cosh(k(d-y))\,\mathrm{csch}(kd)e^{-\frac{\mu_e\mu_{ve}}{\eta(\mu_e+\mu_{ve}\coth(kd))}t}\right), \qquad 0<y<d, \quad (2.54a)$$

$$u_{ve}(x,y,t) = \frac{Ae^{ky}\sin(kx)\left(\mu_e+\mu_{ve}\coth(kd)\left(1-e^{-\frac{\mu_e\mu_{ve}}{\eta(\mu_e+\mu_{ve}\coth(kd))}t}\right)\right)}{k\mu_e(\mu_e\sinh(kd)+\mu_{ve}\cosh(kd))}, y<0. \quad (2.54b)$$

The elastic and viscous components of this solution are shown in Figures 2.11 and 2.12 respectively.

2.6.5 Elastic layer with zero shear traction at base

In this variation, I have an elastic layer ($0<y<d$) with boundary conditions given in equations (2.28 and 2.31) with the additional constraint of

$$\left.\frac{\partial u}{\partial y}\right|_{y=0} = 0. \quad (2.55)$$

The general solution is the same as u_+ from equation (2.33), and the boundary conditions given in equations (2.31 and 2.55) give us two equations (2.56) to be solved for

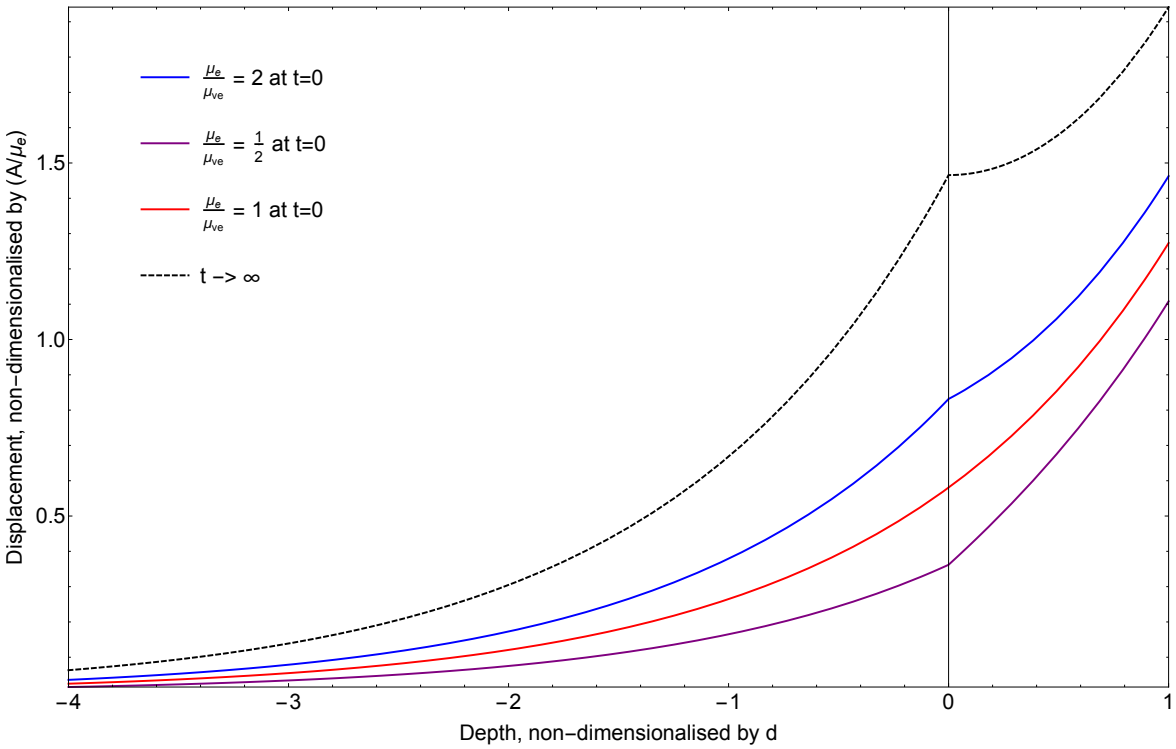


Figure 2.11: Soution for elastic layer over viscoelastic half space for $w = 8$, taken as vertical slice at $x = 2$ for maximum displacement. Contrasting shear modulus, total displacement.

α and β,

$$\alpha = \beta, \tag{2.56a}$$

$$\alpha = \frac{1}{2k\mu \sinh(kd)}, \tag{2.56b}$$

with solution,

$$u(x, y) = \frac{A \sin(kx) \cosh(ky)}{k\mu \sinh(kd)}. \tag{2.57}$$

2.6.6 Viscoelastic half-space

If we take the solution to the elastic half-space, equation (2.29), make the substitution from equation (2.58) and take the inverse Laplace transform of the result (including a

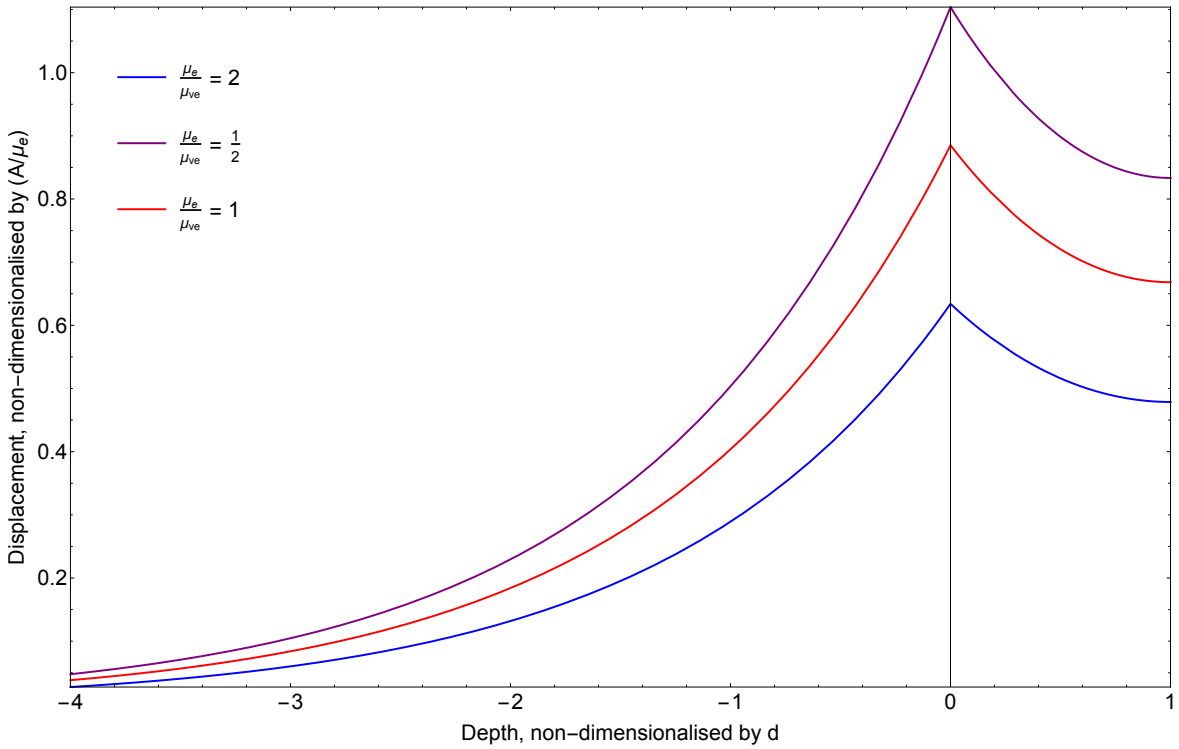


Figure 2.12: Displacement due to viscous relaxation as $t \to \infty$ for elastic layer over viscoelastic half space with $w = 8$, taken as vertical slice at $x = 2$ for maximum displacement. Contrasting shear modulus, total displacement due to viscous relaxation

factor of $1/s$) I can calculate the solution to the case of a viscoelastic half-space with

$$\mu \to \frac{s}{\mu^{-1}s + \eta^{-1}}, \tag{2.58}$$

and

$$\begin{aligned} u(x,y,t) &= \mathcal{L}^{-1}\left\{\frac{A(\mu^{-1}s+\eta^{-1})}{ks^2}e^{ky}\sin(kx)\right\}, \\ &= \frac{A}{k}e^{ky}\sin(kx)\,\mathcal{L}^{-1}\left\{\frac{\mu^{-1}s+\eta^{-1}}{s^2}\right\}, \\ &= \frac{A}{k}e^{ky}\sin(kx)\left(\frac{1}{\mu}\mathcal{L}^{-1}\left\{\frac{1}{s}\right\}+\frac{1}{\eta}\mathcal{L}^{-1}\left\{\frac{1}{s^2}\right\}\right), \\ &= \frac{A}{k}H(t)e^{ky}\sin(kx)\left(\frac{1}{\mu}+\frac{t}{\eta}\right). \end{aligned} \tag{2.59a}$$

2.6.7 Analysis of solution to elastic layer over viscoelastic half-space

Once the equations for the elastic layer over the viscoelastic half-space have been derived, equations (2.54), the limiting cases can be examined. Firstly, the elastic response at $t = 0$

$$u_e(x, y, 0) = \frac{A \sin(kx) (\mu_e \cosh(ky) + \mu_{ve} \sinh(ky))}{k\mu_e(\mu_e \sinh(kd) + \mu_{ve} \cosh(kd))}, \qquad 0 < y < d, \tag{2.60a}$$

$$u_{ve}(x, y, 0) = \frac{Ae^{ky} \sin(kx)}{k(\mu_e \sinh(kd) + \mu_{ve} \cosh(kd))}, \qquad y < 0, \tag{2.60b}$$

which reproduces the solution for the case of an elastic lid over an elastic half-space, equation (2.37), with $\mu_+ \to \mu_e$ and $\mu_- \to \mu_{ve}$. If the substitutions $\mu_e = \mu$ and $\mu_{ve} = \mu$ are made then we recover a variant of equation (2.29) with a translation $u(x, y + d)$,

$$u(x, y) = \frac{A}{k\mu} e^{k(y-d)} \sin(kx). \tag{2.61}$$

Next, I calculate the fully relaxed solution, with $t \to \infty$,

$$\lim_{t\to\infty}[u_e(x, y, t)] = \frac{A \sin(kx) \cosh(ky)}{k\mu_e \sinh(kd)}, \qquad 0 < y < d, \tag{2.62a}$$

$$\lim_{t\to\infty}[u_{ve}(x, y, t)] = \frac{Ae^{ky} \sin(kx)}{k\mu_e \sinh(kd)}, \qquad y < 0, \tag{2.62b}$$

where the solution for u_e as $t \to \infty$ matches the solution for the zero basal traction, equation (2.57). I can also calculate the case of the viscoelastic half-space from the limit of u_{ve} as $d \to 0$,

$$\lim_{d\to 0}[u_{ve}(x, y, t)] = \frac{A}{k} H(t) e^{ky} \sin(kx) \left(\frac{1}{\mu_{ve}} + \frac{t}{\eta}\right), \qquad y < 0, \tag{2.63}$$

where the solution for u_{ve} as $d \to 0$ given in equation (2.63) matches the solution for a viscoelastic half-space, equation (2.59a). The relaxation time can also be extracted

from equation (2.54),

$$t_R = \frac{\eta}{\mu_e \mu_{ve}}(\mu_e + \mu_{ve} \coth(kd)). \tag{2.64}$$

For the case of constant shear modulus ($\mu_e = \mu_{ve}$), the relaxation time reduces to match that in equation (2.47),

$$t_R = \frac{\eta}{\mu}(1 + \coth(kd)). \tag{2.65}$$

Alternatively, these can be expressed in terms of wavelength rather than wavenumber with $k = 2\pi/w$,

$$t_R = \frac{\eta}{\mu}\left(1 + \coth\left(\frac{2\pi d}{w}\right)\right). \tag{2.66}$$

To investigate the limiting behaviour of equation 2.66, illustrated in Figure 2.13, I

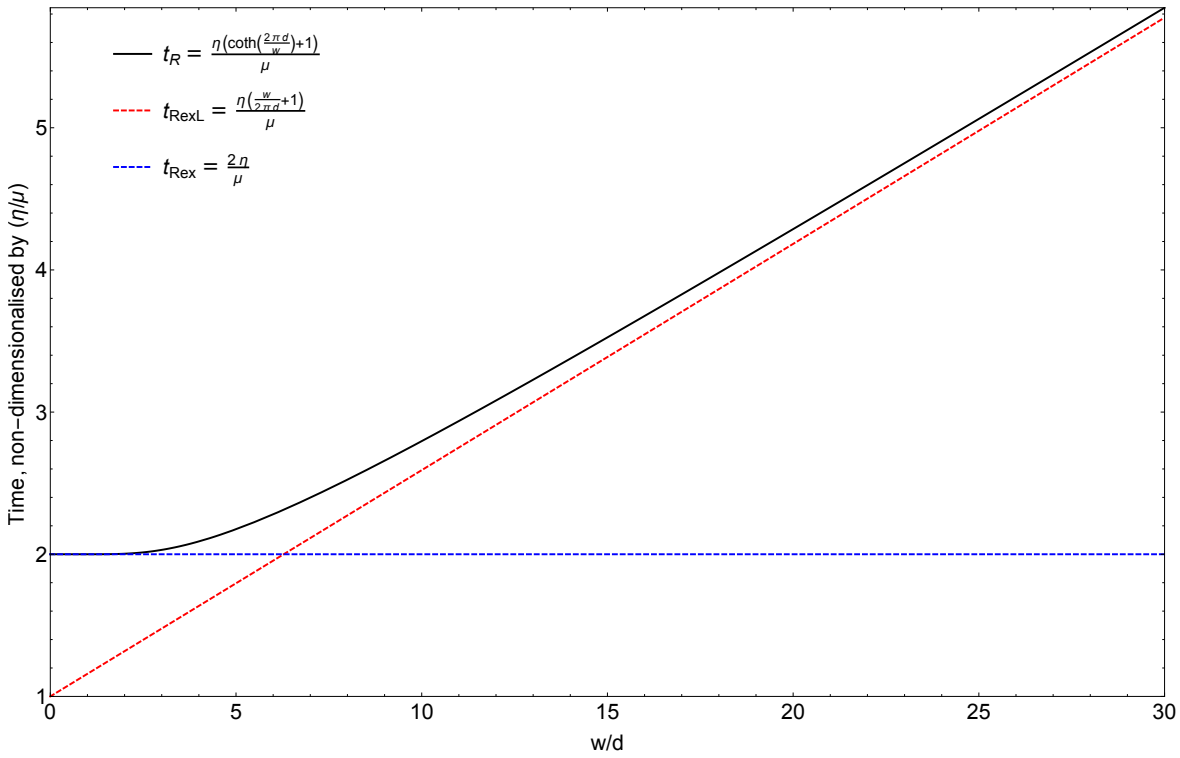


Figure 2.13: Time constant t_R for constant shear modulus as a function of wavelength over elastic lid thickness, (w/d), for shear stress condition $\sigma_{yz}(x, d) = A\sin(\frac{2\pi x}{w})$.

examine the series expansion of t_R about $w = 0$ and as $w \to \infty$. For the former, I

require the Taylor expansion of t_R about $w = 0$,

$$\begin{aligned} t_{Rex} &= \lim_{w\to 0}\left[\frac{\eta}{\mu}\left(\coth\left(\frac{2\pi d}{w}\right)+1\right)\right], \\ &= \frac{\eta}{\mu}\left(1+1\right), \\ &= \frac{2\eta}{\mu}. \end{aligned} \tag{2.67}$$

For the latter limiting case of $w \to \infty$, this is strictly equivalent to taking the limit

$$\lim_{w\to\infty}\left[1+\coth\left(\frac{1}{w}\right)\right] \equiv \lim_{x\to 0}\left[1+\coth\left(x\right)\right]. \tag{2.68}$$

Therefore, to examine the behaviour of the function t_R in this region I require the Laurent expansion of $\coth(x)$,

$$\coth x = x^{-1} + \frac{x}{3} - \frac{x^3}{45} + \frac{2x^5}{945} + \cdots = x^{-1} + \sum_{n=1}^{\infty}\frac{2^{2n}B_{2n}x^{2n-1}}{(2n)!}. \tag{2.69}$$

In the limiting case of $x \to 0$, only the first term of equation (2.69) does not tend to zero, so the Laurent expansion of t_R as $w \to \infty$ is

$$t_{RexL} = \frac{\eta}{\mu}\left(1+\frac{w}{2\pi d}\right). \tag{2.70}$$

2.6.8 Analytic solution for periodic displacement condition at the surface

In this section, I investigate the related problem of an enforced displacement condition at the surface at $t = 0$. The setup of the problem is similar, but the new boundary conditions are equations (2.28) and

$$u_+(x,d) = A\sin\left(kx\right), \tag{2.71}$$

with interface conditions from equation (2.32). The general solutions are the same as in equation (2.33) with equations to match the boundary conditions,

$$\delta = 0, \tag{2.72a}$$

$$\alpha + \beta = \gamma, \tag{2.72b}$$

$$\gamma\mu_- + \beta\mu_+ - \alpha\mu_+ = 0, \tag{2.72c}$$

$$\alpha e^{dk} + \beta e^{-dk} = A. \tag{2.72d}$$

The coefficients that satisfy equations (2.72) are

$$\alpha = \frac{A(\mu_- + \mu_+)}{2\mu_+ \cosh(kd) + 2\mu_- \sinh(kd)}, \tag{2.73a}$$

$$\beta = \frac{A(-\mu_- + \mu_+)}{2\mu_+ \cosh(kd) + 2\mu_- \sinh(kd)}, \tag{2.73b}$$

$$\gamma = \frac{A\mu_+}{\mu_+ \cosh(kd) + \mu_- \sinh(kd)}, \tag{2.73c}$$

which gives the solution for an elastic layer over an elastic half-space with contrasting shear moduli

$$u_+(x,y) = \frac{A\sin(kx)(\mu_- \sinh(ky) + \mu_+ \cosh(ky))}{\mu_+ \cosh(kd) + \mu_- \sinh(kd)}, \qquad 0 < y < d, \tag{2.74a}$$

$$u_-(x,y) = \frac{Ae^{k(d+y)} \sin(kx)\mu_+}{-\mu_- + \mu_+ + e^{2kd}(\mu_- + \mu_+)}, \qquad y < 0. \tag{2.74b}$$

If I substitute $\mu_+ \to \mu$ and equation (2.40) into the elastic solution, equation (2.74), I have the solution in the inverse Laplace domain to the corresponding viscoelastic problem which can be inverted to give the solution to an elastic layer over a viscoelastic

half-space,

$$u_e(x,y,t) = H(t)Ae^{-kd}\sin(kx)\Bigg(\cosh(ky)(1+\tanh(kd)) - \operatorname{sech}(kd)\sinh(k(d-y))e^{-\frac{t}{t_R}}\Bigg), \qquad 0<y<d, \tag{2.75a}$$

$$u_{ve}(x,y,t) = H(t)Ae^{k(y-d)}\sin(kx)\left(1+\tanh(kd)-\tanh(kd)e^{-\frac{t}{t_R}}\right),\ y<0, \tag{2.75b}$$

$$t_R = \frac{\eta}{\mu}(1+\tanh(kd)), \tag{2.75c}$$

$$t_R = \frac{\eta}{\mu}\left(1+\tanh\left(\frac{2\pi d}{w}\right)\right). \tag{2.75d}$$

This time constant t_R in equation (2.75d) is plotted in Figure 2.14, with data points

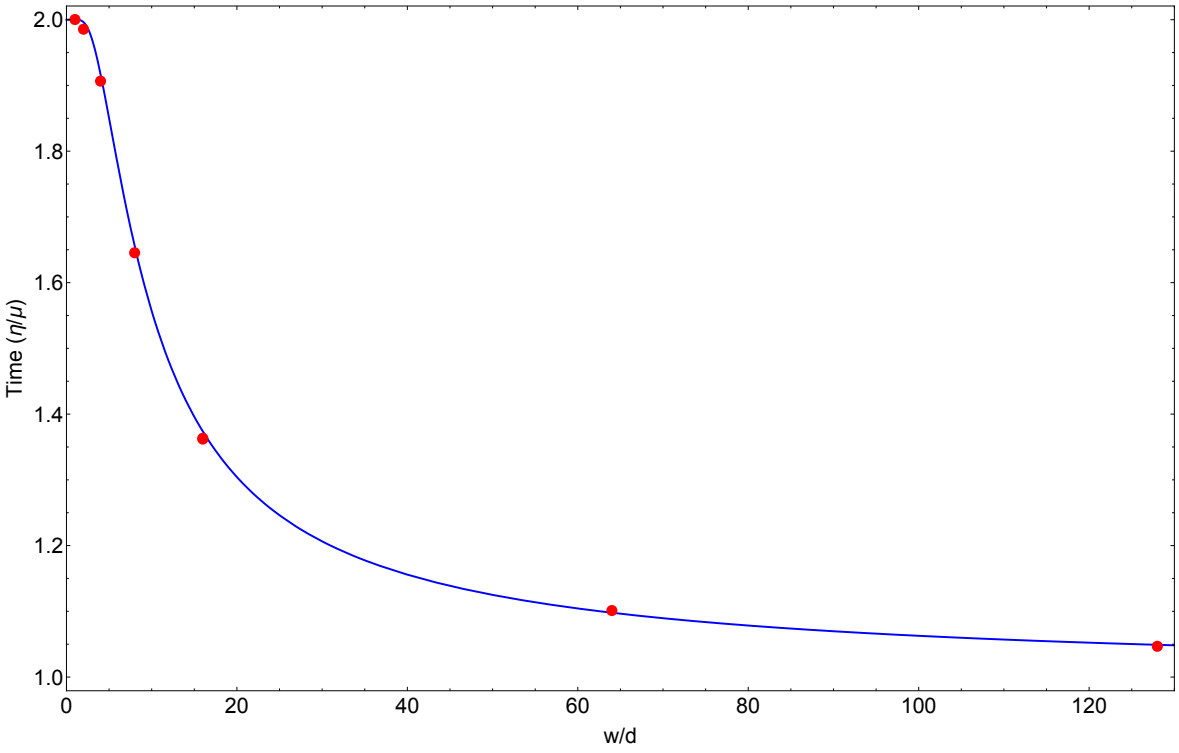


Figure 2.14: Time constant t_R for constant shear modulus as a function of wavelength over elastic lid thickness $\left(\frac{w}{d}\right)$, for imposed displacement $u_e(x,d) = A\sin(\frac{2\pi x}{w})$. Blue line is t_R from (2.78c) with red points simulations run in *oregano_ve*.

from *oregano_ve* simulations, and the displacements are plotted in Figure 2.15. The numerical model is constructed with an elastic layer ($\mu = 1, \eta = 10^{20}$ Pa s) of thickness

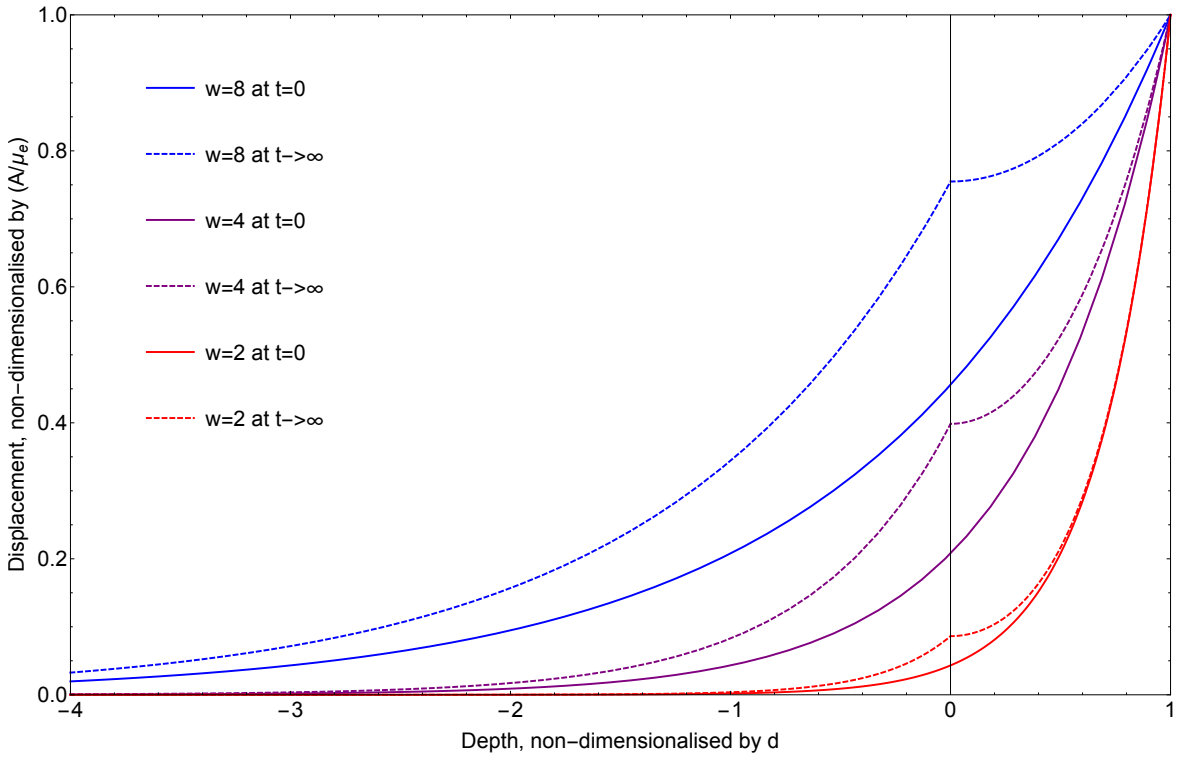


Figure 2.15: Solution for elastic layer over viscoelastic half space with imposed surface displacement condition, taken as a vertical slice at $x = \frac{w}{4}$ for maximum displacement. This plot shows total displacement for a constant shear modulus and variable wavelength.

1 over a viscoelastic layer ($\mu = 1, \eta = 1$) of thickness 19 units to approximate an infinite half-space. The simulations are run with zero displacement boundary conditions on the sides and base, and equation (2.71) at the top. To obtain the time constant t_R in the simulations, the point $u(w/4, 0, t)$ is tracked, Figure 2.16. The resulting N pairs of data points (t, u) are processed to be plotted as follows, where u_0 is the first data point and u_N the last, assuming $u_N \approx \lim_{t\to\infty} u(t)$,

$$\frac{t}{t_R} \approx -\log\left[1 - \frac{u - u_0}{u_N - u_0}\right]. \tag{2.76}$$

The data is processed using equation (2.76) and plotted against t in Figure 2.17, the gradient of which is t_R. Each calculated relaxation timescale t_R is included with the theoretical curve in Figure 2.14.

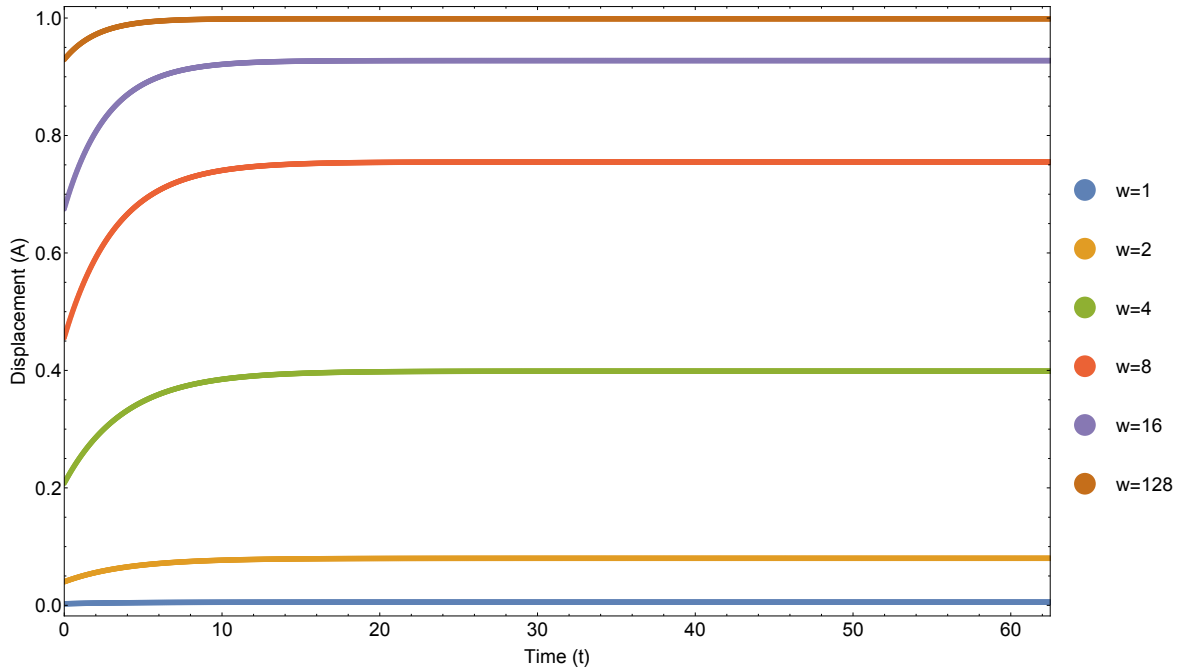


Figure 2.16: Time series of viscoelastic relaxation for imposed periodic displacement at surface, tracking point $u(w/4, 0)$, elastic layer over viscoelastic half-space

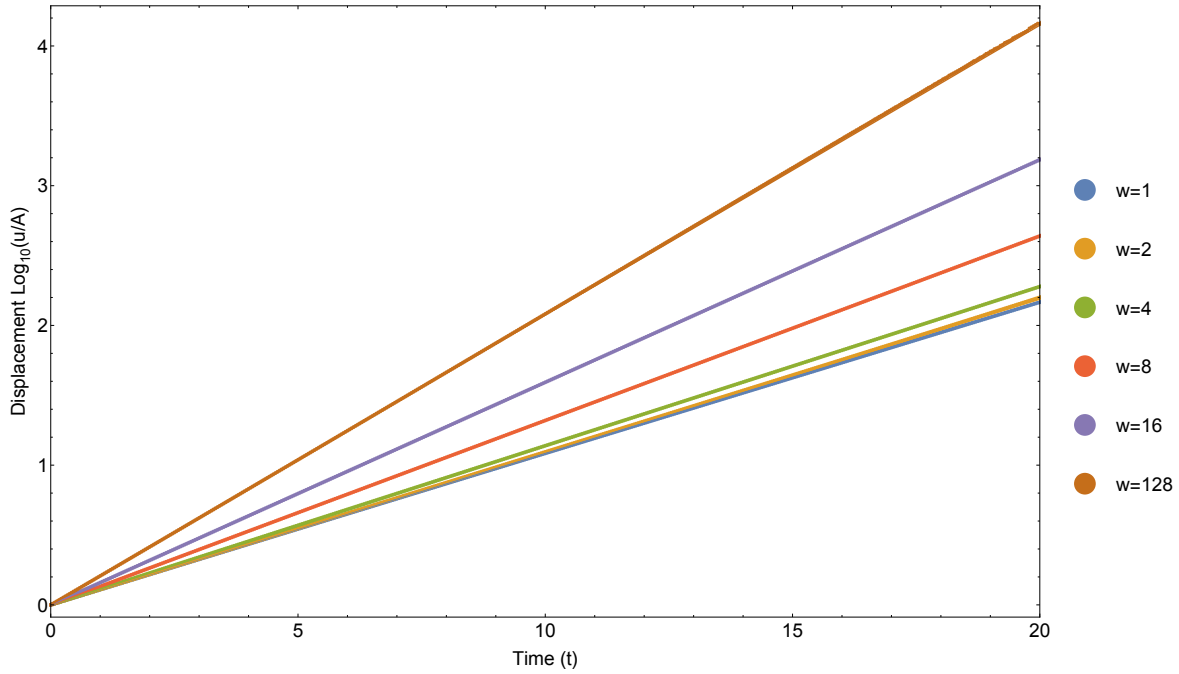


Figure 2.17: Time series of viscoelastic relaxation for imposed periodic displacement at surface, tracking point $u(w/4, 0)$, data processed using equation (2.76).

2.6.9 Discussion

The key results are summarised below,

For the shear stress surface conditions, displacements and the relaxation time are given by

$$u_e(x,y,t) = \frac{H(t)Ae^{-kd}\sin(kx)}{k\mu}\Bigg(\cosh(ky)(1+\coth(kd)) - \cosh(k(d-y))\operatorname{csch}(kd)e^{-\frac{t}{t_R}}\Bigg), \qquad 0<y<d, \tag{2.77a}$$

$$u_{ve}(x,y,t) = \frac{H(t)Ae^{k(y-d)}\sin(kx)}{k\mu}\left[(1+\coth(kd)) - \coth(kd)e^{-\frac{t}{t_R}}\right], y<0, \tag{2.77b}$$

$$t_R = \frac{\eta}{\mu}(1+\coth(kd)). \tag{2.77c}$$

Alternatively, for a displacement surface condition the displacements and the relaxation time are given by

$$u_e(x,y,t) = H(t)Ae^{-kd}\sin(kx)\Bigg(\cosh(ky)(1+\tanh(kd)) - \operatorname{sech}(kd)\sinh(k(d-y))e^{-\frac{t}{t_R}}\Bigg), \qquad 0<y<d, \tag{2.78a}$$

$$u_{ve}(x,y,t) = H(t)Ae^{k(y-d)}\sin(kx)\left(1+\tanh(kd)-\tanh(kd)e^{-\frac{t}{t_R}}\right), y<0, \tag{2.78b}$$

$$t_R = \frac{\eta}{\mu}(1+\tanh(kd)). \tag{2.78c}$$

In both of these cases, the $t = 0$ case recovers the purely elastic solution as required. The fully relaxed solutions in the limit as $t \to \infty$ can also be calculated:

Shear stress surface condition

$$\lim_{t\to\infty}[u_e(x,y,t)] = \frac{A\sin(kx)\cosh(ky)}{k\mu_e\sinh(kd)} \qquad 0<y<d \tag{2.79a}$$

$$\lim_{t\to\infty}[u_{ve}(x,y,t)] = \frac{Ae^{ky}\sin(kx)}{k\mu_e\sinh(kd)} \qquad y<0 \tag{2.79b}$$

Displacement surface condition

$$\lim_{t\to\infty}[u_e(x,y,t)] = A\sin(kx)\cosh(ky)\,\mathrm{sech}(kd) \qquad 0 < y < d \tag{2.80a}$$

$$\lim_{t\to\infty}[u_{ve}(x,y,t)] = Ae^{ky}\sin(kx)\,\mathrm{sech}(kd) \qquad y < 0 \tag{2.80b}$$

This is a novel result in that the effective relaxation time for a Maxwell rheology under these conditions is a function of the wavenumber. If we analyse t_R for shear stress surface condition (equation (2.77c)) with a substitution to convert wavenumbers to wavelengths ($k = 2\pi/w$), it has two asymptotes. For small ratios of the wavelength to the elastic lid thickness w/d, the effective relaxation time behaves as a constant $2\eta/\mu$, but as that ratio increases it approaches a linear function $\eta/\mu(1 + w/(2\pi d))$, shown in Figure 2.13. Physically this would mean that components of stress with relatively short wavelengths will decay with the classic maxwell relaxation time, but longer wavelength stress fields can persist for much greater lengths of time.

This effective relaxation time can be used to test our finite element code *oregano_ve* by tracking a point on the interface between the two media, as shown in Figure 2.16. Manipulation of the fully relaxed solutions, equations (2.79b and 2.80b), and elastic solutions, equations (2.77c and 2.78c with $t = 0$), allows us to directly obtain the time constant t_R from the simulations. The time constant t_R from equation (2.78c) is plotted in figure 2.14, and the data points from *oregano_ve* simulations are in agreement.

The assumption that $u_N \approx \lim_{t\to\infty} u(t)$ is ultimately very important; initially the simulations were run for 500 time steps, however when processing the data it becomes apparent that small deviations from the fully relaxed solution resulted in large departures from linearity with increasing t. Increasing the number of time steps to 1500 minimised this effect, though approximate fits can be obtained using small t values for both cases.

2.7 Conclusions

I have taken the existing finite element code *oregano_ve* and modified it to include periodic boundary conditions, so that I may investigate anti-plane problems. Primarily I am concerned with deformation associated with an infinitely long strike-slip fault, as there exist a number of analytical solutions (*Savage et al.* (1978) and others) that can aid in understanding more complex earthquake cycle problems, such as interpreting the effect of depth-dependent and non-linear rheologies. Once the code was modified, I used a suite of benchmarking tests to ensure it was operating correctly. I first compared it to the solution for post-seismic relaxation of a strike-slip fault presented in *Nur and Mavko* (1974), where I found the inverse model to be in agreement with the input parameters within the resolution of the grid spacing.

To benchmark the implementation of depth dependent viscosity in *oregano_ve* I have derived a series of novel analytical solutions for simple shear of a rectangular block with viscosity structures to match those in *oregano_ve*, including one for a power-law rheology. For all the linear rheologies with depth dependence, *oregano_ve* behaved as predicted; however, the initial results for the non-linear benchmark were not in agreement. This discrepancy arose due to two definitions used for the second invariant of the strain-rate tensor in the code by different collaborators. My benchmark allowed us to identify this bug and correct it.

Finally, I present a series of novel results for viscoelastic relaxation of imposed harmonic tractions and displacements. I examined the effect of a harmonic shear stress or displacement condition at the surface of a 2D plane, with motions assumed to be anti-plane. I found the time constant associated with the relaxation was dependent of the wavelength of the forcing stress or displacement condition (though the two cases did not behave in the same fashion). Short wavelength components of imposed shear stresses relax more quickly than longer wavelength components and have smaller maximum amplitude. Physically this would mean that components of stress with relatively

short wavelengths will decay with the classic maxwell relaxation time, but longer wavelength stress fields can persist for much greater lengths of time. The reverse is true for displacement conditions, where longer wavelength displacements decay more rapidly. My interpretation of this dichotomy is that the stresses for short wavelength components diffuse outwards more rapidly than longer wavelength components, whilst short wavelength displacements are filtered by the elastic lid and are therefore restricted in how rapidly they can diffuse, until $w \gg T_e$. Multiple mechanisms have been involved in the literature to explain the observed variability in decay of short versus long wavelength signals (*Hetland and Hager* (2005, 2006); *Barbot et al.* (2008); *Yamasaki and Houseman* (2012b)) and this study shows that multiple timescales may be involved with a relatively simple model of an elastic lid over a maxwell viscoelastic half space. Finally, these results could be used as the basis for a Fourier decomposition of an imposed stress or displacement field.

Chapter 3

Scaling of viscous shear zones with depth dependant viscosity and power law stress-strain dependence

This chapter represents a version of an article, *Moore and Parsons* (2015)[1].

3.1 Introduction

We observe faults as quasi-planar structures at the Earth's surface, but what governs the distribution of shear at depth? Geologic observations of exhumed shear zones (*Poirier* (1980), *White et al.* (1980), *Rutter* (1999), and *Norris and Cooper* (2003)) indicate that whilst the motion is no longer planar, it can still be localised near the down-dip extension of the fault; however, the degree of localisation is still uncertain (*Bürgmann and Dresen* (2008), *Wilson et al.* (2004)). In this chapter, I am interested in the structural form of distributed shear beneath a strike-slip fault and the physical mechanisms that have the potential to localise a shear zone. Previous work has focused on the effect of thermomechanical coupling. *Yuen et al.* (1978) used a one-dimensional (1D) model and found that once a shear zone forms it will remain localised due to

[1]All work was conducted by J.Moore, under guidance from B. Parsons

shear-stress heating. *Thatcher and England* (1998) expanded upon this by looking at the effect of shear heating on a two-dimensional (2D) idealised strike-slip fault in an elastic lid over a linear, Newtonian, viscous layer. They found that, whilst a broad range of behaviour could be simulated, for reasonable parameter choices shear zones are narrow. The key conclusion was that shear localisation is driven by dissipative heating near the axis of the shear zone, causing a localised reduction in viscosity. *Takeuchi and Fialko* (2012) used a three-dimensional (3D) model with periodic boundary conditions to simulate the 2D case of an rigid lid over a viscoelastic layer and concluded that a thermally-coupled non-linear power law rheology could also generate such localisation. They found a maximum thermal anomaly of 240°C, in agreement with *Thatcher and England* (1998), capable of reducing the viscosity by up to 2.5 orders of magnitude in the region immediately beneath the fault tip. More recently, *Duretz et al.* (2014) modelled the development of shear zones in a 2D viscous sheet undergoing compression. They concluded that the degree of shear zone localisation depends both on thermomechanical coupling and a non-linear rheology, and predict half-widths of 1 - 7km with typical temperature increases of $50 - 200$°C due to thermomechanical coupling for crustal shear zones.

The degree of localisation of shear at depth and the effective viscosity structure beneath the fault has important implications for the interpretation of geodetic observations of surface deformation throughout the earthquake cycle. If shear is localised at depth, then the region of applicability for elastic half-space models of fault creep at depth (*Savage and Burford* (1973)) is greater than previously thought. A localised shear zone would allow inter-seismic strain accumulation resembling the well-known arctangent distribution of surface displacements. The development of this viscosity structure may also explain the relatively low apparent viscosities in the vicinity of a fault required to fit post-seismic data (*Yamasaki et al.* (2014)).

In this chapter I investigate viscous shear zones and examine the interplay between

three controls on the localisation of shear in a viscous shear zone: the depth dependence of the viscosity, shear-stress heating, and a power law stress-strain rate relation. I start from the model used to investigate ductile shear zones in *Thatcher and England* (1998). Scaling laws are derived which can be used to infer shear zone widths for a given viscosity structure and I show that shear zone localisation can be produced by a linear rheology with a depth dependent viscosity (DDV) alone.

3.2 Model

The model construction is that of an idealised strike-slip fault, infinite in length, embedded in an rigid lid over a viscous layer, as presented in *Thatcher and England* (1998). I treat this as a two-dimensional (2D) anti-plane problem ($\dot{\epsilon}_{xx} = \dot{\epsilon}_{yy} = \dot{\epsilon}_{zz} = \dot{\epsilon}_{xy} = 0$, where $\dot{\epsilon}_{ij}$ is the strain rate tensor), so the region of interest becomes a vertical slice taken perpendicular to the fault (Figure 3.1) with all velocities assumed to be orthogonal to the plane and a function of position, $u(x, z)$. As the problem is antisymmetric about $x = 0$, I need only analyse one half of the domain. Unless otherwise stated, all length scales are non-dimensionalised by the thickness of the viscous layer (d) and velocities by the driving velocity (v_0). The direction z is defined positive downwards, so the domain of u is ($0 \leq x \leq \omega$, $0 \leq z \leq 1$), with range $0 \leq u \leq 1$ and aspect ratio, $\omega = x_{max}/d$.

The main assumption of this model is that the system is fully relaxed beneath the rigid lid, such that I can neglect the effect of elasticity in this region, modelling it as a purely viscous process. This assumption is valid as long as the effective viscosities are low enough in the viscous layer that the relaxation timescale for a viscoelastic material, η/μ is less than the earthquake recurrence interval. Motion is driven by fixed far-field velocities, and the overlying lid is assumed to have block motion in a time-averaged earthquake cycle. The base of the layer is assumed to be shear-stress free, and motion

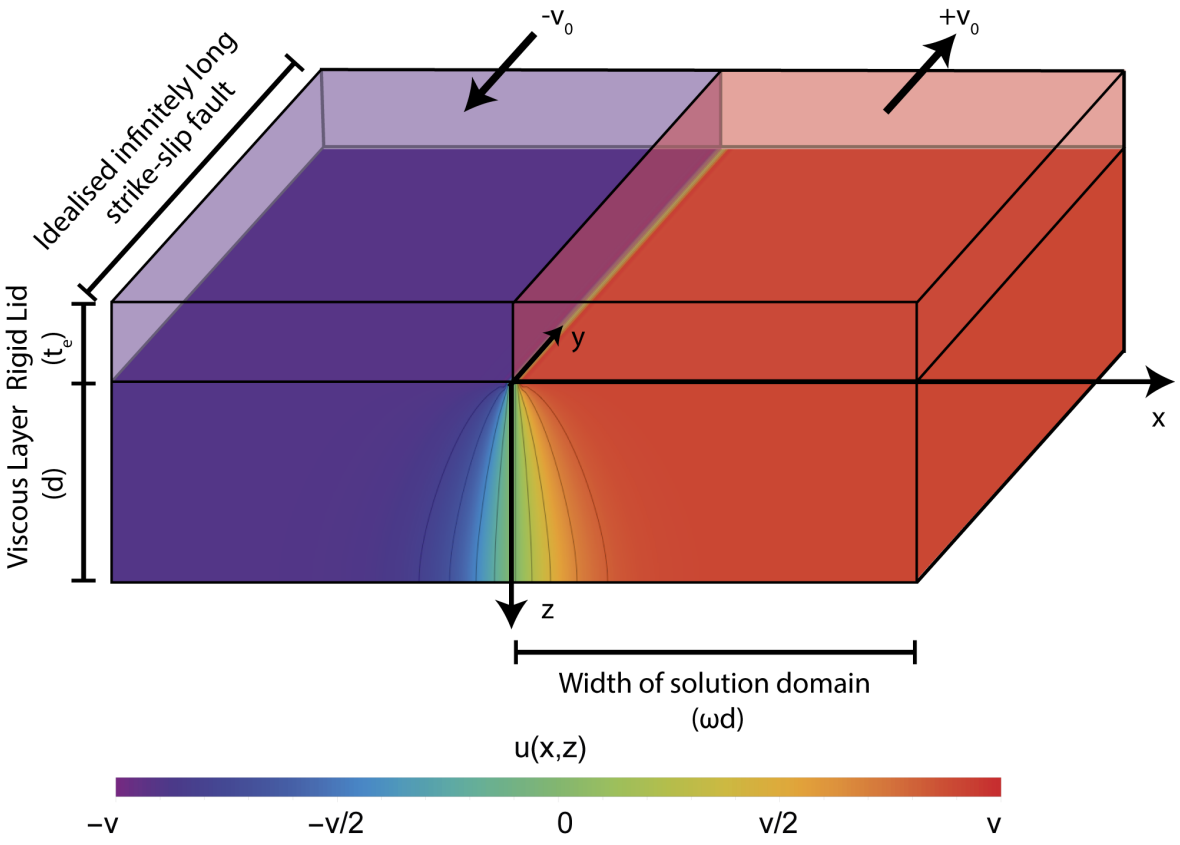


Figure 3.1: Schematic of the shear zone model: strike slip fault in a rigid lid over a viscous layer

in the viscous layer is governed by the force balance equation

$$\frac{\partial \tau_{ji}}{\partial x_j} = -\rho g_i, \tag{3.1}$$

with a general viscous rheology (viscosity η)

$$\tau_{ij} = \eta(x, z)\dot{\epsilon}_{ij}, \tag{3.2}$$

which for a 2D anti-plane problem (neglecting body force terms) reduces to

$$\nabla^2 u = -\left(\frac{\partial \ln(\eta)}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial \ln(\eta)}{\partial z}\frac{\partial u}{\partial z}\right). \tag{3.3}$$

At this stage equation (3.3) is still general, but may be analysed further once a specific viscous rheology is selected (see Table 3.1). The boundary conditions are

$$u(0, z) = 0, \tag{3.4a}$$

$$u(\omega, z) = 1, \tag{3.4b}$$

$$u(x, 0) = 1, \tag{3.4c}$$

$$\left.\frac{\partial u}{\partial z}\right|_{z=1} = 0. \tag{3.4d}$$

We also investigated the second model from *Thatcher and England* (1998), in which the lid is allowed to be passively displaced (shear stress free on the $z = 0$ surface), but it may be shown that in the absence of shear-stress heating that the velocity field is a linear ramp ($u(x) = x/\omega$) for all choices of rheology. However, even with shear stress heating, for a wide range of parameters the shear zones widen to form a broad region of distributed deformation, as noted in *Thatcher and England* (1998). Thus I did not develop that model further and instead concentrated on the model presented above. Finally, to aid direct comparison between the results of our models, a definition of shear zone width (δ_w) is required; I use the half-width at the base of the layer, $u(\delta_w, z = 1) = 1/2$.

	Uniform properties	Depth dependent viscosity	Arrhenius law
Linear:	$\eta = \eta_0$	$\eta_D = \eta_0 \exp\left(-\frac{z}{z_0}\right)$	$\eta_A = \frac{T}{2B} \exp\left(\frac{Q}{RT}\right)$
Non-linear:	$\eta_n = \frac{1}{B} J_2^{\frac{1}{n}-1}$	$\eta_{Dn} = \frac{1}{B} \exp\left(-\frac{z}{z_0}\right) J_2^{\frac{1}{n}-1}$	$\eta_{An} = \frac{T}{2B} \exp\left(\frac{Q}{RT}\right) J_2^{\frac{1}{n}-1}$

Table 3.1: Viscosity structures, where $J_2 = 2^{-\frac{1}{2}} \sqrt{(\partial_x u)^2 + (\partial_z u)^2}$. Arrhenius law reproduced from *Thatcher and England* (1998) after *Yuen et al.* (1978) where the factor T is introduced to provide an approximation for nonlinear creep processes and grain size variation. Rheological parameters used in the calculations are taken from *Hirth and Kohlstedt* (2003); *Hirth et al.* (2001) (with appropriate unit adjustments).

3.3 Linear ductile shear zones

3.3.1 Uniform properties

For reference, I first investigate the case of a constant viscosity ($\eta = \eta_0$). As there are no gradients in viscosity, equation (3.3) reduces to solving Laplace's equation, $\nabla^2 u = 0$, on a rectangular domain. This has a Fourier series solution,

$$u(x,z) = \frac{x}{\omega} + \frac{2}{\pi}\sum_{m=1}^{\infty}\frac{1}{m}\frac{\cosh\left(m\pi(1-z)/\omega\right)}{\cosh\left(m\pi/\omega\right)}\sin\Big(\frac{m\pi x}{\omega}\Big), \tag{3.5}$$

and is plotted for $\omega = 4$ in Figure 3.2. This solution is pointwise convergent, exhibiting the Gibbs phenomenon at the point $x = 0$, $z = 0$ due to the step discontinuity. The shear zone half-width for a constant viscosity layer, $\delta_w = 0.5611$, can be used to directly compare results for different rheologies and provides a maximum width of the shear zone. For $\omega \gg 1$, the shear zone width should be independent of ω, governed only by the thickness of the viscous layer, d. As the solution is a Fourier series it cannot be directly inverted to calculate δ_w. However, I can investigate the behaviour of the limiting case of $\omega \to \infty$ by numerically calculating δ_w as a function of ω. The Fourier series was evaluated with 50,000 terms and $\delta_w(\omega)$ calculated for $\omega \in [1, 4000]$ using a numerical minimisation routine in Mathematica (*Wolfram Research* (2012)). The limiting behaviour is illustrated in Figure 3.3, and for comparison with the numerical models I select $\omega = 4$ to give $\delta_w = 0.5611$ with precision of 4 significant figures (4sf).

3.3.2 Exponentially depth dependent viscosity

The next rheology I investigate is that of an exponentially depth dependent viscosity; as I show later, this is approximately what arises from the geothermal gradient with a temperature dependent viscosity. Substituting the depth dependent linear viscosity structure (η_D) from Table 3.1 into equation (3.3) gives

$$\nabla^2 u = \frac{1}{z_0}\frac{\partial u}{\partial z}. \tag{3.6}$$

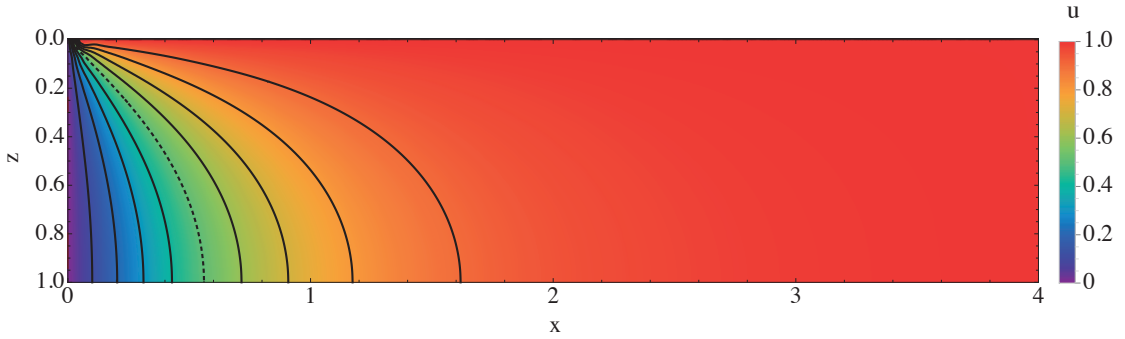


Figure 3.2: Constant viscosity analytical solution, equation (3.5). Contours are plotted at 10% intervals of the driving velocity, with a dashed contour for 50%, the aspect ratio $\omega = 4$ and the shear zone half-width $\delta_w = 0.561$.

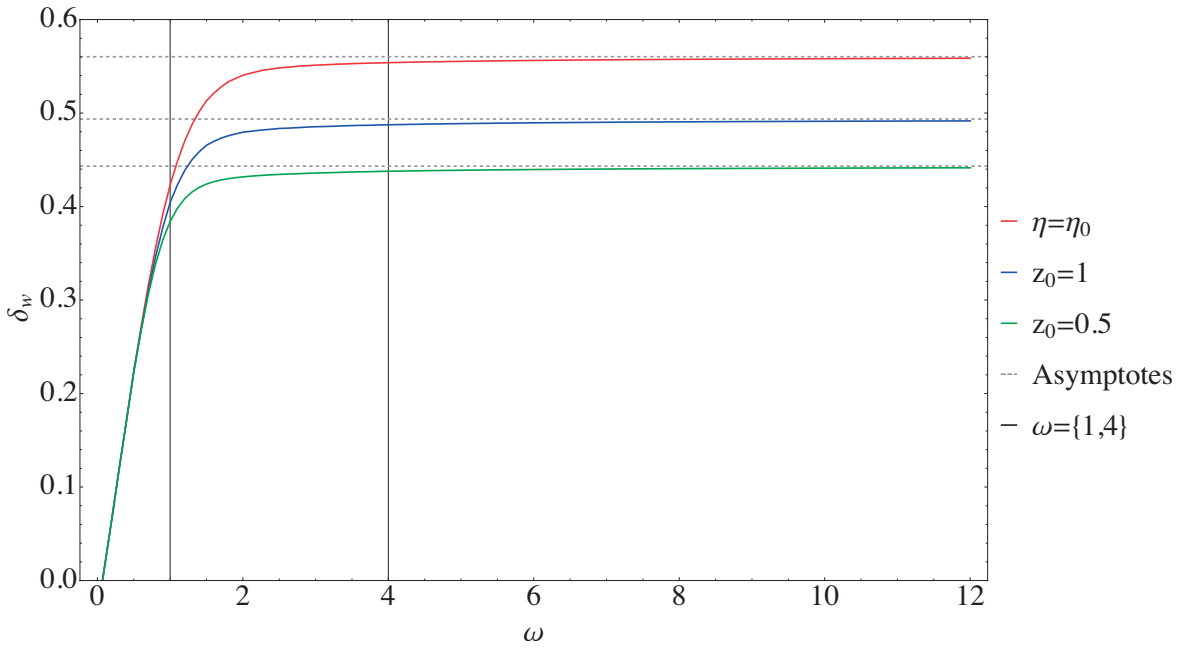


Figure 3.3: Sensitivity of solution to aspect ratio for constant viscosity (red line) and depth dependent viscosity (green and blue lines). Dashed lines are numerically derived asymptotes for $\omega \to \infty$ and vertical gridlines are included for $\omega = 1$ and $\omega = 4$.

An analytical solution may be obtained for the depth dependent viscosity by first removing a linear ramp,

$$u(x, z) = \frac{x}{\omega} + \tilde{u}(x, z), \tag{3.7}$$

and solving

$$\nabla^2 \tilde{u}(x,z) = \frac{1}{z_0}\frac{\partial \tilde{u}(x,z)}{\partial z}, \tag{3.8}$$

for $\tilde{u}(x,z)$ with the modified boundary conditions

$$\tilde{u}(0,z) = 0, \tag{3.9a}$$

$$\tilde{u}(\omega,z) = 0, \tag{3.9b}$$

$$\tilde{u}(x,0) = 1 - x/\omega, \tag{3.9c}$$

$$\frac{\partial \tilde{u}(x,1)}{\partial z} = 0. \tag{3.9d}$$

I then utilise the method of separation of variables to construct a Fourier series solution assuming

$$\tilde{u}(x,z) = X(x)Z(z). \tag{3.10}$$

Substituting $\tilde{u}(x,z)$ from equation (3.10) into equation (3.8) and rearranging gives,

$$\frac{X''(x)}{X(x)} = \frac{\frac{1}{z_0}Z'(z) - Z''(z)}{Z(z)} = -\lambda, \tag{3.11}$$

where λ must be constant. These produce separate equations for $X(x)$ and $Z(z)$,

$$X''(x) + \lambda X(x) = 0, \tag{3.12}$$

which admits solutions

$$X(x) = C_1 \sin\left(\sqrt{\lambda}x\right) + C_2 \cos\left(\sqrt{\lambda}x\right), \tag{3.13}$$

and

$$Z''(z) - \frac{1}{z_0}Z'(z) - \lambda Z(z) = 0, \tag{3.14}$$

which admits solutions

$$Z(z) = C_3 \exp\left(\frac{z(1-\sqrt{1+4z_0^2\lambda})}{2z_0}\right) + C_4 \exp\left(\frac{z(1+\sqrt{1+4z_0^2\lambda})}{2z_0}\right). \tag{3.15}$$

where λ, C_1, C_2, C_3 and C_4 are constants determined by the boundary conditions, equations (3.9a) to (3.9d). To satisfy boundary conditions (3.9a) and (3.9b), $C_2 = 0$, $C_1 = a_m$ and

$$\lambda = \frac{m^2\pi^2}{\omega^2}. \tag{3.16}$$

Thus equations (3.13) and (3.15) simplify to

$$X_m(x) = a_m \sin\left(\frac{m\pi x}{\omega}\right), \tag{3.17a}$$

$$Z_m(z) = C_3 \exp\left(\frac{z}{2z_0} - \frac{z\sqrt{4m^2\pi^2z_0^2+\omega^2}}{2\omega z_0}\right) + C_4 \exp\left(\frac{z}{2z_0} + \frac{z\sqrt{4m^2\pi^2z_0^2+\omega^2}}{2\omega z_0}\right). \tag{3.17b}$$

As the boundary condition (3.9c) is dependent on x, I solve for constants C_3 and C_4 from equation (3.17b) using boundary conditions $Z(0) = 1$ and $Z'(1) = 0$ giving $C_4 = 1 - C_3$ and

$$C_3 = \frac{2\pi^2m^2z_0^2 e^{\frac{\sqrt{4\pi^2m^2z_0^2+\omega^2}}{\omega z_0}}}{2\pi^2m^2z_0^2\left(e^{\frac{\sqrt{4\pi^2m^2z_0^2+\omega^2}}{\omega z_0}}+1\right) + \omega\left(\omega - \sqrt{4\pi^2m^2z_0^2+\omega^2}\right)}. \tag{3.18}$$

Substituting C_3 and C_4 into equation (3.17b) gives

$$Z_m(z) = e^{\frac{z}{2z_0}}\left(\frac{\alpha\cosh\left(\frac{\alpha(1-z)}{2z_0\omega}\right) + \omega\sinh\left(\frac{\alpha(1-z)}{2z_0\omega}\right)}{\alpha\cosh\left(\frac{\alpha}{2z_0\omega}\right) + \omega\sinh\left(\frac{\alpha}{2z_0\omega}\right)}\right), \tag{3.19a}$$

$$\alpha = \sqrt{4m^2\pi^2z_0^2+\omega^2}. \tag{3.19b}$$

The Fourier series solution for $\tilde{u}(x,z)$ is

$$\tilde{u}(x,z) = \sum_{m=1}^{\infty} a_m e^{\frac{z}{2z_0}} \left(\frac{\alpha \cosh\left(\frac{\alpha(1-z)}{2z_0\omega}\right) + \omega \sinh\left(\frac{\alpha(1-z)}{2z_0\omega}\right)}{\alpha \cosh\left(\frac{\alpha}{2z_0\omega}\right) + \omega \sinh\left(\frac{\alpha}{2z_0\omega}\right)} \right) \sin\left(\frac{m\pi x}{\omega}\right), \tag{3.20}$$

where a_m may be found by applying the boundary condition (3.9c); $a_m = 2/(m\pi)$. The solution to the original problem (3.7) is then given by

$$\begin{aligned} u(x,z) &= \frac{x}{\omega} + \frac{2e^{\frac{z}{2z_0}}}{\pi} \sum_{m=1}^{\infty} \frac{1}{m} \left(\frac{\alpha \cosh\left(\frac{\alpha(1-z)}{2z_0\omega}\right) + \omega \sinh\left(\frac{\alpha(1-z)}{2z_0\omega}\right)}{\alpha \cosh\left(\frac{\alpha}{2z_0\omega}\right) + \omega \sinh\left(\frac{\alpha}{2z_0\omega}\right)} \right) \sin\left(\frac{m\pi x}{\omega}\right), \\ \alpha &= \sqrt{4m^2\pi^2 z_0^2 + \omega^2}. \end{aligned} \tag{3.21}$$

Visualising these solutions (Figure 3.4) indicates a clear dependence of the shear zone half-width (δ_w) on the e-folding length scale (z_0). Velocity profiles at $z = 1$ are shown in Figure 3.5, with the black profile representing the constant viscosity solution. Taking the limit as $z_0 \to \infty$ in equation (3.21) recovers equation (3.5), as expected.

The strong dependence of the shear zone width on the vertical e-folding length scale prompted me to perform a simple scaling analysis to investigate their relationship. In the model, the shear-stress free bottom boundary condition is such that it forces all velocity contours to be perpendicular to the base of the layer, where the differential operators can be approximated under the assumption that u varies over the length scale d in the vertical direction. If I assume that z_0 is small then $\partial_{zz}u \ll \min\{\partial_{xx}u, (1/z_0)\partial_z u\}$, where ∂_i/∂_{ii} indicates a first/second order partial derivative operator in the ith direction. The $\partial_{zz}u$ term may then be neglected in equation (3.6), so that

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{1}{z_0}\frac{\partial u}{\partial z}. \tag{3.22}$$

Replacing the derivatives by order of magnitude estimates gives

$$\frac{u}{\delta_w^2} \propto \frac{1}{z_0}\frac{u}{1}, \tag{3.23}$$

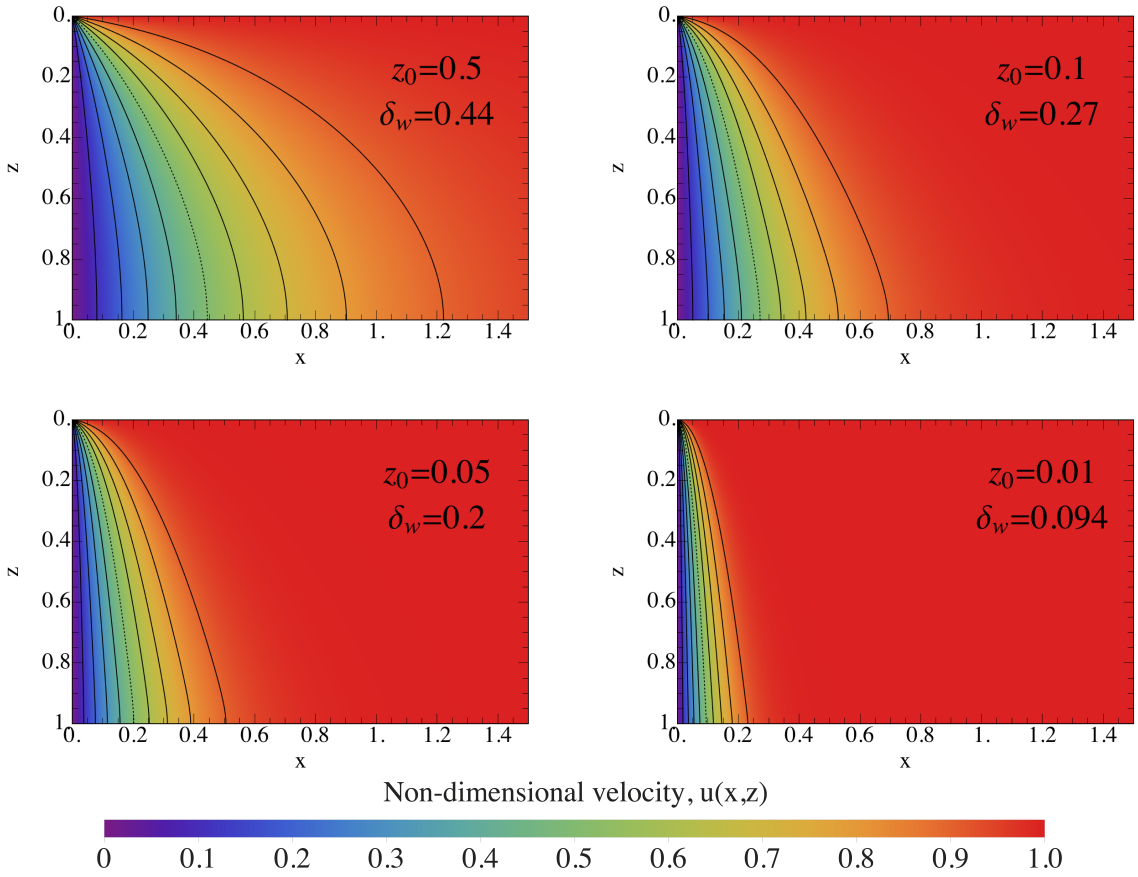


Figure 3.4: Exponentially depth dependent viscosity analytical solution, equation (3.21). Contours are plotted at 10% intervals of the driving velocity, with a dashed contour for 50%, the aspect ratio $\omega = 4$, and plots are labelled with z_0 and the shear zone half-widths, δ_w.

and the resulting scaling relation:

$$\delta_w \propto \sqrt{z_0}. \tag{3.24}$$

This scaling relation is plotted in Figure 3.6 versus the shear zone half-width obtained by numerically solving equation (3.21) for x when $z = 1$ and $u = 1/2$.

3.3.3 Arrhenius temperature dependence

The next increase in complexity is to incorporate an Arrhenius law, η_A from Table 3.1, which allows us to model the temperature dependence of viscosity. At this stage I am neglecting the feedback mechanism due to shear-stress heating; for a temperature

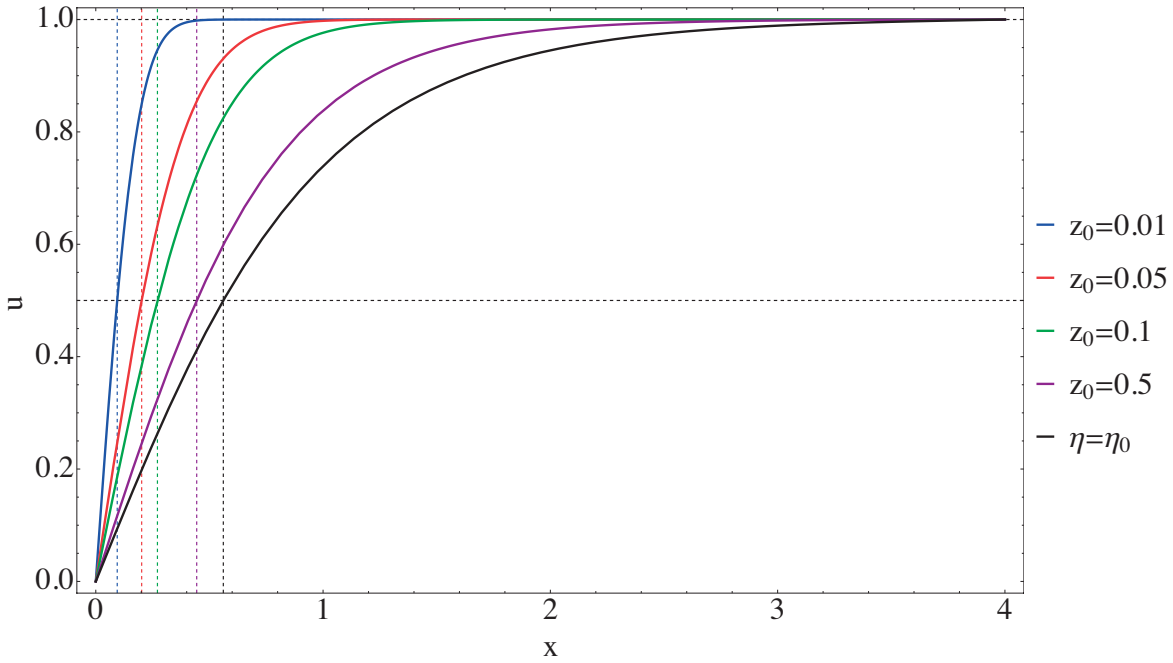


Figure 3.5: Velocity profiles at $z = 1$ for exponentially depth dependent viscosities, equation (3.21), with constant viscosity solution in black. Vertical dashed gridlines correspond to shear zone half-widths (δ_w) for each z_0.

dependent viscosity, $\eta(T)$, equation (3.3) becomes

$$\nabla^2 u = -\frac{\partial \ln \eta}{\partial T}\left(\frac{\partial T}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial T}{\partial z}\frac{\partial u}{\partial z}\right), \tag{3.25}$$

and substituting for $\eta = \eta_A$ gives,

$$\nabla^2 u = \left(\frac{Q}{RT^2} - \frac{1}{T}\right)\left(\frac{\partial T}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial T}{\partial z}\frac{\partial u}{\partial z}\right). \tag{3.26}$$

I assume a linear geotherm, $T(z) = T_s + \beta d(d_e + z)$, where d_e is the dimensionless lid thickness (t_e/d), T is in K, and equation (3.26) becomes

$$\nabla^2 u = \beta d\left(\frac{Q}{R(T_s + \beta d(d_e + z))^2} - \frac{1}{T_s + \beta d(d_e + z)}\right)\frac{\partial u}{\partial z}. \tag{3.27}$$

As the depth, z, appears in the denominator of the coefficient of the $\partial_z u$ term, equation (3.27) requires a numerical method of solution. The numerical codes to solve equation

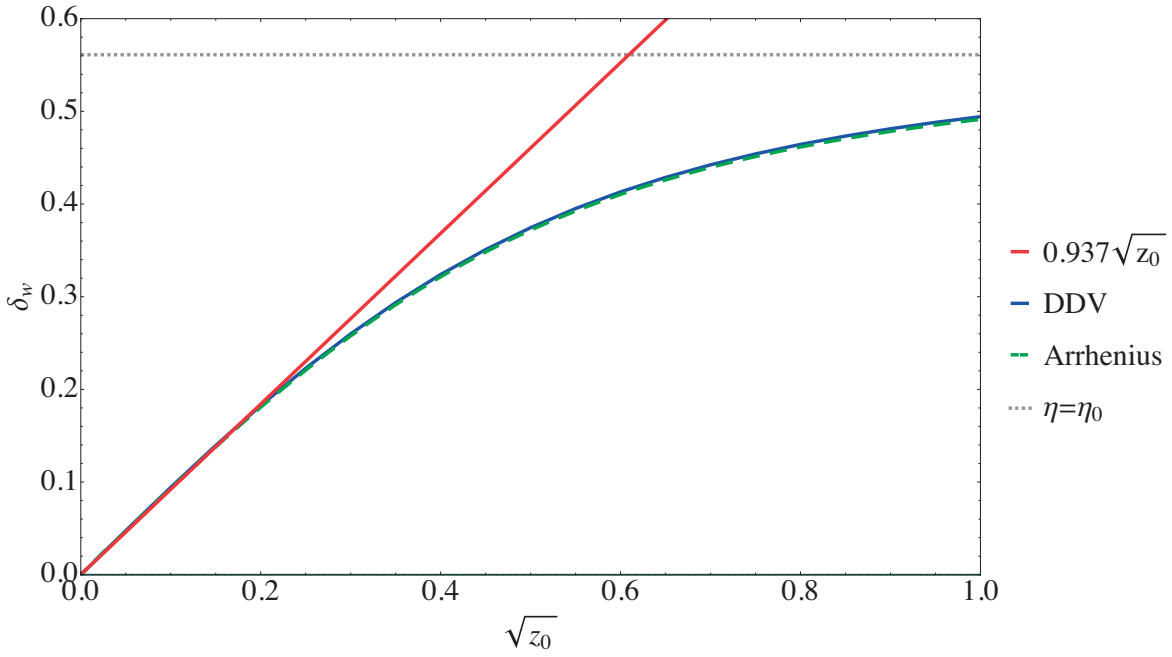


Figure 3.6: Shear zone half-width for depth dependent viscosity and Arrhenius law as a function of z_0 and effective z_0 (equation (3.28)) respectively. Scaling analysis of the DDV equation shows that for small z_0 the fit should be proportional to $\sqrt{z_0}$. The red line is $c\sqrt{z_0}$, where the coefficient $c = 0.937$ was found empirically using a least squares minimisation for $z_0 < 0.3$. The limiting case of a constant viscosity is included as the dashed grey line.

(3.27) utilise a successive over-relaxation, finite-difference method written in Mathematica 9.0 (*Wolfram Research* (2012)) and were benchmarked using the analytical solutions, equations (3.5) and (3.21). Rheological parameters were selected to cover a wide range of lithologies and continental geothermal gradients (*Hirth and Kohlstedt* (2003),*Hirth et al.* (2001)). Resolution tests for our numerical models were run with grid spacings from 1/10 to 1/200 and showed no more than 5% variation in δ_w, arising solely due to interpolation errors between grid points, where quadratic interpolation was used. Thus the numerical solution fields are essentially independent of the grid resolution.

Upon close examination of the numerical solutions it is apparent that the shear zone half-width (δ_w) varied in a similar fashion, for certain parameter choices, to that of an exponentially depth dependent viscosity. Motivated by that similarity, I attempted to identify an e-folding length scale equivalent to z_0. The zeroth-order Taylor approxima-

tion of equation (3.27) is equivalent to equation (3.6). The equation is Taylor expanded about $z = 1/2$ as this gives the closest approximation to the gradients of the viscosities for $1/2 < z < 1$ (Figure 3.9). This was tested by computing $\partial_z \ln \eta$ for the approximation and the Arrhenius law, and calculating the root mean square deviation (RMSD) as a function of expansion location ($0 \leq z \leq 1$), which showed a minima at $z \approx 0.5$. The effective length scale is then

$$z_0 \approx \left(\frac{Q\beta d}{R(T_s + \left(d_e + \frac{1}{2}\right)\beta d)^2} - \frac{\beta d}{T_s + \left(d_e + \frac{1}{2}\right)\beta d} \right)^{-1}. \tag{3.28}$$

If I use equation (3.28) to define an effective z_0 I find a velocity profile at $z = 1$ that is visually indistinguishable between the numerical solution and the analytical solution for the exponentially depth dependent viscosity, equation (3.21), with that value of z_0. Additionally, the shear zone half-width, δ_w, for the numerical solutions are found to be equivalent, to four significant figures, with that of the corresponding analytical solution. The velocity fields from these numerical solutions are included in the first column of Figure 3.14. A comparison of δ_w from the numerical Arrhenius solutions and the analytical DDV solution, equation (3.21), is illustrated in Figure 3.6, where equation (3.28) is used to define an effective z_0 for the Arrhenius model. The scaling of the shear zone half-width then follows as equation (3.24) with z_0 given by equation (3.28). This empirically confirms our choice of expansion location, as a plot of δ_w against effective z_0 only produces close agreement with the depth dependent viscosity results when the expansion is taken about $z = 1/2$.

This approximate solution accurately captures the behaviour of a shear zone governed by an Arrhenius law near $z = 1$; the absolute deviation of the velocity field is less than 6% for $z > 1/2$ (Figure 3.7). However, it fails to explain significant narrowing of the shear zone when $z < 1/2$. This may be quantified by looking at the absolute deviation of the velocity field, which shows less than 6% deviation for $z > 1/2$ (Figure 3.7). The narrower shear zone at shallow depths for an Arrhenius model compared to an exponentially depth dependent model (Figure 3.8) is due to the much larger vis-

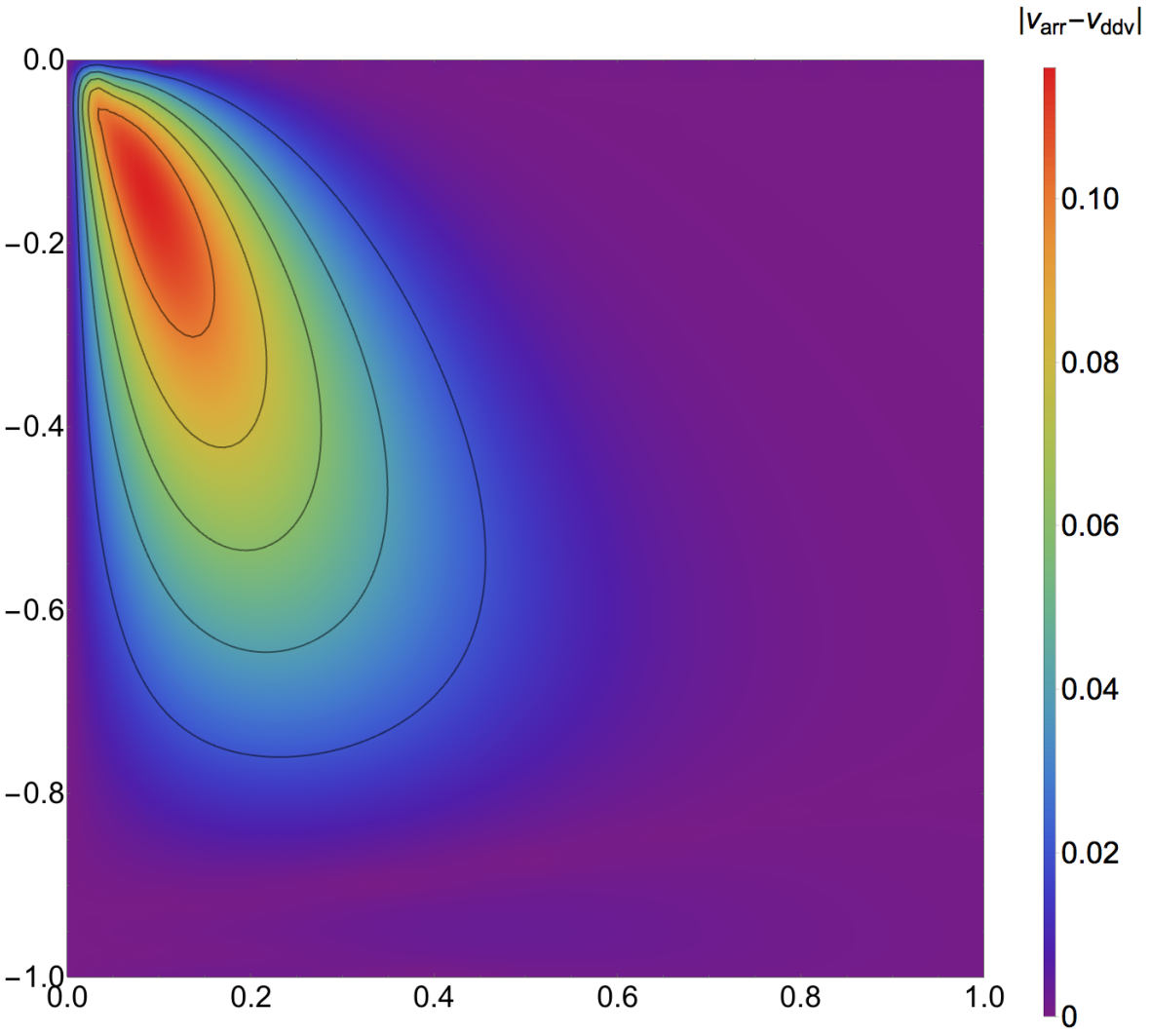


Figure 3.7: Absolute deviation of velocity field for the Arrhenius law with wet quartzite lithology and exponentially depth dependent viscosity with z_0 from equation (3.28). Contours are plotted at 2% intervals of the driving velocity, and aspect ratio of $\omega = 4$.

cosity gradients at shallow depth in an Arrhenius model (Figure 3.9). To explore the behaviour for $z < 1/2$, a first-order Taylor expansion of equation (3.27) about $z = 1/2$ is also solved, which quantitatively matches the numerical results (Figure 3.8).

Taking equation (3.27) and conducting the Taylor expansion of the $\partial_z u$ coefficient

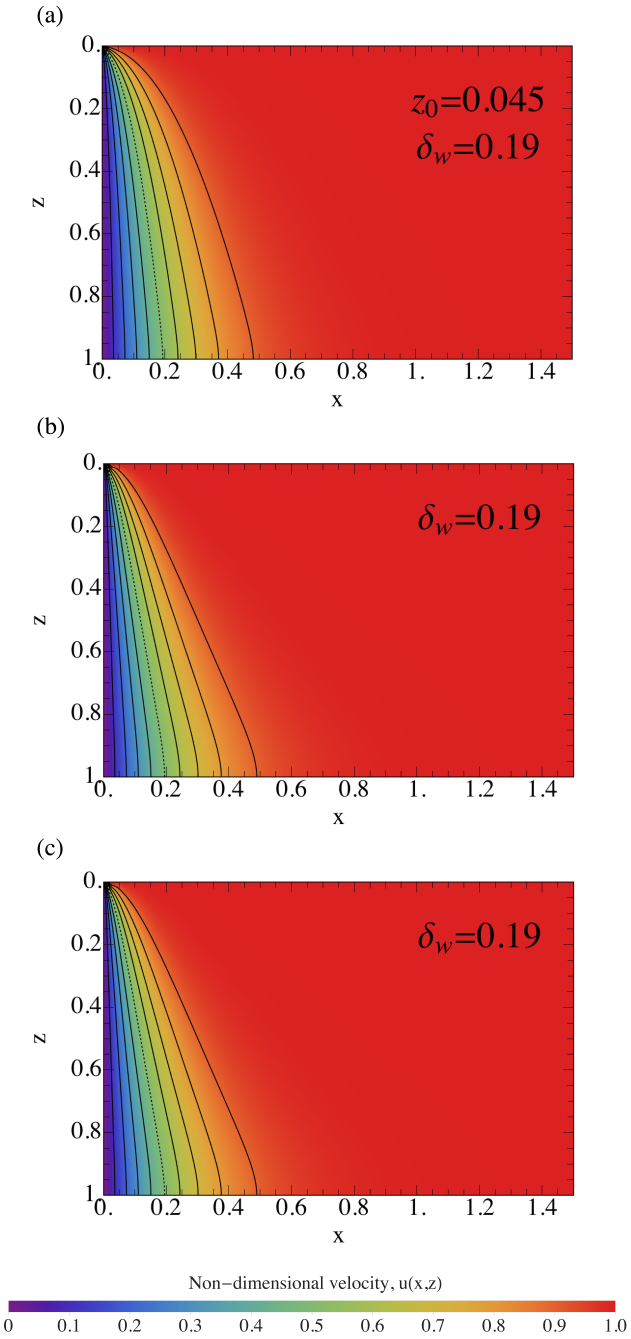


Figure 3.8: Comparison of velocity field for the analytical solution for exponentially depth dependent viscosity (0th order approximation to Arrhenius law) with effective $z_0 = 0.045$ (a), numerical solution (b), and first order approximate solution for Arrhenius law with wet quartzite lithology, effective $z_0 = 0.045$ (c). Contours are plotted at 10% intervals of the driving velocity, with a dashed contour for 50%, and an aspect ratio $\omega = 4$. Shear zone half-widths, δ_w, are included and z_0 where applicable.

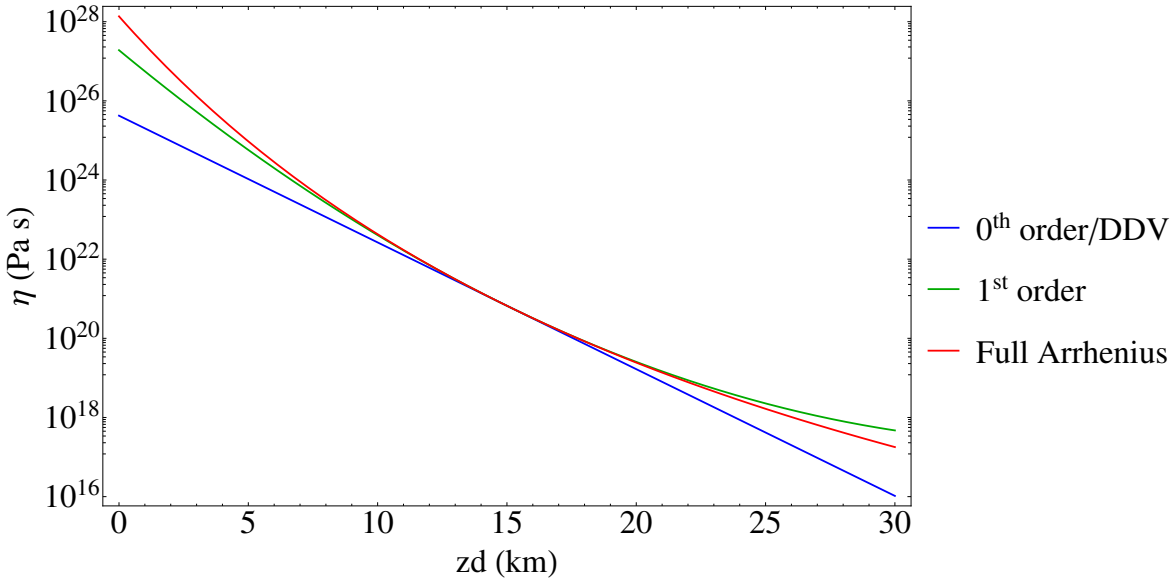


Figure 3.9: Log plot of viscosities for Arrhenius law, with zeroth (equivalent to DDV) and first order Taylor approximations. Parameters are for wet quartzite ($Q = 247$ kJ mol^{-1}, $B = 2.6 \times 10^{-4}$ K m kg^{-1}) with layer thickness $d = 30$km, elastic lid $t_e = 15$km, and effective $z_0 = 0.045$ from equation (3.28).

about $z = 1/2$, to a second approximation equation (3.27) becomes

$$\nabla^2 u = (a + bz)\partial_z u, \tag{3.29a}$$

$$a = -\frac{4\beta d(R[T_s + \beta d(d_e + 1)][2T_s + \beta d(2d_e + 1)] - Q[2T_s + \beta d(2d_e + 3)])}{R(2T_s + \beta d(2d_e + 1))^3}, \tag{3.29b}$$

$$b = \frac{4\beta d(R[T_s + \beta d(2d_e + 1)] - 4Q)}{R(2T_s + \beta d(2d_e + 1))^3}. \tag{3.29c}$$

This may be solved by the same method as in section 3.3.2, with the separable equations

$$X''(x) + \lambda_1 X(x) = 0, \tag{3.30a}$$

$$Z''(z) - (a + bz)Z'(z) - \lambda_1 Z(z) = 0, \tag{3.30b}$$

$$\lambda_1 = \frac{m^2\pi^2}{\omega^2}. \tag{3.30c}$$

These admit a Fourier series solution as before, though the Fourier coefficients are

correspondingly more elaborate,

$$u(x,z) = \frac{x}{\omega} + \frac{2}{\pi}\sum_{m=1}^{\infty}\frac{1}{m}F(a,b,\lambda,z)\sin\left(\frac{m\pi x}{\omega}\right), \tag{3.31a}$$

$$F(a,b,\lambda,z) = \frac{(a+b)H_\lambda\left(\frac{a+bz}{\sqrt{2b}}\right)\chi + \sqrt{2b}H_{\lambda-1}\left(\frac{a+b}{\sqrt{2b}}\right){}_1F_1\left(-\frac{\lambda}{2},\frac{1}{2},\frac{(a+bz)^2}{2b}\right)}{(a+b)H_\lambda\left(\frac{a}{\sqrt{2b}}\right)\chi + \sqrt{2b}H_{\lambda-1}\left(\frac{a+b}{\sqrt{2b}}\right){}_1F_1\left(-\frac{\lambda}{2},\frac{1}{2},\frac{a^2}{2b}\right)}, \tag{3.31b}$$

$$\chi = {}_1F_1\left(1-\frac{\lambda}{2},\frac{3}{2},\frac{(a+b)^2}{2b}\right), \tag{3.31c}$$

$$\lambda = -\frac{\lambda_1}{b}. \tag{3.31d}$$

There are two special functions in equation (3.31b): the confluent hypergeometric function of the first kind (${}_1F_1$) and the Hermite function (H_λ), which is an extension of the generating function for the Hermite polynomials for non-integer values. These are defined as

$$H_\lambda(x) = \frac{\Gamma(\lambda+1)}{2\pi i}\oint\frac{e^{2tz-t^2}}{t^{\lambda+1}}dt, \tag{3.32a}$$

$${}_1F_1(r,s,z) = \frac{\Gamma(s)}{\Gamma(s-r)\Gamma(r)}\int_0^1 e^{zt}t^{r-1}(1-t)^{s-r-1}dt, \tag{3.32b}$$

respectively, where $\Gamma(x)$ is the Gamma function.

3.4 Non-linear ductile shear zones

The shear zone model may be extended to examine the effect of a non-linear power law stress strain relation $\dot{\epsilon} \propto \tau^n$. If I define the second invariant of the strain rate tensor to be $J_2 = 2^{-\frac{1}{2}}\sqrt{(\partial_x u)^2 + (\partial_z u)^2}$, then non-linear equivalents to the rheologies of section 3 can be constructed with effective viscosities given by Table 3.1.

3.4.1 Uniform physical properties

Taking the effective viscosity, η_n from Table 3.1 and substituting it into equation (3.3) gives the governing equation for the non-linear rheology with uniform physical proper-

ties,

$$\nabla^2 u = \left(1 - \frac{1}{n}\right) \left(\partial_x [\ln (J_2)] \, \partial_x u + \partial_z [\ln (J_2)] \, \partial_z u\right). \tag{3.33}$$

This equation is solved numerically using a successive over-relaxation finite-difference method written in Mathematica 9.0 (*Wolfram Research* (2012)). The velocity fields are illustrated in Figure 3.10 with the shear zone half-width at the base of the layer (δ_w). Inspecting the results of the variation of δ_w with n on a log-log plot (Figure 3.11) leads us to infer a fit of the form $\delta_w \propto 1/\sqrt{n}$.

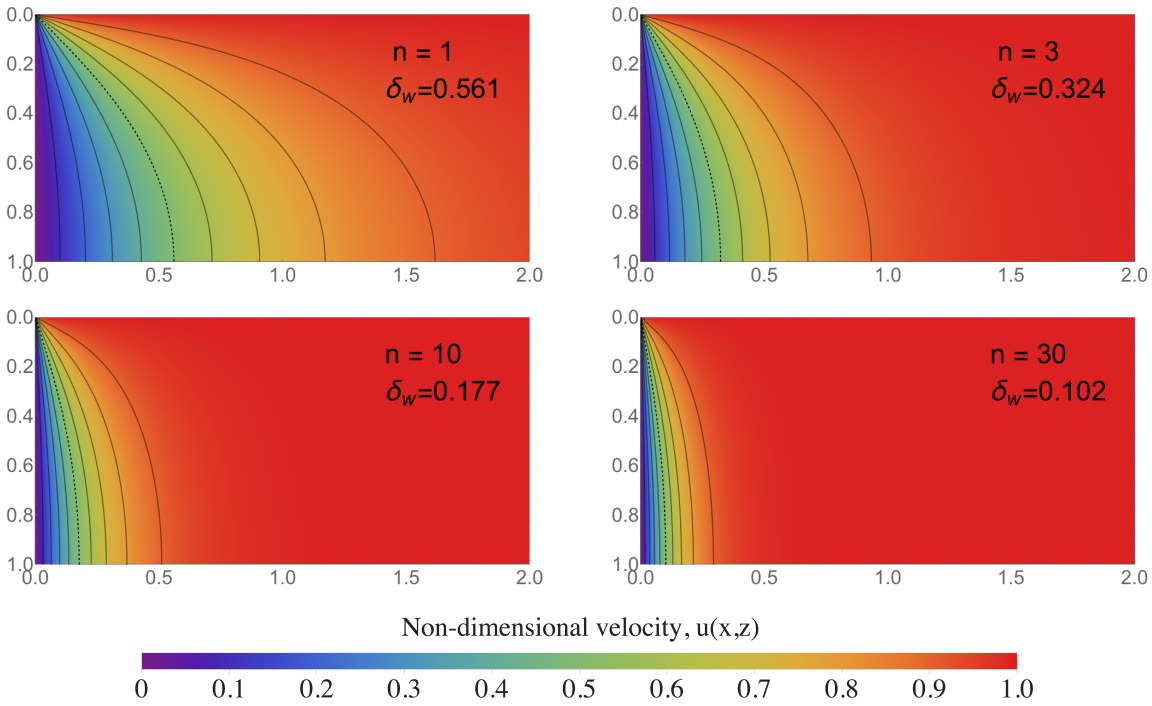


Figure 3.10: Uniform properties non-linear velocity field, equation (3.39). Contours are plotted at 10% intervals of the driving velocity, with a dashed contour for 50%, the aspect ratio $\omega = 4$ and the shear zone half-widths, δ_w, for each n are included.

Using the numerical solutions as a guide to the analysis of equation (3.33), it is apparent that for the majority of the shear zone the horizontal gradients in velocity are significantly greater than the vertical. If I assume that this is generally true then I may solve an approximation of equation (3.33). Starting with the equation for a

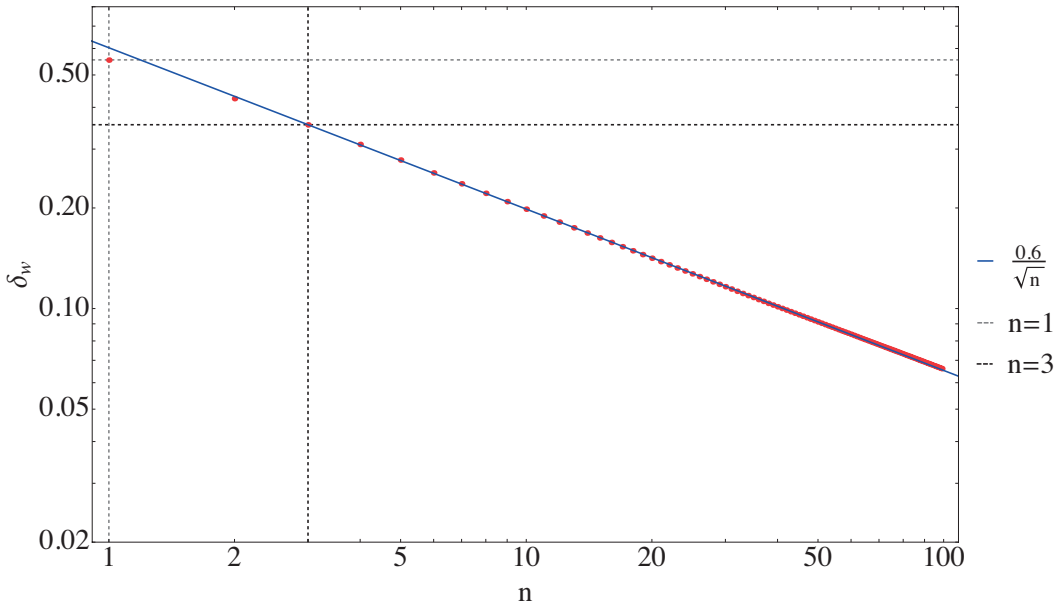


Figure 3.11: Log-log plot of shear zone half-width against n for power-law stress-strain relation with uniform properties. The empirical fit is $0.6/\sqrt{n}$.

non-linear ductile shear zone with constant viscosity

$$\nabla^2 u = \left(1 - \frac{1}{n}\right)\left(\partial_x \ln\left(J_2\right)\partial_x u + \partial_z \ln\left(J_2\right)\partial_z u\right) \tag{3.34}$$

we may substitute for the second invariant of the strain-rate tensor,

$$J_2 = 2^{-\frac{1}{2}}\sqrt{(\partial_x u)^2 + (\partial_z u)^2}, \tag{3.35}$$

to obtain an expression purely in $u(x, z)$ and its derivatives

$$\begin{aligned}
\ln\left(J_2\right) &= \frac{1}{2}\ln((\partial_x u)^2 + (\partial_z u)^2) - \frac{1}{2}\ln(2), \\
\partial_x\left(\frac{1}{2}\ln\left(J_2\right)\right) &= \frac{\partial_{xx} u \partial_x u + \partial_{xz} u \partial_z u}{(\partial_x u)^2 + (\partial_z u)^2}, \\
\partial_z\left(\frac{1}{2}\ln\left(J_2\right)\right) &= \frac{\partial_{xz} u \partial_x u + \partial_{zz} u \partial_z u}{(\partial_x u)^2 + (\partial_z u)^2}, \\
\nabla^2 u &= \left(1 - \frac{1}{n}\right)\frac{\partial_{xx} u (\partial_x u)^2 + \partial_{zz} u (\partial_z u)^2 + 2\partial_{xz} u \partial_x u \partial_z u}{(\partial_x u)^2 + (\partial_z u)^2}.
\end{aligned} \tag{3.36}$$

With the definition of $\nabla^2 = \partial_{xx} + \partial_{zz}$ this may be rearranged to give

$$\frac{1}{n}(\partial_{xx}u(\partial_x u)^2 + \partial_{zz}u(\partial_z u)^2) + \partial_{xx}u(\partial_z u)^2 + \partial_{zz}u(\partial_x u)^2 = \left(\frac{2}{n} - 2\right)\partial_{xz}u\partial_x u\partial_z u. \quad (3.37)$$

Near the base of the shear zone, the behaviour of equation (3.37) is dominated by the contribution from the horizontal gradients, $\partial_x u$, as the vertical gradients, $\partial_z u$, approach zero at $z = 1$ (equation 3.9d). This is particularly clear in the numerical solutions, where the horizontal gradients are several orders of magnitude higher than that vertical gradients. Thus assuming $(\partial_x u)^2 \gg \max\{(\partial_z u)^2, \partial_x u\partial_z u\}$, I conclude from the coefficient of the $(\partial_x u)^2$ terms that

$$\partial_{xx}u + n\partial_{zz}u = 0. \quad (3.38)$$

Subject to the boundary conditions (3.4d), this admits a Fourier series solution

$$u(x,z) = \frac{x}{\omega} + \frac{2}{\pi}\sum_{m=1}^{\infty}\frac{1}{m}\frac{\cosh\left(\frac{m\pi(1-z)}{\omega\sqrt{n}}\right)}{\cosh\left(\frac{m\pi}{\omega\sqrt{n}}\right)}\sin\left(\frac{m\pi x}{\omega}\right). \quad (3.39)$$

Equation (3.39) produces a velocity field that is in agreement within the accuracy of the numerical experiments (3sf) on all grid points for $1 \leq n \leq 100$, where the Fourier series is truncated at an upper Nyquist frequency appropriate for the grid spacing. It is apparent from equation (3.39) that the length scale in the Fourier coefficient is modified by a factor of $1/\sqrt{n}$, matching that found empirically from the numerical solutions. It also exactly reduces to the analytical solution for the linear case when $n = 1$. The other limit of interest is that of purely plastic behaviour, or $n \to \infty$, where it can be shown that equation (3.39) reduces to

$$\lim_{n\to\infty}[u(x,z)] = \begin{cases} 0, & x = 0, \\ 1, & x \neq 0. \end{cases} \quad (3.40)$$

Physically, as we approach plastic behaviour, all motion is localised onto a plane on

the down-dip extension of the fault, with $\delta_w = 0$. Numerical solutions also show shear collapsing onto a plane in the limiting case of $n \to \infty$ for equation (3.33). A scaling analysis of the approximate governing equation (3.38) produces a relation

$$\delta_w \propto \frac{1}{\sqrt{n}}, \tag{3.41}$$

which is commensurate with both the numerical results (Figure 3.11) and the analytical limit, equation (3.40).

The result of equation (3.41) is similar to that found in *England et al.* (1985), who were investigating the equation for creeping flow in a thin viscous sheet with a non-Newtonian viscosity with imposed velocity boundary conditions. Their main assumption was that the velocity components normal to the imposed velocity boundary condition could be neglected. Though they were modelling deformation in plan view, and their starting equations were somewhat different, their equation could also be approximated as a quasi-Laplace equation modified by n, resulting in deformation decaying away from the driving boundary on a length scale dependent on $1/\sqrt{n}$.

3.4.2 Exponentially depth dependent viscosity

The modifications required to include depth dependence in the problem are relatively straightforward. Substituting the viscosity structure η_{Dn} from Table 3.1 into equation (3.3) produces

$$\nabla^2 u = \frac{1}{z_0}\partial_z u + \left(1 - \frac{1}{n}\right)\left(\partial_x[\ln(J_2)]\,\partial_x u + \partial_z[\ln(J_2)]\,\partial_z u\right), \tag{3.42}$$

which is solved numerically. The half-width of the shear zone at the base of the layer is then plotted in Figure 3.12 as a function of n and z_0. Using the same approximations as for the non-linear uniform properties case the governing equation simplifies to

$$\partial_{xx} u + n\partial_{zz} u = \frac{1}{z_0}\partial_z u, \tag{3.43}$$

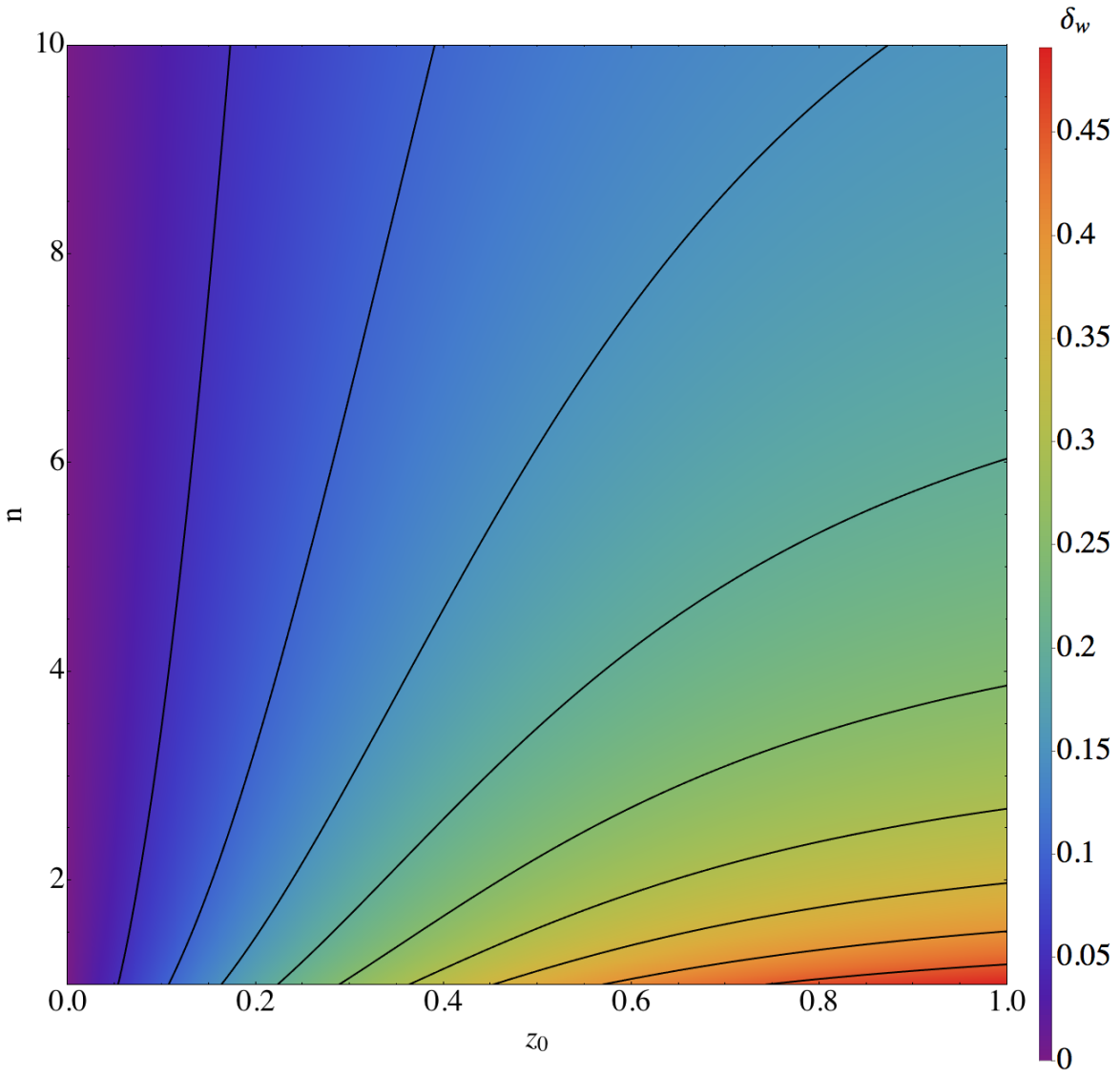


Figure 3.12: Non-linear depth dependent shear zone half-width as a function of n and z_0, from numerical solutions. For lines of constant n, profiles appear as in Figure 3.6, scaled by $1/\sqrt{n}$. Contours are plotted at intervals of 0.05 in δ_w. For small z_0 these are lines of constant $\sqrt{z_0/n}$.

with solution

$$u(x,z) = \frac{x}{\omega} + \frac{2e^{\frac{z}{2z_0}}}{\pi} \sum_{m=1}^{\infty} \frac{1}{m} \left(\frac{\alpha \cosh\left(\frac{\alpha(1-z)}{2z_0\omega\sqrt{n}}\right) + \omega\sqrt{n}\sinh\left(\frac{\alpha(1-z)}{2z_0\omega\sqrt{n}}\right)}{\alpha \cosh\left(\frac{\alpha}{2z_0\omega\sqrt{n}}\right) + \omega\sqrt{n}\sinh\left(\frac{\alpha}{2z_0\omega\sqrt{n}}\right)} \right) \sin\left(\frac{m\pi x}{\omega}\right),$$

$$\alpha = \sqrt{4m^2\pi^2 z_0^2 + \omega^2 n}. \tag{3.44}$$

As expected, this simplifies to give the solution for the depth dependent viscosity in the limit of $n = 1$, equation (3.21), and the approximate solution for a non-linear ductile shear zone with uniform properties in the limit of $z_0 \to \infty$, equation (3.39). Equation (3.44) also produces a velocity field that is in agreement within the resolution of the numerical experiments (3sf) on all grid points for $1 \leq n \leq 100$. The scaling of the shear zone half-width with both exponentially depth-dependent viscosity and non-linear rheology follows as:

$$\delta_w \propto \sqrt{\frac{z_0}{n}}. \tag{3.45}$$

3.4.3 Arrhenius temperature dependence

The governing equation for this rheology is obtained by substituting η_{An} from Table 3.1 into equation (3.3),

$$\nabla^2 u = \left(\frac{Q}{RT^2} - \frac{1}{T}\right)(\partial_x T \partial_x u + \partial_z T \partial_z u) + \left(1 - \frac{1}{n}\right)(\partial_x \ln|J_2| \partial_x u + \partial_z \ln|J_2| \partial_z u), \tag{3.46}$$

and for the temperature structure $T(z) = T_s + \beta d(d_e + z)$ this becomes

$$\begin{aligned} \nabla^2 u = &\left(\frac{Q\beta d}{R(T_s + \beta d(d_e + z))^2} - \frac{\beta d}{T_s + \beta d(d_e + z)}\right)\partial_z u \\ &+ \left(1 - \frac{1}{n}\right)(\partial_x \ln|J_2| \partial_x u + \partial_z \ln|J_2| \partial_z u), \end{aligned} \tag{3.47}$$

which is also solved numerically. Velocity fields for a non-linear rheology with Arrhenius temperature dependence are plotted in Figure 3.14, with dimensional shear zone half-widths, δ_w, for $d = 30$km ranging from 2.34 - 3.52km. From examining the numerical solutions is it apparent that the scaling relations already presented, equations (3.28) and (3.45), predict δ_w with an accuracy greater than 98%. The approximate analytical solution for an exponentially depth dependent viscosity, equation (3.44), combined with the effective e-folding length scale provided by equation (3.28) also provides a qualitatively similar velocity field to the numerical solutions. However, the same caveat as illustrated in Figure 3.8 applies in that the approximate analytical solution produces

a wider shear zone than the numerical simulation for $z < 1/2$. Again a first-order Taylor approximation improves this considerably, at the expense of significantly increased complexity. If I start with equation (3.47), utilise the same approximations as for the non-linear uniform properties case, and conduct the Taylor expansion of the $\partial_z u$ coefficient about $z = 1/2$, to a second approximation equation (3.47)becomes

$$\partial_{xx}u + n\partial_{zz}u = (a + bz)\partial_z u, \tag{3.48a}$$

$$a = -\frac{4\beta d(R[T_s + \beta d(d_e + 1)][2T_s + \beta d(2d_e + 1)] - Q[2T_s + \beta d(2d_e + 3)])}{R(2T_s + \beta d(2d_e + 1))^3}, \tag{3.48b}$$

$$b = \frac{4\beta d(R[T_s + \beta d(2d_e + 1)] - 4Q)}{R(2T_s + \beta d(2d_e + 1))^3}. \tag{3.48c}$$

Again the solution is found using the method of separation of variables, with

$$u(x, z) = \frac{x}{\omega} + \frac{2}{\pi}\sum_{m=1}^{\infty}\frac{1}{m}F(a, b, \lambda, z)\sin\left(\frac{m\pi x}{\omega}\right), \tag{3.49a}$$

$$F(a, b, \lambda, z) = \frac{(a + b)H_\lambda\left(\frac{a+bz}{\sqrt{2bn}}\right)\chi + \sqrt{2bn}H_{\lambda-1}\left(\frac{a+b}{\sqrt{2bn}}\right){}_1F_1\left(-\frac{\lambda}{2}, \frac{1}{2}, \frac{(a+bz)^2}{2bn}\right)}{(a + b)H_\lambda\left(\frac{a}{\sqrt{2bn}}\right)\chi + \sqrt{2bn}H_{\lambda-1}\left(\frac{a+b}{\sqrt{2bn}}\right){}_1F_1\left(-\frac{\lambda}{2}, \frac{1}{2}, \frac{a^2}{2bn}\right)}, \tag{3.49b}$$

where the special functions H_λ and ${}_1F_1$ defined in equations (3.32a) and (3.32b).

3.5 Shear-stress heating and thermomechanical coupling

The last physical mechanism for shear zone localisation I investigate is thermomechnical coupling. The equations to be solved are based upon the linear and non-linear equations (3.25) and (3.46) where the temperature field is now allowed to vary laterally. The solution for the temperature field arises from shear-stress heat production and heat diffusion governed by the heat equation.

$$\rho c\frac{\partial T}{\partial t} = K\nabla^2 T + \tau_{ij}\dot{\epsilon}_{ij}, \tag{3.50}$$

using the rate of shear heating at each time step,

$$\tau_{ij}\dot{\epsilon}_{ij} = \begin{cases} 0, & -d_e < z < 0, \\ \eta(J_2)^2 = \frac{1}{2}\eta\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial z}\right)^2\right], & 0 < z < 1, \end{cases} \tag{3.51}$$

to modify the temperature field and hence the viscosities. The dissipative heating term, equation (3.51), is dependent on both the velocity field, $u(x, z)$, and the temperature field through the definition of viscosity η. I must also include the rigid lid (Figure 3.1) as part of the domain to be solved, because heat will be lost via conduction through this layer, and we neglect the contribution of shear heating in the brittle fault zone. Frictional heating on the fault plane ($z < 0$) does not significantly affect the temperature field in the viscous layer as it is dominated by the heat generated by the stress singularity at the base of the fault.

For the solution of equation (3.50) I choose $z = 0$ to lie on the interface between the rigid lid and the viscous layer, so the domain of T is ($0 \leq x \leq \omega$, $-d_e \leq z \leq 1$), where d_e is the non-dimensionalised elastic layer thickness, t_e/d. A linear geothermal gradient, $T(t = 0) = T_s + \beta d(d_e + z)$, is used as the initial condition on temperature for the heat equation, and dissipative heating, equation (3.51), allowed in the region $0 \leq z \leq 1$ only. The boundary conditions used for the heat equation (3.50) are

$$\left.\frac{\partial T}{\partial x}\right|_{x=0} = 0, \tag{3.52a}$$

$$T(\omega, z) = T_s + \beta d(d_e + z), \tag{3.52b}$$

$$T(x, -d_e) = T_s, \tag{3.52c}$$

$$\left.\frac{\partial T}{\partial z}\right|_{z=d} = \beta. \tag{3.52d}$$

3.5.1 Arrhenius temperature dependence

For the Arrhenius temperature dependence with a linear stress-strain rate relationship I replace η in equation (3.51) with η_A from Table 3.1. The initial velocity condition

for the shear heating term, $u(t = 0)$, is then taken to be that of the Arrhenius solution with a linear geothermal gradient. Equation (3.27) is solved first, followed by the heat equation (3.50). The steady state solution is obtained by iteration between solving equations (3.26) and (3.50).

The numerical codes for solving the heat equation utilise the method of lines with a pseudospectral Chebyshev grid for the spatial discretisation, as implemented in the NDSolve routine in Mathematica 9.0 (*Wolfram Research* (2012)). A Chebyshev grid is used to ensure small thermal transients near the axis of heating are accurately captured. The model is run until it reaches an approximately steady state for both temperature and velocity fields. I define steady state as less than a 0.1% change in the thermal and velocity fields over 1Ma.

To derive an approximation for the production of shear heating in our model I start with the expression for an infinite line source of strength q (*Carslaw and Jaeger* (1959), p.262 equation (7)) buried in a half-space at $x = 0, z = z'$ with the boundary condition $T(x, 0) = 0$,

$$T(x,z) = -\frac{q}{4\pi\kappa}\left(Ei\left[-\frac{x^2+(z-z')^2}{4\kappa t}\right] - Ei\left[-\frac{x^2+(z+z')^2}{4\kappa t}\right]\right), \tag{3.53}$$

where Ei is the exponential integral,

$$-Ei(-x) = \int_x^\infty \frac{e^{-u}}{u}\,du. \tag{3.54}$$

We then substitute $q = q_0 \exp\left(-(z' - t_e)/z_1\right)$ to give a source strength that decays with depth (as shear heating is proportional to viscosity, which decays exponentially with depth),

$$T_{line}(x,z) = -\frac{q_0 e^{\frac{t_e - z'}{z_1}}}{4\pi\kappa}\left(Ei\left[-\frac{x^2+(z-z')^2}{4\kappa t}\right] - Ei\left[-\frac{x^2+(z+z')^2}{4\kappa t}\right]\right). \tag{3.55}$$

To obtain the solution for an exponentially depth dependent infinite half plane heat source, buried at a depth t_e, I need to integrate equation (3.55) with respect to the

source location z' between $z' = t_e$ and $z' = d$. Equation (3.55) cannot be integrated analytically; however, if I assume that $(x^2 + (z \pm z')^2))/(4\kappa t)$ is small, then I can use the Taylor expansion of the exponential integral function, $Ei(-x)$, about $x = 0$,

$$Ei(-x) = \gamma + \ln(x) - x + \frac{1}{4}x^2 + O(x^3), \tag{3.56}$$

where γ is Euler's constant. If we assume the dimensions are no bigger than 100km in x or z then this expansion is valid for $t \gg 50$ years. Also, as the shear heating at $z' = d$ is much less than at $z' = t_e$ I can let $d \to \infty$ for the purposes of simplifying the result, within error of the Taylor expansion. Thus to first order equation (3.55) becomes

$$T_{line}(x,z) \approx \frac{q_0 e^{\frac{t_e - z'}{z_1}}}{4\pi\kappa}\left(2\,\mathrm{arctanh}\left[\frac{2zz'}{x^2+z^2+z'^2}\right] - \frac{zz'}{\kappa t} + O\left(\frac{1}{t^2}\right)\right), \tag{3.57}$$

with this approximation the only time dependence is $zz'/\kappa t$. The solution for T_{plane} is then obtained by integration,

$$T_{plane} = \int_{t_e}^{\infty} T_{line}\, dz', \tag{3.58}$$

which gives

$$\begin{aligned} T_{plane} =& \frac{q_0 z_1}{\kappa}\left(e^{\frac{t_e}{z_1}}\sin\left(\frac{|x|}{z_1}\right)\sinh\left(\frac{y}{z_1}\right) - \frac{z(t_e+z_1)}{4\pi\kappa t}\right. \\ &\left. + \frac{1}{2\pi}\,\mathrm{arctanh}\left(\frac{2t_e z}{t_e^2+x^2+z^2}\right) + \frac{A}{4\pi}\right), \end{aligned} \tag{3.59a}$$

$$\begin{aligned} A =& e^{\frac{t_e+ix-z}{z_1}}Ei\left[\frac{-t_e-ix+z}{z_1}\right] + e^{\frac{t_e-ix-z}{z_1}}Ei\left[\frac{-t_e+ix+z}{z_1}\right] \\ &- e^{\frac{t_e+ix+z}{z_1}}Ei\left[\frac{-t_e-ix-z}{z_1}\right] - e^{\frac{t_e-ix+z}{z_1}}Ei\left[\frac{-t_e+ix-z}{z_1}\right]. \end{aligned} \tag{3.59b}$$

There is only one term depending on t, so the effective thermal diffusion timescale is found by substituting $z = d$,

$$\tau = \frac{dt_e}{4\pi\kappa}, \tag{3.59c}$$

which for our model ranges from 0.5-1Ma.

A comparison of the velocity fields with and without shear-stress heating are given in Figure 3.14, while Figure 3.13 illustrates the temperature increase due to shear-stress heating at steady state ($t \sim 5\tau$). The variation in shear heating is strongly dependent on the initial width of the shear zone, which is itself controlled by the rheology. If the strain-rates are not large enough, or equivalently the effective z_0 not small enough ($z_0 < 0.2$), then no significant shear heating will be generated on the thermal diffusion timescale and the temperature field remains relatively unperturbed. Thus for thermomechanical coupling to be a major contributing factor in shear zone localisation, it requires rheological parameters that produce a narrow shear zone initially, with effective $z_0 < 0.2$. It should also be noted that even when $z_0 = 0.1$ and significant shear heating is generated, such as for dry olivine, the change in the shear zone width is only $\sim 20\%$.

3.5.2 Non-linear rheology with Arrhenius temperature dependence

Thermomechanical coupling for a non-linear Arrhenius law is modelled in the same way as described in section 3.5.1, except that the non-linear effective viscosity η_{An} is used in place of the linear η_A (Table 3.1). If I look at the shear heating term, equation 3.51, the dependence on the gradients in velocity is actually weaker for a non-linear rheology than a linear one. Substituting η_{An} into equation 3.51 gives, for $0 < z < 1$,

$$\tau_{ij}\dot{\epsilon}_{ij} = \frac{T}{B}\exp\left(\frac{Q}{RT}\right)2^{-\frac{(1+3n)}{2n}}\left[(\partial_x u)^2 + (\partial_z u)^2\right]^{\frac{n+1}{2n}}. \tag{3.60}$$

The exponent for the last term in equation (3.60), $(n+1)/2n$, is maximal for $n = 1$ and decreases monotonically, approximating $1/2$ in the limit $n \to \infty$; the strongest dependence of the shear heating production on the velocity gradients is therefore for a linear rheology, though the initial width of the shear zone decreases with increasing n. The numerical solvers iterate between equation (3.46) and the heat equation (3.50)

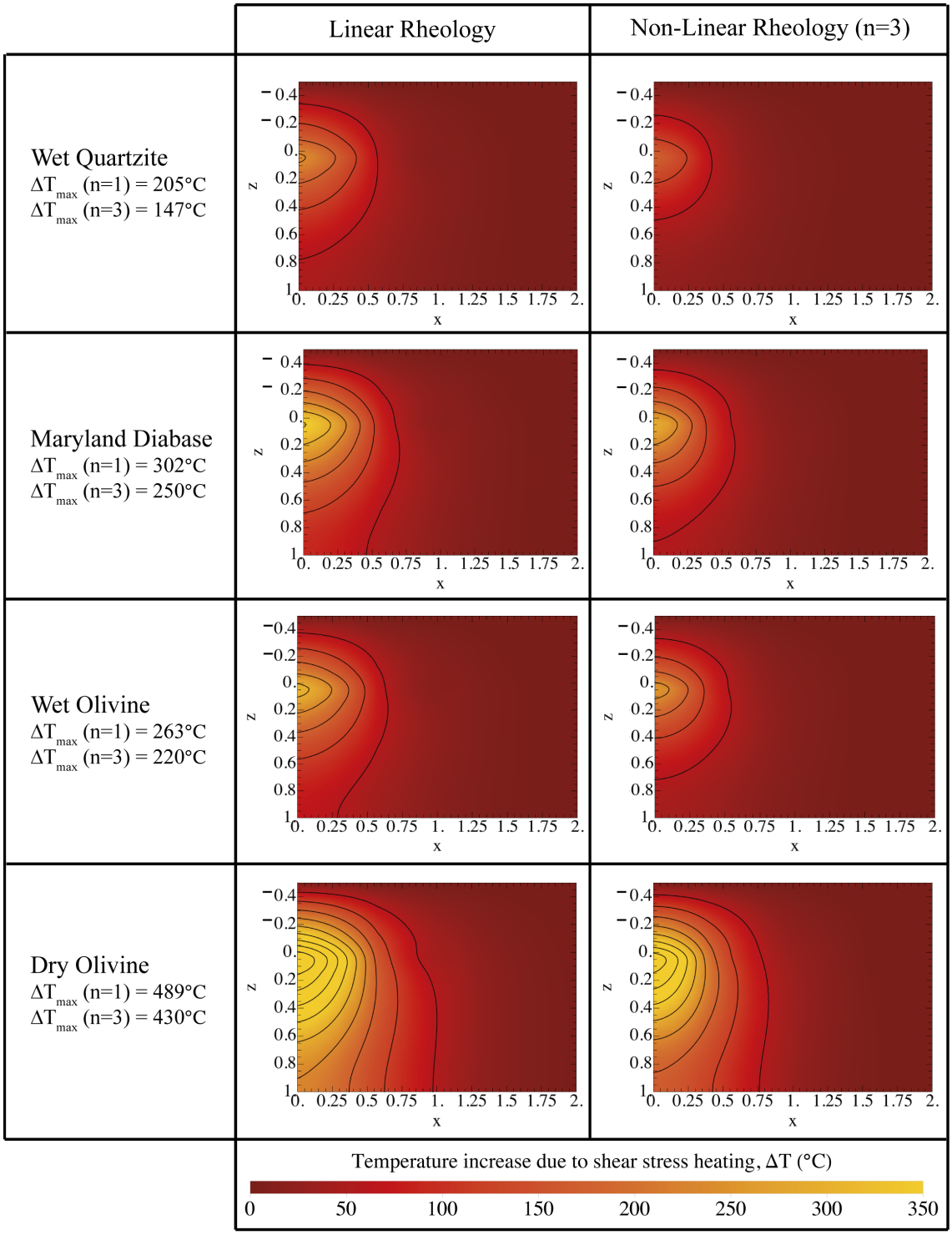


Figure 3.13: Temperature fields relative to the steady state geothermal gradient, driving velocity of 35mm/yr, geothermal gradient 20°C/km, elastic lid thickness d_e = 15km, and viscous layer thickness d = 30km. Contours at 50°C intervals.

with the shear heating term given by equation (3.60). Velocity profiles for a non-linear Arrhenius rheology with shear heating are plotted on Figure 3.14, next to the equivalent model without shear heating. Figure 3.13 illustrates the temperature increase due to shear-stress heating, in comparison to a linear rheology. One consequence of our model is that despite starting the iteration for the thermomechanically coupled model with a narrower shear zone due to the non-linear rheology, which in a linear rheology would result in greater shear heating, less shear heating is generated overall because of strain rate softening. However, because of the non-linearity of equation (3.46) significant localisation of shear can still occur, in some cases more than for a linear rheology despite having a lower maximum temperature increase, ΔT_{max}.

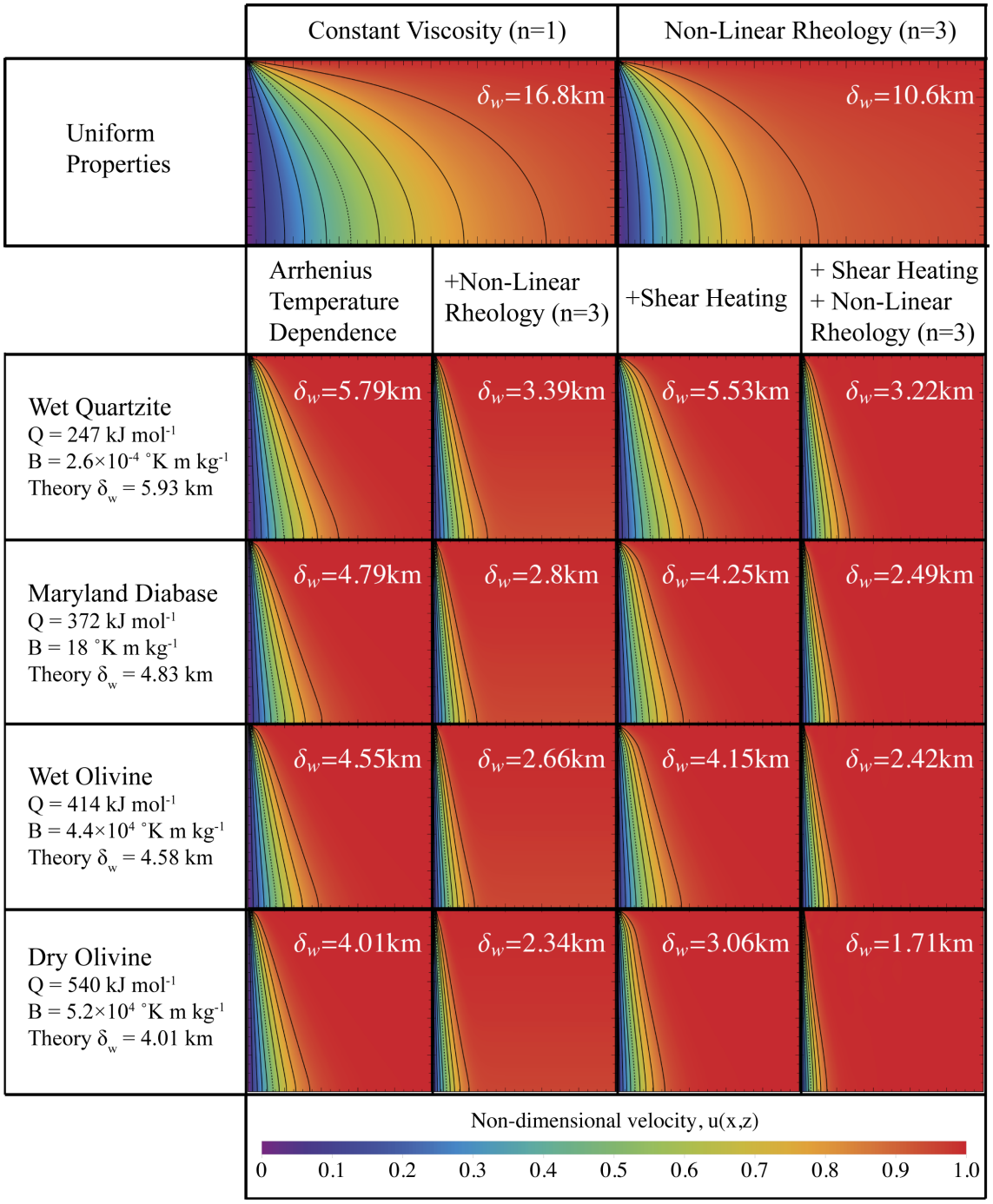


Figure 3.14: Comparison of velocity fields for given material parameters (*Hirth and Kohlstedt* (2003),*Hirth et al.* (2001)). Each square plot has height and width d, whilst the rectangular plots have width $2d$. Each plot is labelled with a dimensional shear zone half-width, δ_w, for $d = 30$km and the left hand column has a theoretical δ_w for the Arrhenius temperature dependence calculated using equation (3.28). Contours are plotted at 10% intervals of the driving velocity, with a dashed contour for 50%, the aspect ratio $\omega = 4$, and plots are labelled with z_0 and the shear zone half-widths, δ_w.

3.6 Discussion

I have investigated the interplay of three physical mechanisms responsible for the localisation of shear at depth beneath a strike-slip fault: depth dependent viscosity, a non-linear stress-strain relation and thermomechanical coupling driven by shear-stress heating. Velocity fields illustrating the effect of each physical mechanism are summarised in Figure 3.14.

The strongest control on the shear zone width is the effective length scale of vertical variation of viscosity, as a result of an Arrhenius law with a linear geothermal gradient. Simplifying the force balance equation (3.1) under the assumption of anti-plane conditions,

$$\frac{\partial \tau_{xy}}{\partial x} = -\frac{\partial \tau_{zy}}{\partial z}, \tag{3.61}$$

and substituting in equation (3.2) produces

$$\frac{\partial}{\partial x}\left(\eta \frac{\partial u}{\partial x}\right) = -\frac{\partial}{\partial z}\left(\eta \frac{\partial u}{\partial z}\right) \tag{3.62}$$

From equation (3.62) it can be seen that when the right hand side is large due to rapid variations in viscosity, to balance it on the left hand side requires a reduction in the horizontal length scale. This leads to the scaling relation where the horizontal length scale is inversely proportional to the square root of the vertical gradients in viscosity.

Relative to the constant viscosity solution, the depth dependence of viscosity alone can produce a reduction in width of $60-75\%$, whilst a non-linear rheology with $n=3$ produces a further reduction in width of $(1-1/\sqrt{3}) \approx 42\%$. Shear-stress heating can add an additional $20-25\%$ to these effects. For a range of rheological properties and geothermal gradients, shear zones with a half width of 1.7 - 6km can develop in continental crust 30 - 40km thick, compared to 17km for a constant viscosity layer or 10km for a layer with uniform non-linear properties and $n=3$. These widths scale linearly with the thickness of the crust, so thinner crust would result in correspondingly narrower shear zones.

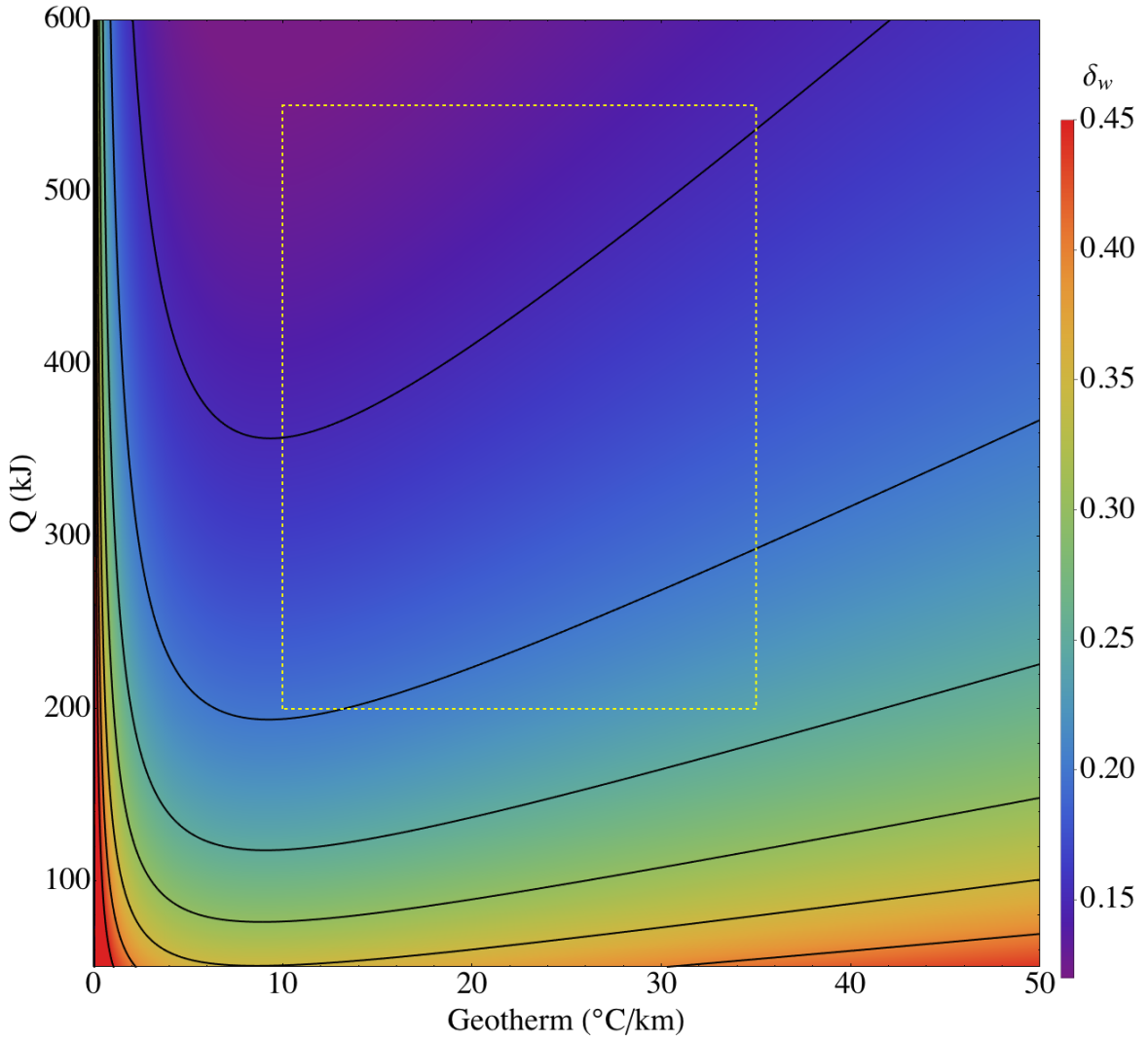


Figure 3.15: Shear zone half-width for $n = 1$ as a function of Arrhenius input parameters, activation energy (Q) and geotherm. Reasonable bounds for continental crust are outlined in yellow dashed box. Contours are plotted at intervals of 0.05 of δ_w.

The half width of the shear zone with an Arrhenius rheology is illustrated in Figure 3.15. To analyse the scaling of the shear zone width for a non-linear Arrhenius rheology, I can combine equations (3.28) and (3.45) using the empirical scaling coefficient 0.937 from Figure 3.6. This is further simplified by noting $Q \gg RT_{\frac{1}{2}}$, where $T_{\frac{1}{2}}$ is the temperature in the middle of the layer, $T_s + (d_e + 1/2)\beta d$, producing a dimensional shear zone half-width

$$\delta_w \approx 0.937 \times T_{\frac{1}{2}} \sqrt{\frac{Rd}{nQ\beta}}. \tag{3.63}$$

This relationship predicts the calculated half-widths from Figure 3.14 to within 2%, and is accurate to within 5% for the full range of parameters consistent with earth rheologies. All linear and non-linear Arrhenius calculations with effective $z_0 < 0.4$ (equivalent to $Q > 100$kJ mol^{-1} for $\beta > 10$°C km^{-1}) are plotted against this predicted width in Figure 3.16, which shows the departure from equation (3.63) at large z_0 (outside the range of reasonable parameters). For most of the parameter regime appropriate to the continental crust, $z_0 < 0.2$.

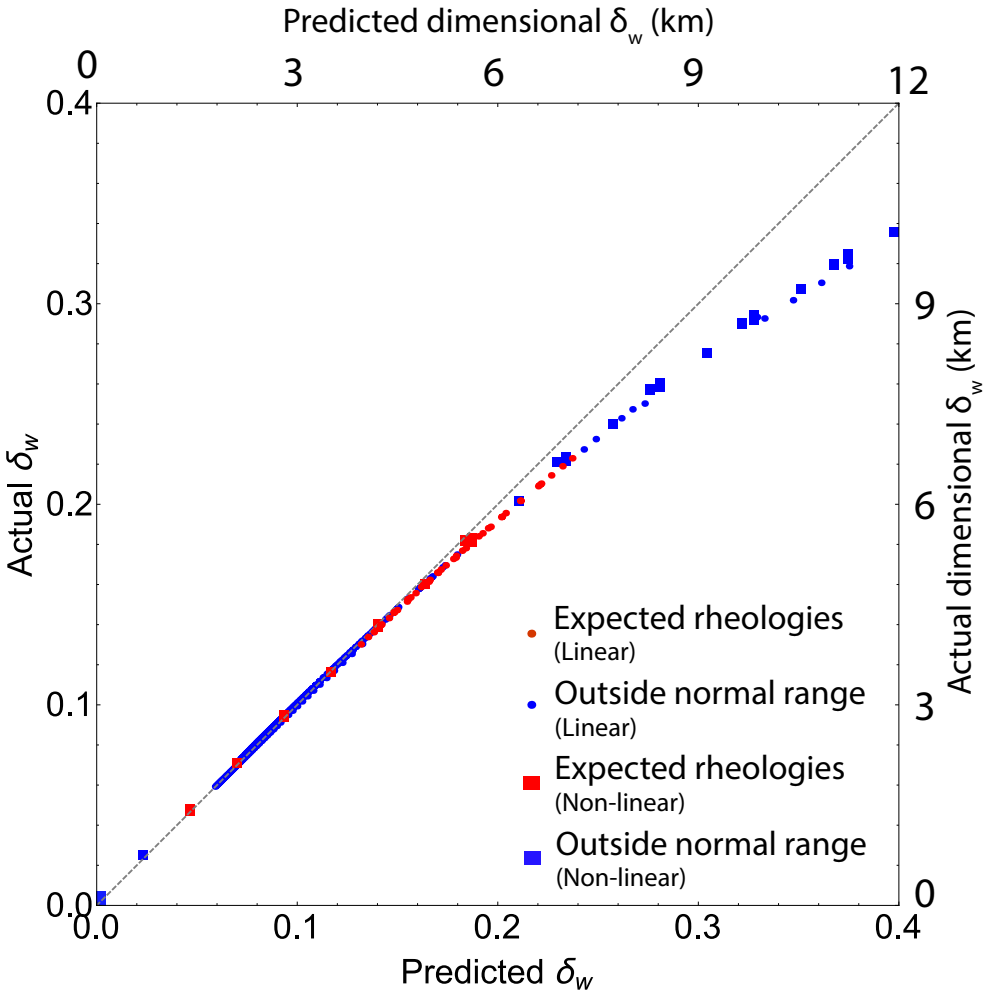


Figure 3.16: Shear zone half width predicted by equation (3.63) plotted against actual half width from all numerical simulations for linear (filled circles) and non-linear Arrhenius rheologies (squares). The total range of parameters are 50kJ$\leq Q \leq$ 3MJ, 5°C km$^{-1} \leq \beta \leq$ 55°C km^{-1} and $1 \leq n \leq 100$. The range of parameters in the earth we might expect are 200kJ$\leq Q \leq$ 550kJ, 10°C km$^{-1} \leq \beta \leq$ 35°C km^{-1} and $1 \leq n \leq 4$, these are plotted in red whilst other simulations are plotted in blue.

The introduction of shear-stress heating and thermomechanical coupling is important in the cases where the shear zone width is already narrow due to the depth dependence of viscosity and a non-linear stress strain-rate relation. The addition of both a non-linear rheology and shear stress heating to the depth dependence due to the geothermal gradient is capable of producing a shear zone of only 1.7km. Depending on the choice of rheological parameters, the viscosities may already be low, in which case the change in temperature due to shear-stress heating is also relatively low and the reduction in viscosity correspondingly smaller. Figure 3.17 illustrates the effective viscosities for each of the parameter sets and physical mechanisms from Figure 3.14 relative to an Arrhenius temperature dependence.

The combined power law and Arrhenius dependence shows a large reduction in viscosity in the immediate vicinity of the shear zone, which returns to a background uniform depth dependence outside of the shear zone. The reductions here are significantly more localised than those generated by shear-stress heating, which produce greater changes in viscosity but over a broader region. The gradients in viscosity, which are the primary control on the shear zone width, are not as large. For both the non-linear rheology and the non-linear rheology with shear stress heating, the viscosities are reduced in a relatively narrow vertical region approximately parallel to the down dip extension of the fault (Figure 3.17). This reduction in viscosities directly underneath the fault should be resolvable when examining postseismic displacements. To construct an earthquake cycle model using a Maxwell viscoelastic rheology consistent with both interseismic and postseismic observations of displacement around the western North Anatolian Fault Zone, *Yamasaki et al.* (2014) required the addition of a rectangular region of low viscosity directly beneath the fault, with relatively sharp boundaries, to explain the amplitude of the post-seismic transient in the near-field ($<$ 30km from the fault). My results suggest that a non-linear thermomechanically coupled rheology could generate such a region, with the effect becoming more pronounced as the shear zone approaches steady state. The non-linear rheology would ensure relatively sharp boundaries, whilst the thermomechanical coupling would localise the low viscosity zone

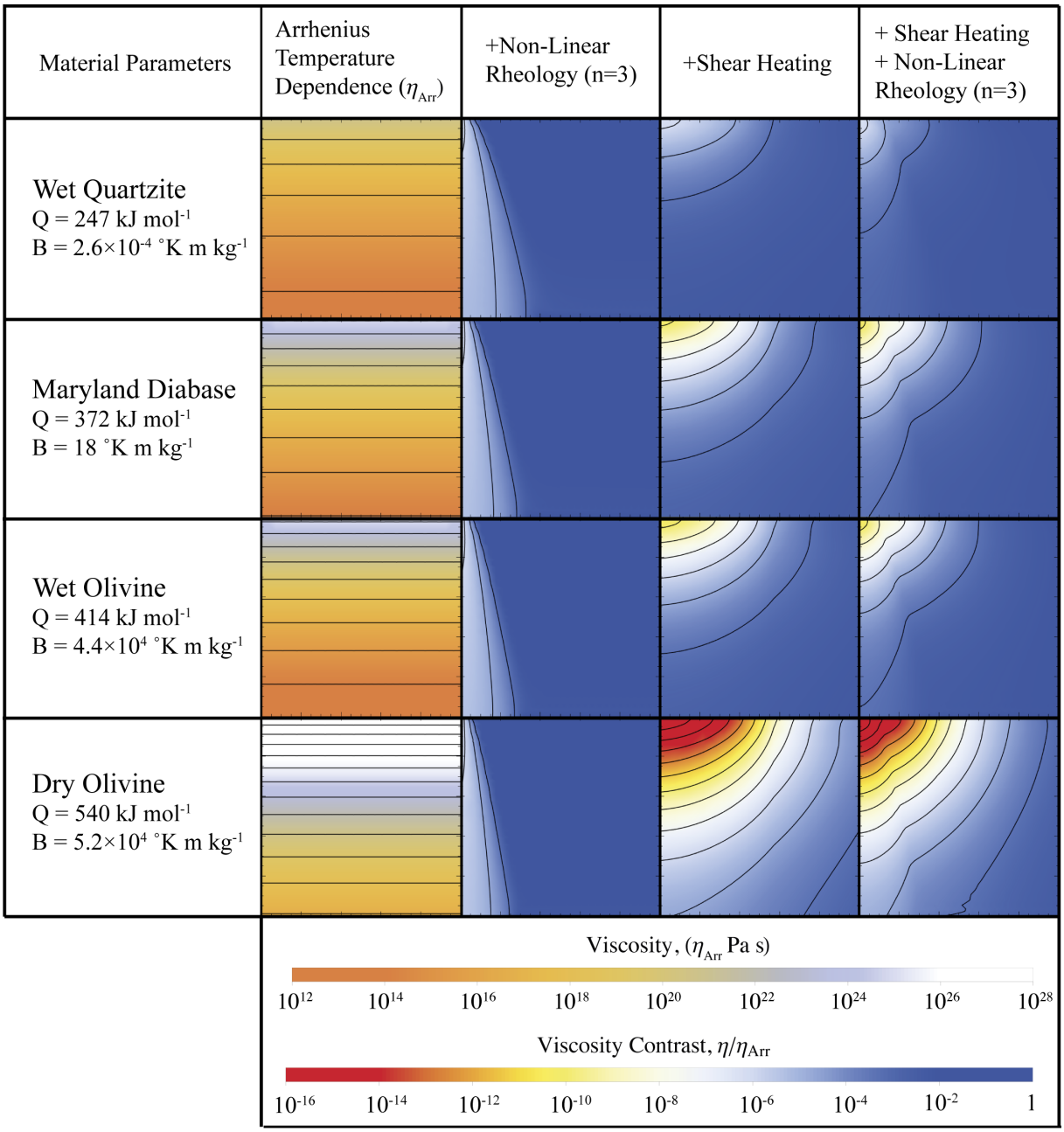


Figure 3.17: Comparison of viscosity fields for given material parameters (*Hirth and Kohlstedt* (2003),*Hirth et al.* (2001)). Arrhenius temperature dependence is shown as absolute viscosities, while additional mechanisms are plotted as viscosity change compared to background Arrhenius temperature dependence. Each square plot has height and width d, whilst the rectangular plots have width $2d$.

to a region surrounding the fault tip (Figure 3.17). A non-linear rheology alone produces a vertically extended zone of reduced viscosity, which would not fit the data from

the North Anatolian Fault Zone (*Yamasaki et al.* (2014)).

The effect of strain localisation due to shear stress heating can only be included in the case of the Arrhenius rheology as temperature dependence is required for the feedback mechanism. Practically, this means that the shear zone has already been localised significantly due to the depth dependence of the viscosity before thermomechanical coupling is considered. The additional reduction in shear zone width is dependent upon this initial width and in most cases only provides between 5 and 20% of additional narrowing, producing shear zone half-widths of 3-6km for linear, or 1.7-3.5km for non-linear rheologies ($n = 3$). However, this reduction in shear zone width would be less pronounced if heat were advected away from the fault, such as in the case where the fault is allowed to migrate laterally, though the depth dependence of viscosity will still generate a narrow shear zone.

Our shear zone half-widths are consistent with those published in *Thatcher and England* (1998), where their non-dimensional constant is equivalent to the reciprocal of our length scale ($Q/(RT_{Max}) \approx 1/z_0$). After conversion their results scale as $\delta_w \approx 1.6\sqrt{z_0}$, with the larger coefficient a direct result of their definition of shear zone half-width $u(\delta_w, 1) = 0.8$. Their work attributed the localisation of shear to thermomechnical coupling, while *Takeuchi and Fialko* (2012) attributed the localisation of shear to both thermomechanical coupling and a non-linear rheology. In considering a linear Maxwell rheology, *Takeuchi and Fialko* (2012) restricted their models to the case of constant viscosity only, where they found no significant localisation. However, in order to exclude a linear Maxwell rheology, it is first necessary to consider the effect of the depth dependence of viscosity. I include viscosity variations with depth in our linear model and suggest that the primary control on the localisation of shear beneath a strike-slip fault is the dependence of the viscosity on the geothermal gradient, with secondary contributions from non-linear creep mechanisms and shear-stress heating.

The main assumption of our model is that the earthquake cycle can be approximated, over geologic timescales, by the block motion of a rigid lid over a viscous layer. For the case of a pure Maxwell rheology, this approximation is valid. However, it is not

entirely clear in the case of a non-linear rheology that we can neglect the contribution of the build up of elastic strains and treat the solution as a fully relaxed viscoelastic material behaving as a viscous media. *Takeuchi and Fialko* (2012) found that to achieve steady state for a non-linear rheology took considerably more cycles than for any linear rheology, and that a non-linear rheology with uniform properties would lead to an unfeasible buildup of stresses (many GPa) due to the elastic strains, which would not be the case for a temperature dependent rheology. Once their models are re-run with a depth dependent Maxwell rheology, I can directly compare their time dependent earthquake cycle model with our approximation. I will also address this point in the next chapter.

When discussing creep mechanisms, the dependence of diffusion creep upon grain size should not be neglected. Cataclasis and dynamic recrystalisation (*Goodwin and Wenk* (1995)) are both mechanisms that can cause a reduction in grain size in mylonites (*Sammis et al.* (1986), *Biegel et al.* (1989), *Marone and Scholz* (1989), *Blenkinsop* (1991), and *Ray* (1999)). The effective viscosity is proportional to the grain size, $\eta \propto d^m$, with the exponent m between 0.7 and 3 (*Gueydan and Précigout* (2014)). The equilibrium grain size depends on both the shear-strain rate ($d \propto 1/\dot{\epsilon}$ for non-negligible strain rates) and temperature ($d \propto T$), assuming homogenous mineral composition. I would expect the flow profile and effective viscosities to demonstrate further localisation of shear, and reduced effective viscosities, close to the down dip extension of the fault where the strain-rates are highest. This should act in a similar fashion to including a non-linear stress-strain relation.

Without further calculations, I cannot rule out the possibility of a grain size dependence resulting in a widening of the shear zone at depth. This is possible if the grain size temperature dependence is stronger than the temperature dependence of the Arrhenius law and the strain rates are low. An increase in grain size with depth would result in an increasing viscosity with depth, producing a negative effective z_0. Evaluating equations (3.6) and (3.21) with a negative z_0 produces δ_w greater than that for a constant viscosity ($\delta_w = 0.5611$). This contribution will be limited by pinning,

where the grains are unable to grow beyond a maximum size, and the exact mineral composition will provide a strong control on the grain size evolution.

3.7 Conclusions

The localisation of shear at depth arises directly from the dependence of the viscosity on the geothermal gradient, with secondary contributions from non-linear rheologies and thermomechanical coupling. For a purely depth dependent rheology the range of parameters consistent with continental lithologies all result in shear zones of less than 6km for continental crust less than 40km thick. A non-linear rheology with uniform properties can generate shear zones of $\sim$ 11km; narrower than that of a constant viscosity layer (18km). Simple scaling relationships are derived which can be used to predict shear zone widths for given parameter combinations, these relationships agree with the numerical results for continental rheologies to within 5%. The results of *Yamasaki et al.* (2014) indicate that a region of low viscosity beneath the rupturing fault is required to fit both pre-seismic and post-seismic rates of deformation on the North Anatolian Fault, and our results suggest that a non-linear rheology plus thermomechnical coupling could produce such a reduction in viscosity in this region. The narrowest shear zones from our model include contributions from thermomechanical coupling, a non-linear rheology, and the depth dependence of viscosity to produce shear zones of 1.7 - 3.5km for a wide range of parameter choices, significantly narrower than the 17km shear zone produced by a constant viscosity layer. Nonetheless, the primary control on shear zone width remains that of the dependence of the viscosity on the geothermal gradient, especially as that is a pre-requisite for significant shear-stress heating.

Chapter 4

Viscoelastic post-seismic relaxation with a non-linear stress strain-rate relationship

4.1 Introduction

Geodetic observations of post-seismic deformation have revealed behaviour over multiple timescales that cannot be explained by a classic Maxwell viscoelastic rheology. Deformation exhibits viscous displacements dependent on the timescale of observation, with relatively focused rapid transient motion in the immediate post-seismic period followed by a longer period of relaxation at greater distances from the fault. Interpretations to explain this phenomena have included depth dependent viscosity (*Yamasaki and Houseman* (2012a)), bi-viscous or Burgers rheologies (*Pollitz* (2005)), more complex non-linear rheologies with poro-elastic rebound (*Barbot and Fialko* (2010)), afterslip governed by rate and state friction laws (*Scholz* (1998)), and full earthquake cycle models with multiple physical mechanisms (*Takeuchi and Fialko* (2013)). InSAR measurements of post-seismic deformation at Manyi, Tibet, following a M_w7.6 in 1997 are indicative of a depth dependent viscosity (*Yamasaki and Houseman* (2012b)), but observations at Izmit, Turkey in 1999 require a low viscosity zone surrounding the

down-dip extension of the fault (*Yamasaki et al.* (2014)).

Long term viscous calculations (Chapter 3) can account for the formation of a persistent zone of weakness beneath a fault, but do not capture the complete sequence of time-dependent behaviour during the earthquake cycle, such as apparent localised weakening in the immediate post-seismic deformation. However, a viscoelastic non-linear rheology could account for localised transient reductions in viscosity due to peaks in local stresses immediately following a coseismic event. In this chapter, I investigate the effect of a power-law stress strain-rate relationship, or dislocation creep, on post-seismic deformation. These models build on analytical Maxwell viscoelastic post-seismic relaxation and earthquake cycle models (*Nur and Mavko* (1974); *Savage and Prescott* (1978)) which I compare to my numerical calculations using the finite element code *oregano_ve*, described in section 2.4.

4.2 Model

The model construction follows section 3.2; I use an idealised strike-slip fault of infinite length. The fault is embedded in an elastic lid over a viscoelastic layer, with displacements restricted to the along-strike direction, such that the region of interest becomes a vertical cross section taken perpendicular to the fault (Figure 4.1). This problem requires a numerical method of solution, for which I use *oregano_ve* as documented in 2.1. As *oregano_ve* is a 3D finite deformation program, I enforce anti-plane conditions ($\dot{\epsilon}_{xx} = \dot{\epsilon}_{yy} = \dot{\epsilon}_{zz} = \dot{\epsilon}_{xy} = 0$, where $\dot{\epsilon}_{ij}$ is the strain rate tensor) with a periodic boundary condition in the z-direction (in *oregano_ve* the direction y is defined positive upwards, with z being the along-strike direction) and

$$u_x(-w) = u_x(w) = 0, \tag{4.1a}$$

$$u_y(0) = u_y(d + t_e) = 0. \tag{4.1b}$$

In the viscoelastic layer, I employ a power-law stress-strain rate relation ($\dot{\epsilon} \propto \sigma'^n$), with effective viscosity

$$\eta_n = A(\sigma'_{ij}\sigma'_{ij})^{\frac{1-n}{2}}. \tag{4.2}$$

The evolution of the system is then governed by the force balance equation

$$\frac{\partial \tau_{ji}}{\partial x_j} = -\rho g_i, \tag{4.3}$$

and viscoelastic constitutive relation,

$$\dot{\epsilon} = \frac{1}{\mu}\dot{\sigma} + \frac{1}{\eta_n}\sigma, \tag{4.4}$$

with effective viscosity defined in equation (4.2).

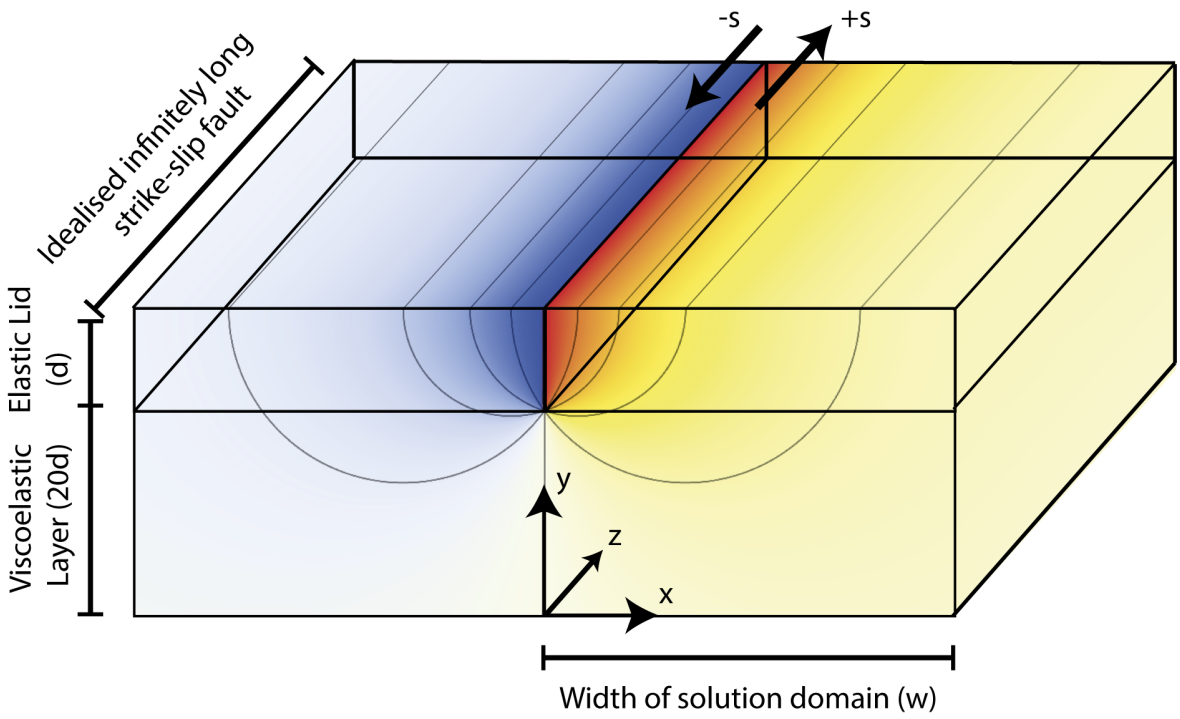


Figure 4.1: Schematic of the general post-seismic model: strike slip fault in an elastic lid of thickness d over a viscoelastic layer of thickness $\gg d$. Contours and colour scheme are illustrative of elastic deformation associated with a co-seismic rupture, the arctan solution discussed in Section 1.1.1.

To investigate post-seismic transients with a non-linear rheology, I examine four

related problems:

- Single rupture with free slip condition on far-field boundary,
- Single rupture with driving velocities on far-field boundary,
- Earthquake cycle with free slip condition on far-field boundary,
- Earthquake cycle with driving velocities on far-field boundary.

4.3 Free slip condition on far-field boundary

The first model to examine incorporates the displacements driven by a single earthquake rupture, with a free slip condition on the boundary

$$\sigma_{xz}(x = w) = 0, \tag{4.5a}$$

$$\sigma_{xz}(x = -w) = 0. \tag{4.5b}$$

The problem is set up in *oregano_ve* with box width $w = 5$ and depth $d = 20$, to approximate an infinite half space ($20d \gg d$). The fault is allowed to rupture at $t = 0$ for a depth range of $0 < y < 1$ with fault offset of 1. For *oregano_ve* jobfiles, see Appendix C. Before interpreting the results of the numerical simulations, I calculated the effective viscosities that would exist at $t = 0$ due to elastic stresses from the coseismic rupture. Deviatoric stress is dependent upon the elastic strain as

$$\sigma'_{ij} = 2\mu\epsilon^e_{ij}. \tag{4.6}$$

Combining equations (4.4) and (4.6) gives us the effective non-linear viscosity

$$\eta_n = A(8\mu^2\epsilon^e_{ij}\epsilon^e_{ij})^{\frac{1-n}{2}}. \tag{4.7}$$

To calculate the effective viscosity following a coseismic rupture on a strike slip fault, I assume anti-plane conditions with slip on an infinite fault, thus $\epsilon_{xx} = \epsilon_{yy} = \epsilon_{zz} =$

$\epsilon_{yz} = 0$. Using the elastic solution for displacements due to such an event,

$$u_x(y,z) = \frac{s}{2\pi}\left(\arctan\left(\frac{y}{z+d}\right) - \arctan\left(\frac{y}{z-d}\right)\right), \tag{4.8}$$

with elastic strains

$$\epsilon^e_{xz} = \frac{1}{2}\frac{\partial u_x}{\partial z} = \frac{dsyz}{\pi(y^2+(d-z)^2)(y^2+(d+z)^2)}, \tag{4.9a}$$

$$\epsilon^e_{xy} = \frac{1}{2}\frac{\partial u_x}{\partial y} = \frac{ds(d^2+y^2-z^2)}{2\pi(y^2+(d-z)^2)(y^2+(d+z)^2)}. \tag{4.9b}$$

Using the elastic strains from equations (4.9), I calculate $\epsilon^e_{ij}\epsilon^e_{ij}$,

$$\epsilon^e_{ij}\epsilon^e_{ij} = \frac{d^2s^2}{4\pi^2\left((d-z)^2+y^2\right)\left((d+z)^2+y^2\right)}. \tag{4.10}$$

The final effective viscosity is thus given by equations (4.4) and (4.10),

$$\eta' = A\pi^2\left[\frac{2\mu^2d^2s^2}{\left((d-z)^2+y^2\right)\left((d+z)^2+y^2\right)}\right]^{\frac{1-n}{2}}. \tag{4.11}$$

Equation (4.11) is plotted in Figure 4.2 for $n = (2,3,4,5)$. It should be noted that closer than approximately half the rupture depth to the base of the fault ($d/2$) (dashed green contour in Figure 4.2), the viscosities are lower than the equivalent linear viscosity, whereas further away they increase by several orders of magnitude. From these calculated viscosities, I would expect displacements to be concentrated in this area initially, then diffuse out as the stresses are raised in the surrounding region, hence reducing the viscosities.

Numerical simulations were run in *oregano_ve* for $n = 1.0$ to $n = 5.0$ for $t = 1$ to $t = 100$, where t has been non-dimensionalised by the linear Maxwell time η_0/μ. Figures 4.3 and 4.4 show the post-seismic surface displacements at $t = 50$ and $t = 100$ respectively (co-seismic elastic displacements at $t = 0$ have been subtracted from the solution file). The linear case is fully relaxed by $t = 50$, and the $n = 2$ case has al-

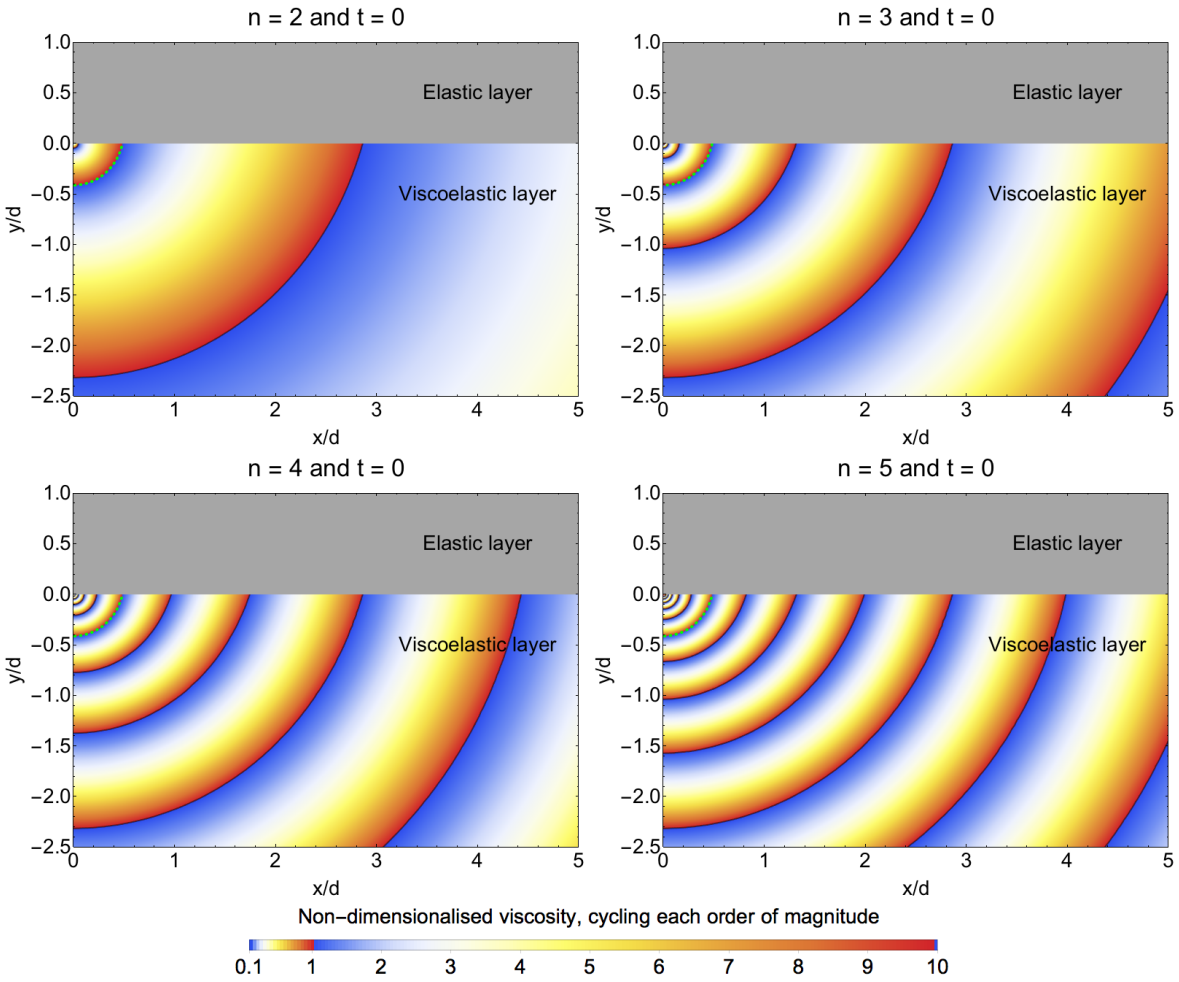


Figure 4.2: Viscosity structures for $n = 2, 3, 4, 5$ at $t = 0$ following co-seismic rupture, where viscosity has been non-dimensionalised by equivalent linear viscosity η_0. Colour scale is logarithmic and cycles every order of magnitude. Contours are each order of magnitude and $\eta = \eta_0$ is plotted as a dashed green contour.

most completely relaxed by $t = 100$. However, higher values of n show little further displacement between $t = 50$ and $t = 100$, especially at distance from the fault. The displacement field at depth is illustrated in Figure 4.8; a general trend of smaller displacements for higher n is visible, with the majority of displacements focussed in the region surrounding the fault tip.

The focussing of displacements in the region near the fault is easier to see in Figures 4.6 and 4.7, where the time series of surface displacements at $x = 1$ and $x = 5$ are

compared for different values of n. The displacement rate at a given point is directly correlated with the viscosity structures beneath that point and displacements at $x = 1$ show a steeper gradient for $n > 1$ than those at $x = 5$. The viscosity structures are illustrated in Figure 4.8, which shows a similar radial pattern to that predicted in Figure 4.2. The range of viscosities in Figure 4.8 are lower than those for $t = 0$ in Figure 4.2 due to the diffusion of stress away from the fault. It is evident from these plots that at $x = 1$ and $t = 100$, the viscosities vary from $(2 - 10)\eta_0$ for $2 \leq n \leq 5$ and $(5 - 40)\eta_0$ at $x = 5$. Thus, whilst a non-linear rheology can support post-seismic transients close to the fault itself, the high viscosities cause the solution to 'lock up' further than approximately $2d$ from the fault. A creep mechanism that generates a non-linear rheology would be dislocation creep; later in this chapter I shall investigate the interplay of both dislocation (non-linear) and diffusion (linear) creep mechanisms.

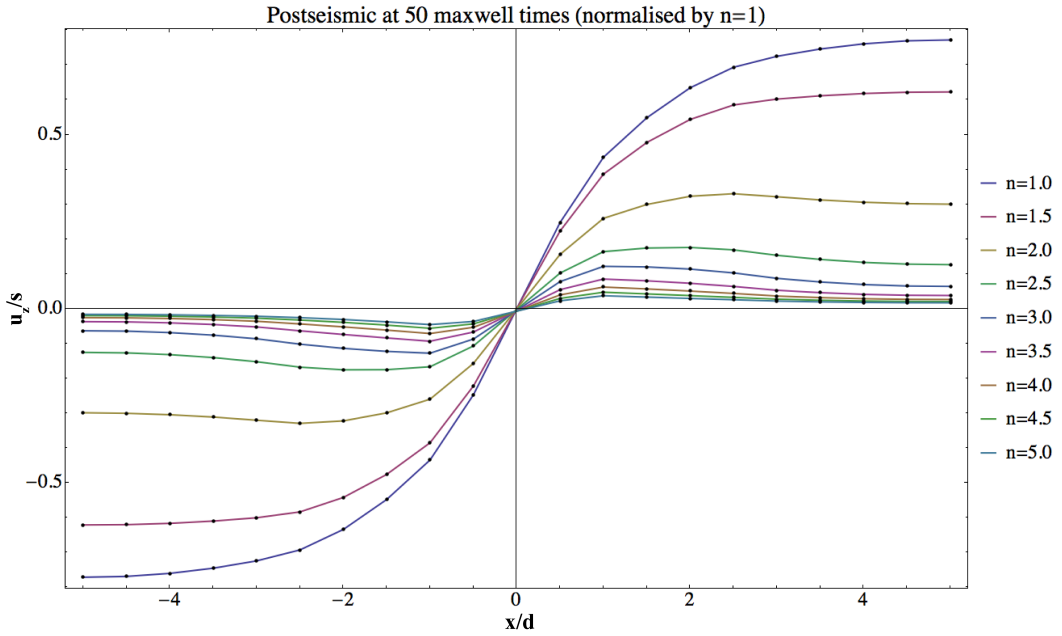


Figure 4.3: Total post-seismic displacements at 50 Maxwell times (for equivalent linear rheology). Displacements are nondimensionalised by slip on the fault s and horizontal distances by rupture depth d.

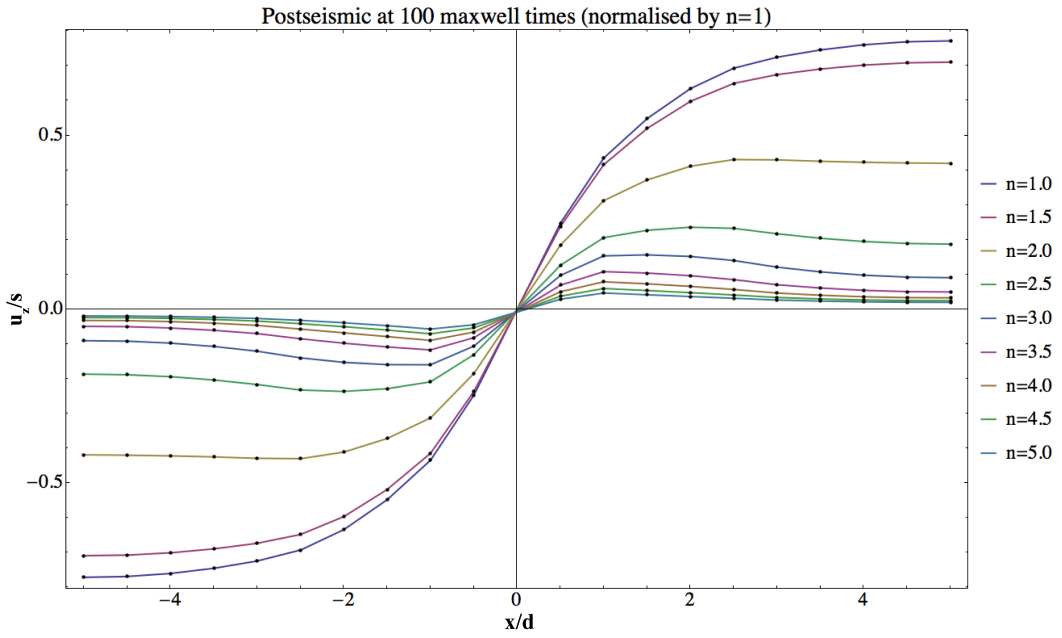


Figure 4.4: Total post-seismic displacements at 100 Maxwell times (for equivalent linear rheology). Displacements are nondimensionalised by slip on the fault s and horizontal distances by rupture depth d.

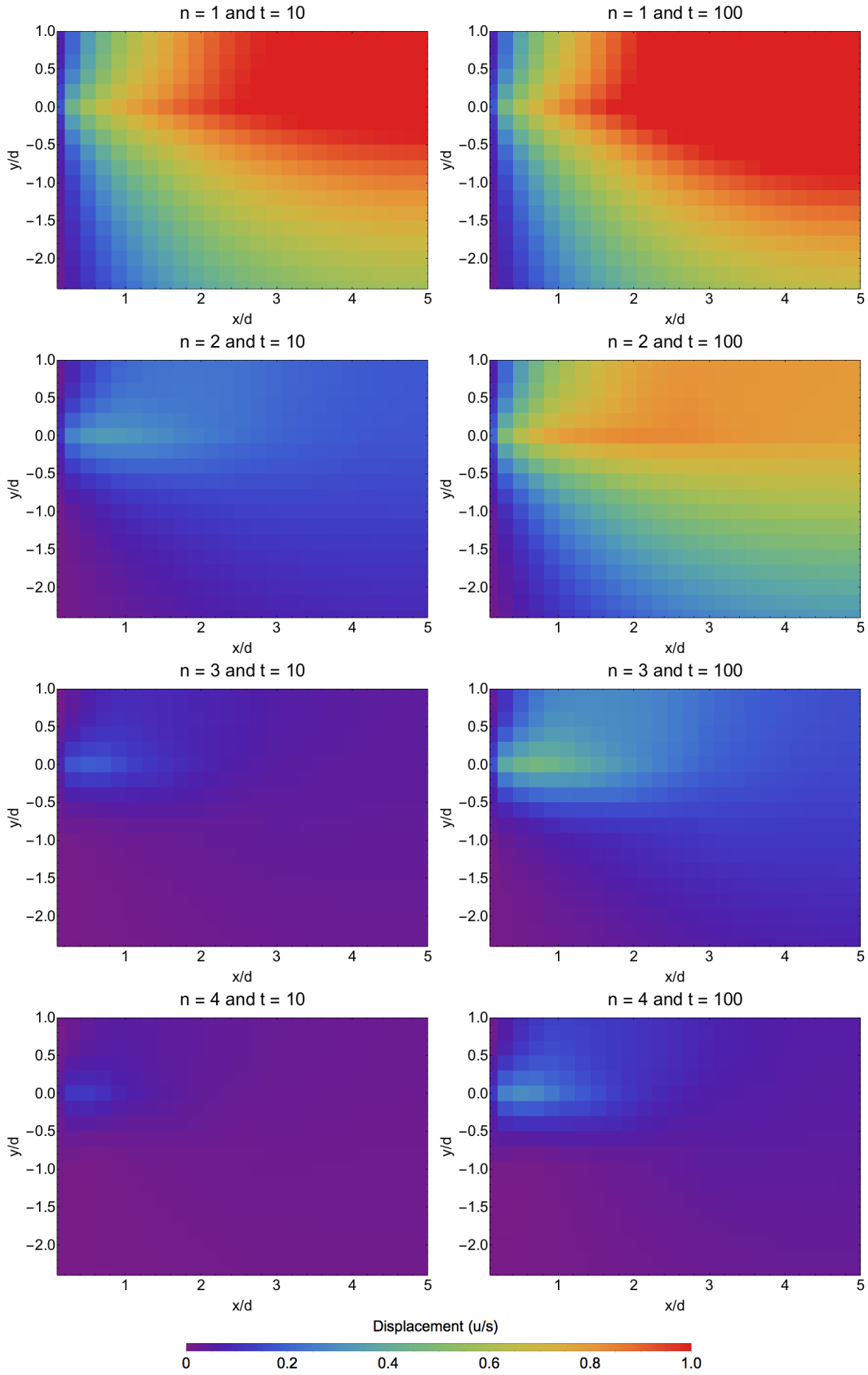


Figure 4.5: Fault parallel post-seismic displacements, with elastic deformation removed. Displacements are nondimensionalised by slip on the fault s and vertical/horizontal distances by rupture depth d. Note the elastic lid is between $0 \leq y \leq 1$, with viscoelastic layer $y < 0$.

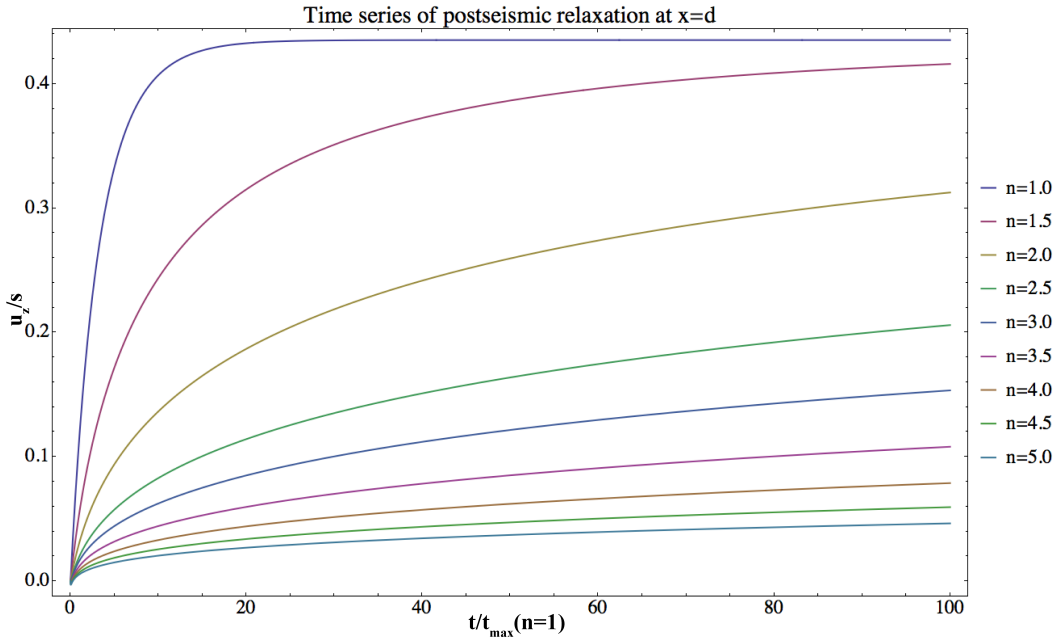


Figure 4.6: Time series of post-seismic displacements at $x = d$. Time nondimensionalised by equivalent linear Maxwell time and displacements by fault slip s. Note that all cases exhibit displacements, though the maximum displacement scales inversely with n.

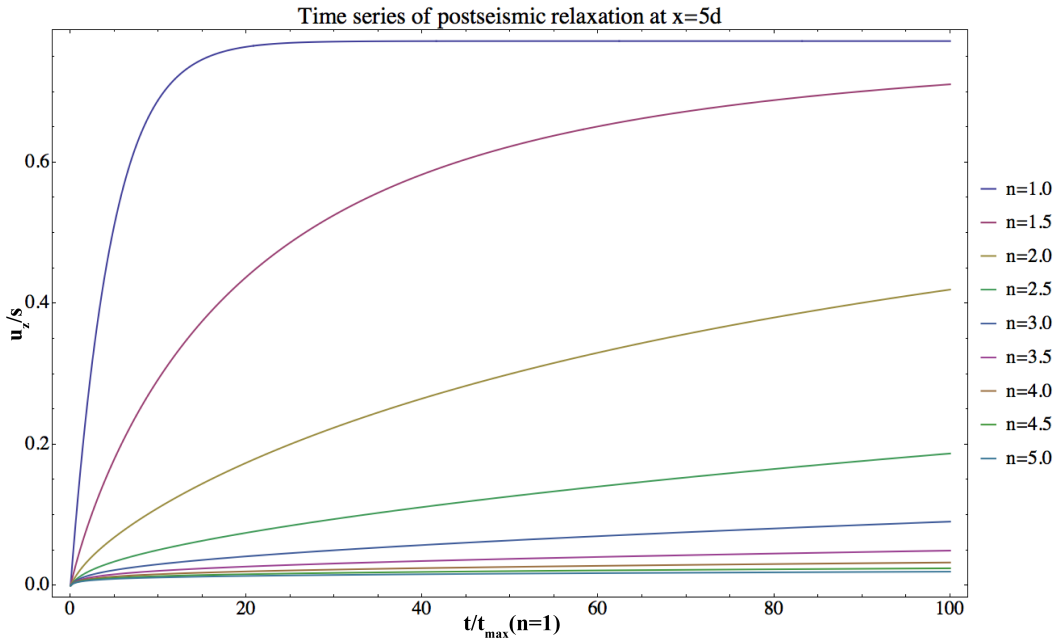


Figure 4.7: Time series of post-seismic displacements at $x = 5d$. Time nondimensionalised by t_{max} and displacements by fault slip s. Note that for higher values of n, little displacement is visible at long times and the timescale of relaxation is longer than in Figure 4.6.

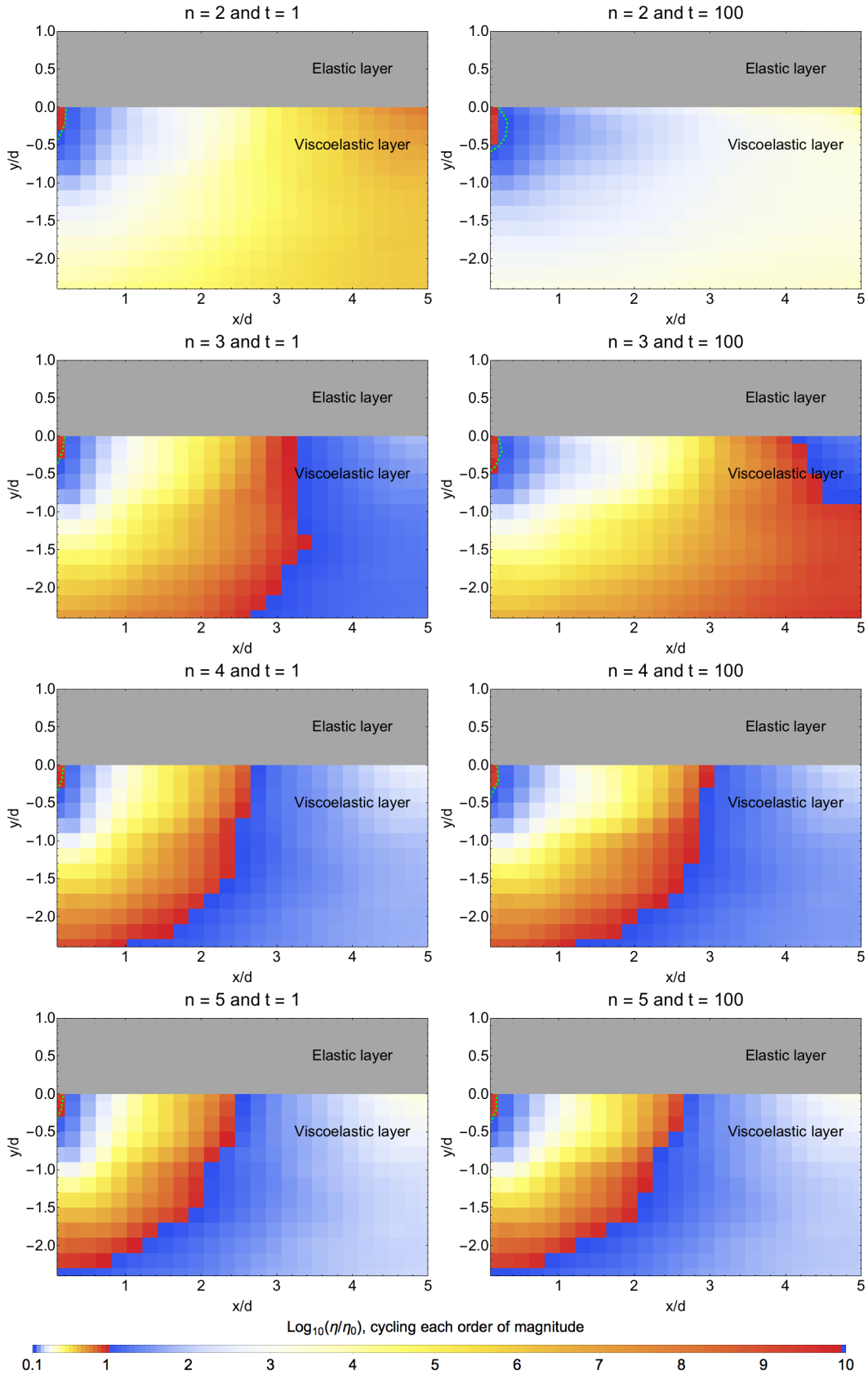


Figure 4.8: Viscosity structures for $n = 2, 3, 4, 5$ where viscosity has been non-dimensionalised by equivalent linear viscosity η_0. Colour scale is logarithmic and cycles every order of magnitude with $\eta = \eta_0$ denoted by the dashed green contour and the elastic lid has been shaded gray. Note the reduction in viscosity below η_0 near the origin, and for $n > 3$ the viscosities in the far field approach $(30 - 60)\eta_0$.

4.4 Loading conditions on far-field boundary

To deal with the unrealistic behaviour of high viscosities causing the post seismic displacements to 'lock up', I introduce a far field loading condition,

$$\dot{u}_z(x = w) = 1, \tag{4.12a}$$

$$\dot{u}_z(x = -w) = -1. \tag{4.12b}$$

The physical interpretation of this constraint is to apply a background constant strain-rate across the domain, reducing the viscosities to values that allow deformation to occur on similar timescales for different values of n. Surface displacements, with the far field loading removed, are illustrated in Figures 4.9 and 4.10. It should be noted that even $n = 5$ is almost fully relaxed at $t = 10$, a time scale an order of magnitude lower than that for the stress free boundary condition (section 4.3). The effect of the interaction between the boundary conditions and box width is investigated in Figures 4.15 and 4.12, where the wider box results in a lower background strain-rate. As the enforced boundary condition is Dirichlet, this pins the far field displacements to the driving velocity and arises as a linear decrease in displacements towards the right hand boundary in Figures 4.15 and 4.12. Again, a higher value of n initially results in reduced displacements away from the fault, but the displacements rapidly approach the linear case as the stresses diffuse outwards.

Looking at the time series of displacements at $x = 1$ and $x = 4$ in Figures 4.13 and 4.14, the linear case behaves as expected and approaches relaxation at an exponentially decreasing rate, but the non-linear case exhibits an s-shaped curve, with an initially increasing displacement rate. The initial increase in displacement rate observed in my model is due to the viscosities reducing as the stresses diffuse out from the coseismic rupture, then displacement continues until the medium is fully relaxed. This has not been observed, and thus puts an upper bound on n or requires a consistent background viscosity (see Section 4.7 for a discussion on multiple creep mechanisms).

The viscosity structures plotted in Figure 4.15 show that after only a few Maxwell times the background viscosity has dropped to less than $5\eta_0$ for $2 \leq n \leq 5$, though a region of low viscosity surrounding the base of the fault persists. It is evident from these plots that the majority of the deformation starts out highly localised then diffuses outwards for this model.

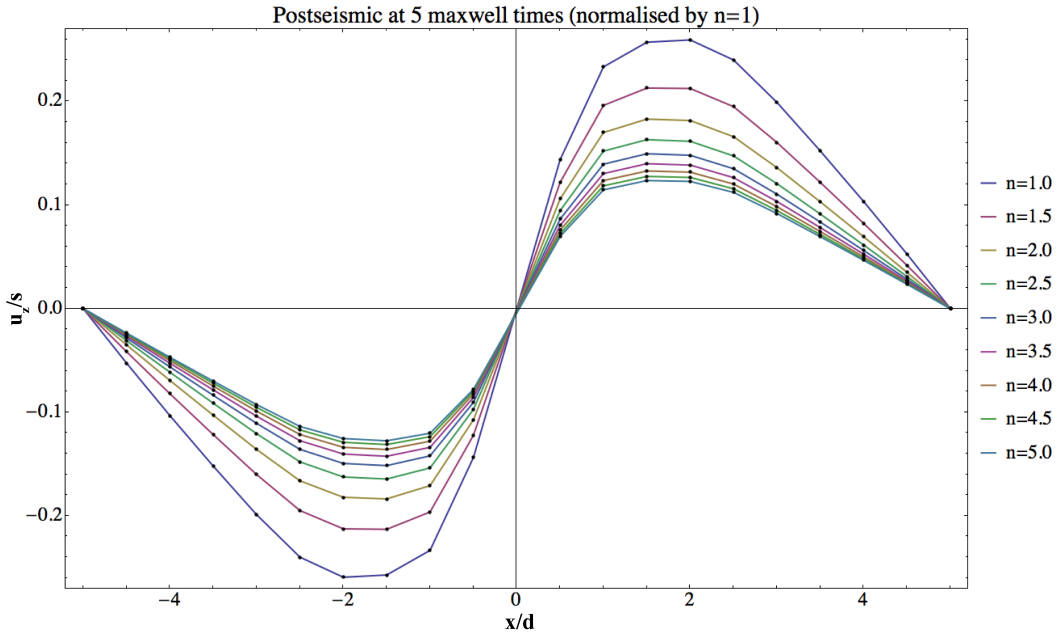


Figure 4.9: Postseismic displacements at 5 Maxwell times (for equivalent linear rheology) with far field loading removed. Displacements are nondimensionalised by slip on the fault s and horizontal distances by rupture depth d.

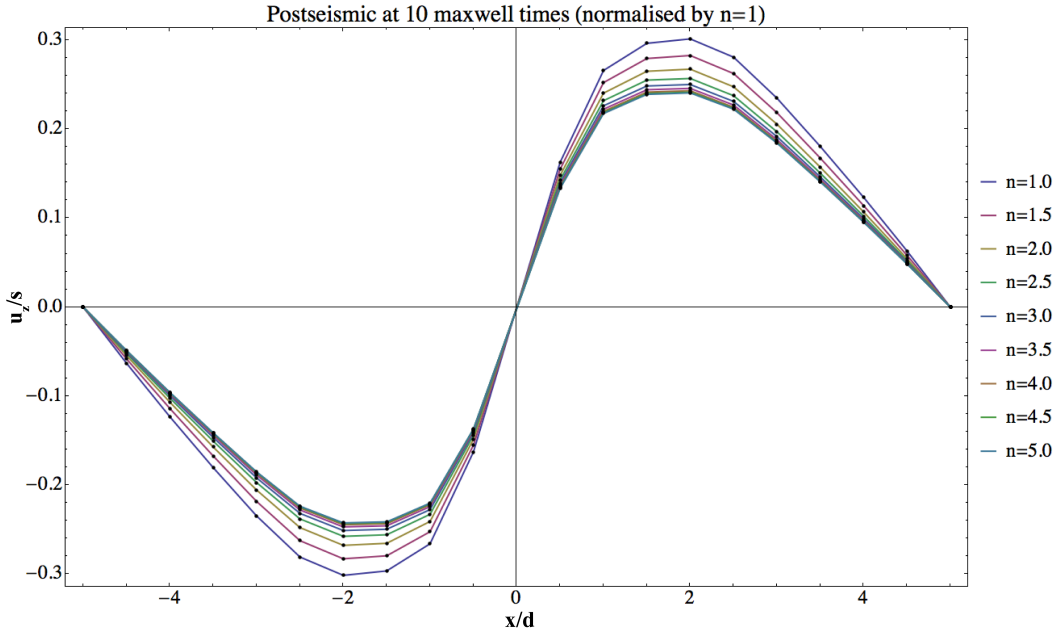


Figure 4.10: Postseismic displacements at 10 Maxwell times (for equivalent linear rheology) with far field loading removed. Displacements are nondimensionalised by slip on the fault s and horizontal distances by rupture depth d.

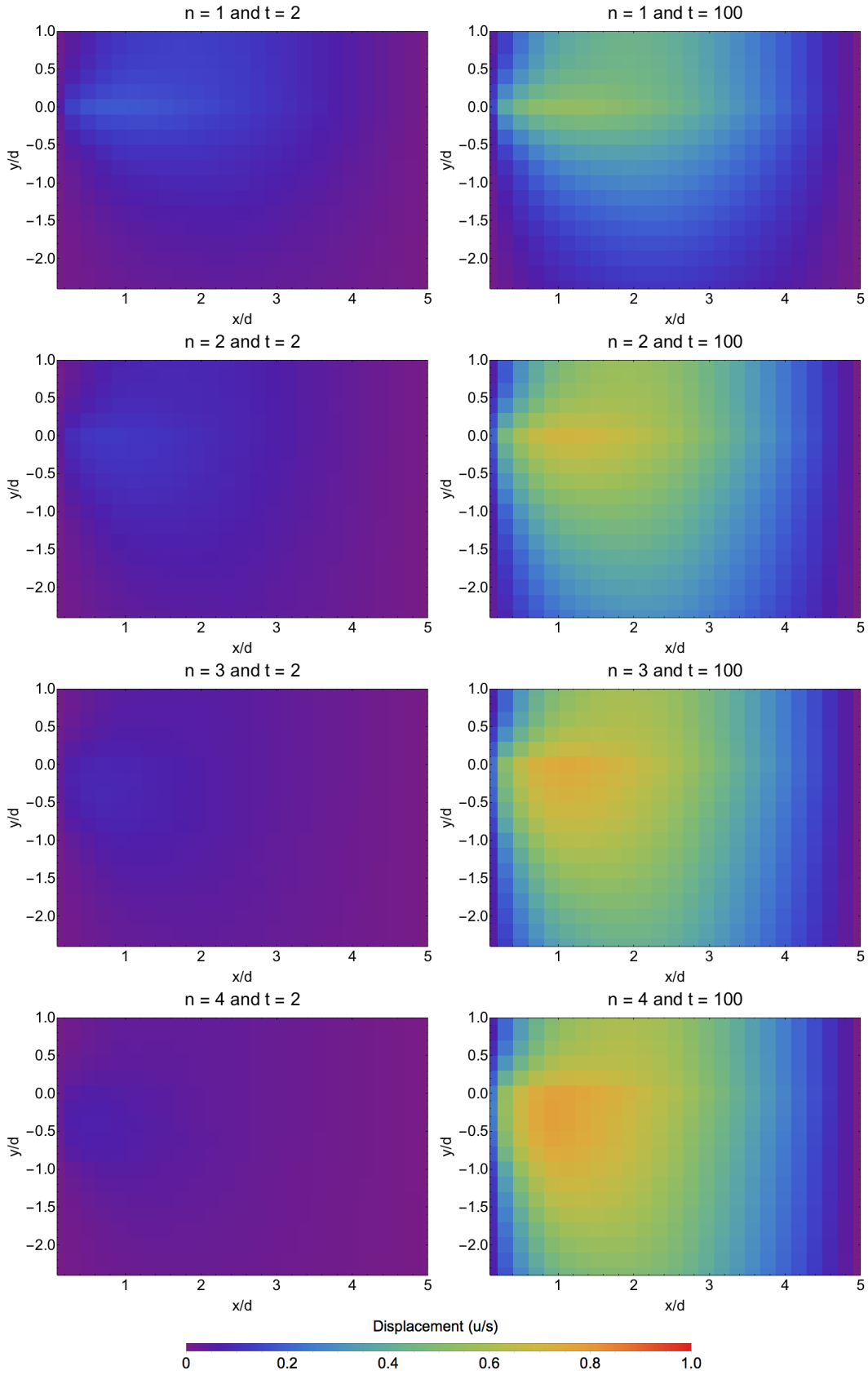


Figure 4.11: Fault parallel post-seismic displacements, with elastic deformation and far field displacements removed, $w = 10$. Displacements are nondimensionalised by slip on the fault s and vertical/horizontal distances by rupture depth d. Note the elastic lid is between $0 \leq y \leq 1$, with viscoelastic layer $y < 0$.

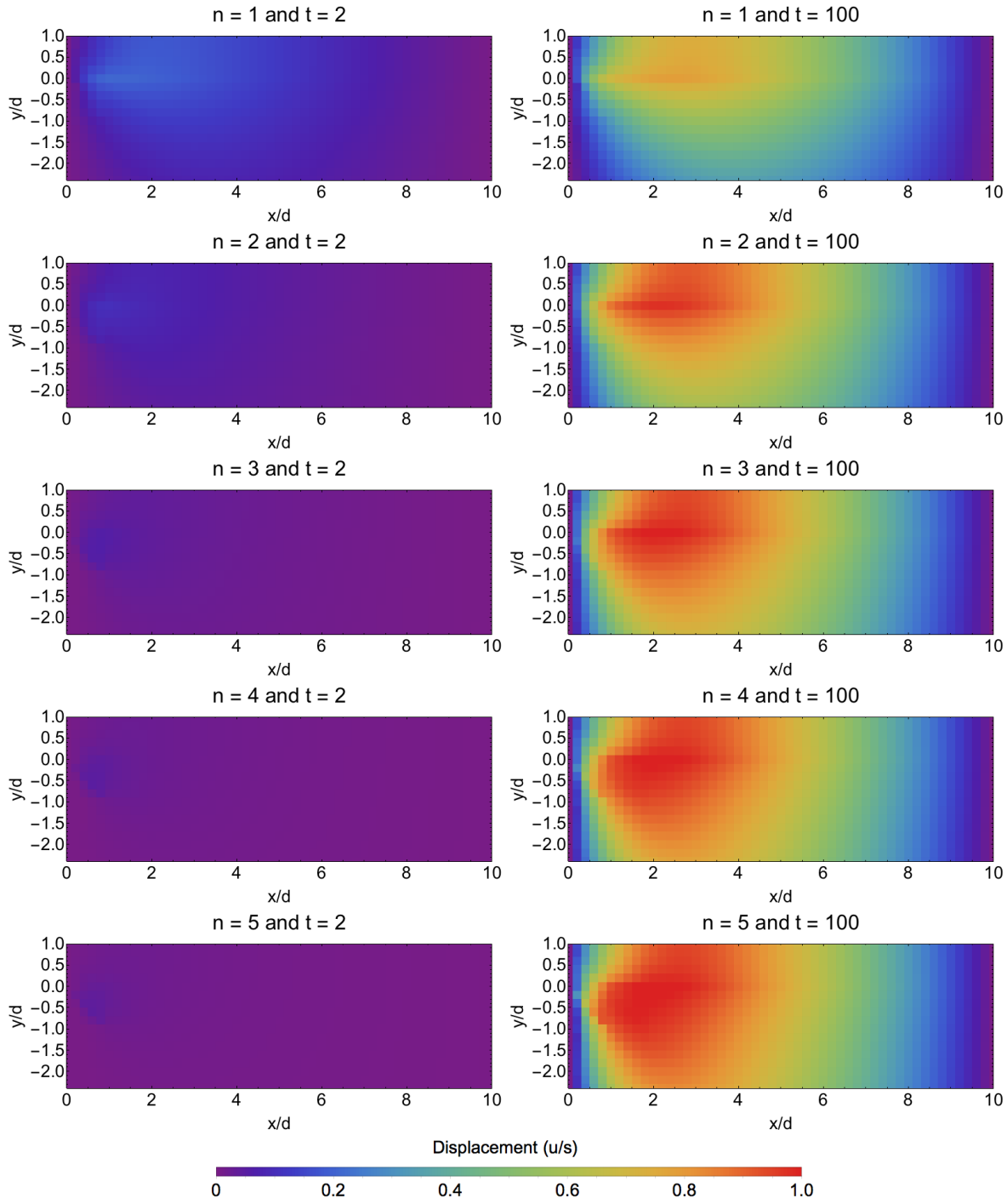


Figure 4.12: Fault parallel post-seismic displacements, with elastic deformation and far field displacements removed, $w = 20$. Displacements are nondimensionalised by slip on the fault s and vertical/horizontal distances by rupture depth d. Note the elastic lid is between $0 \leq y \leq 1$, with viscoelastic layer $y < 0$.

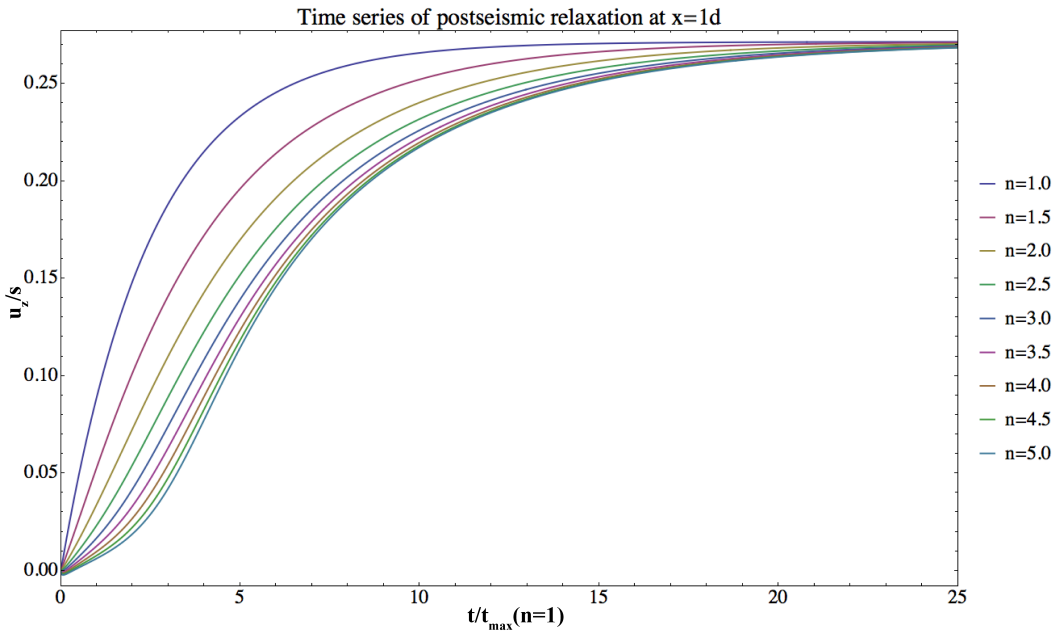


Figure 4.13: Temporal evolution of post-seismic displacements at $x = 1$. Time nondimensionalised by equivalent linear Maxwell time and displacements by fault slip s. Note the distinctive s-shaped curve for the non-linear rheologies that is not present in the linear case.

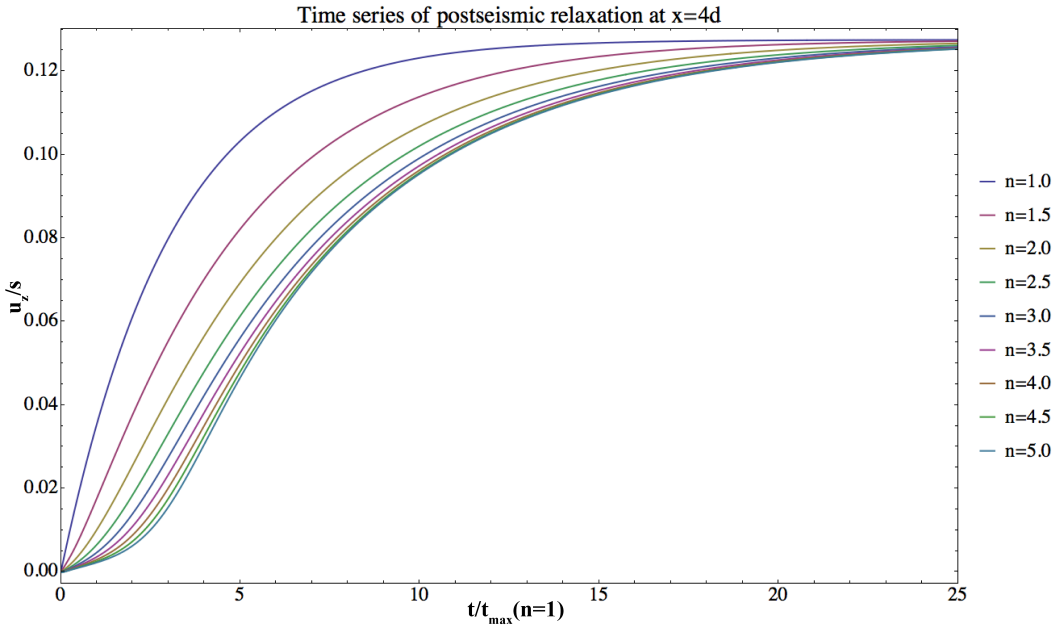


Figure 4.14: Temporal evolution of post-seismic displacements at $x = 4$. Time nondimensionalised by equivalent linear Maxwell time and displacements by fault slip s. At this distance, the stresses take longer to diffuse out, so the initial increase in rates of displacement is slower than for Figure 4.13.

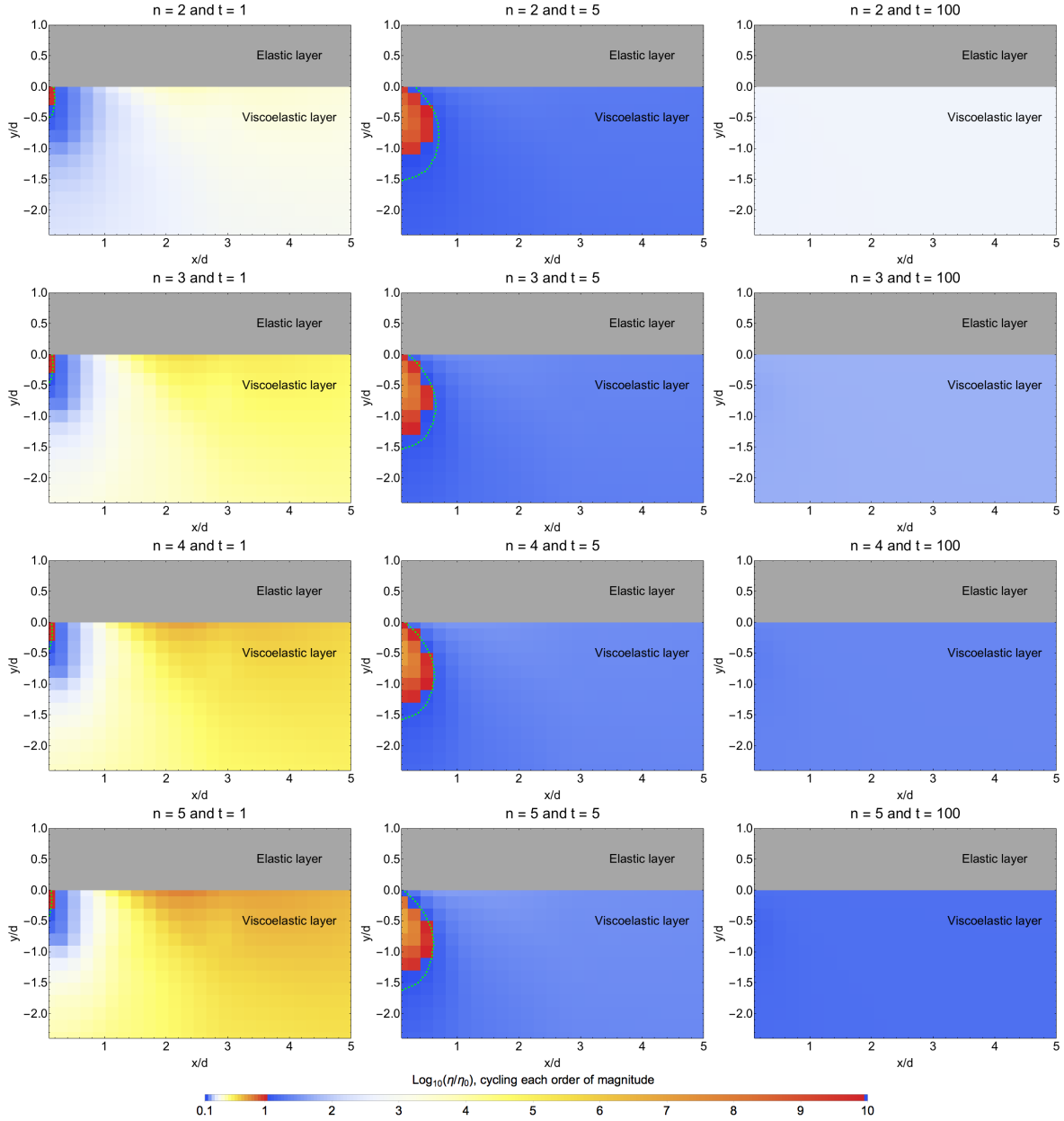


Figure 4.15: Viscosity structures for $n = 2, 3, 4, 5$ where viscosity has been non-dimensionalised by equivalent linear viscosity η_0. Dashed green contour denotes $\eta = \eta_0$ and the elastic lid has been shaded gray. By $t = 100$ the system is fully relaxed and has assumed a uniform viscosity.

4.5 Earthquake cycle

The next increase in complexity for the model is to introduce repeated ruptures, initially with free-slip boundary conditions as given in equations (4.5). A full investigation of parameter space for this problem would have to incorporate the ratio of earthquake repeat time to Maxwell time, but a suite of initial calculations should elucidate the kind of behaviour I would expect. For these calculations, I chose a fairly rapid repeat time of one Maxwell time. For most lithologies this would correspond to a repeat time in the order of decades, so presents an end member case of a strong crust.

The models are run for up to 100 cycles and the velocity fields plotted in Figure 4.16. These plots differ from those in the earlier section in that we are no longer removing the elastic displacements; whilst the plots for $t = 1$ appear similar, there is considerably more post seismic motion for $n = 1$ than for $n > 1$. At $t = 100$, the cycles appear to approach block motion in the elastic lid, though there is a slight linear ramp visible for higher n and a zone of distributed shear develops in the viscoelastic layer. It should be noted that the width of the shear zone is independent of n, in contradiction to the results from Chapter 3. This distinction exists because the shear zones include far field loading which is absent in this model. The far field loading acts to help localise the zone of weakness onto the down-dip extension of the fault, and contribute to the formation of a narrow shear zone.

Similarly, the viscosity fields illustrated in Figure 4.17 rapidly approach steady state after only a few cycles, and show a marked zone of weakness beneath the fault. This region of weakness, like the shear zone width, appears to depends little on n, being approximately $3d$ wide and $2d$ deep.

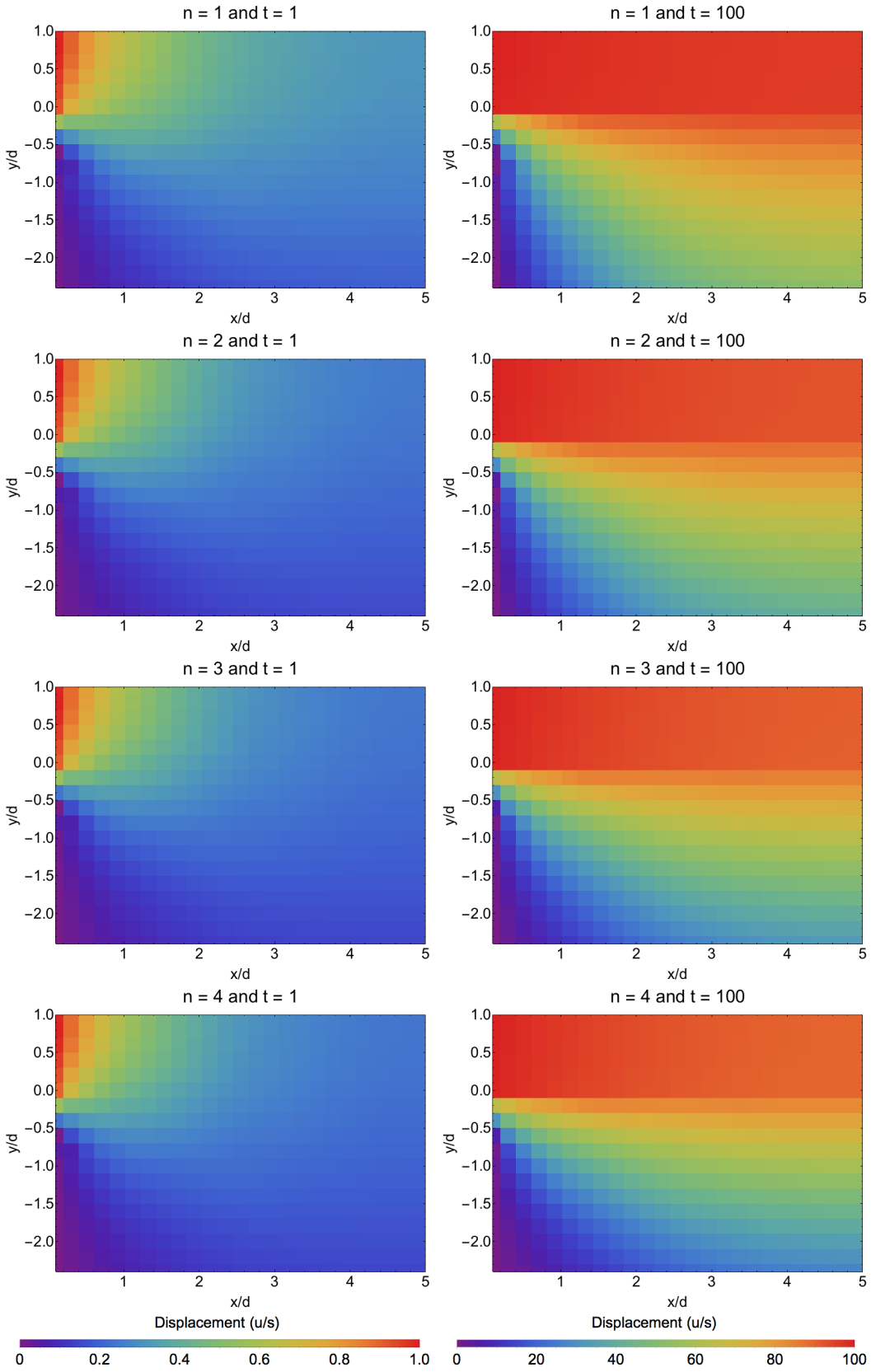


Figure 4.16: Fault parallel post-seismic displacements, including repeated earthquakes and associated co-seismic deformation. Displacements are nondimensionalised by slip on the fault s and vertical/horizontal distances by rupture depth d. Note the elastic lid is between $0 \leq y \leq 1$, with viscoelastic layer $y < 0$.

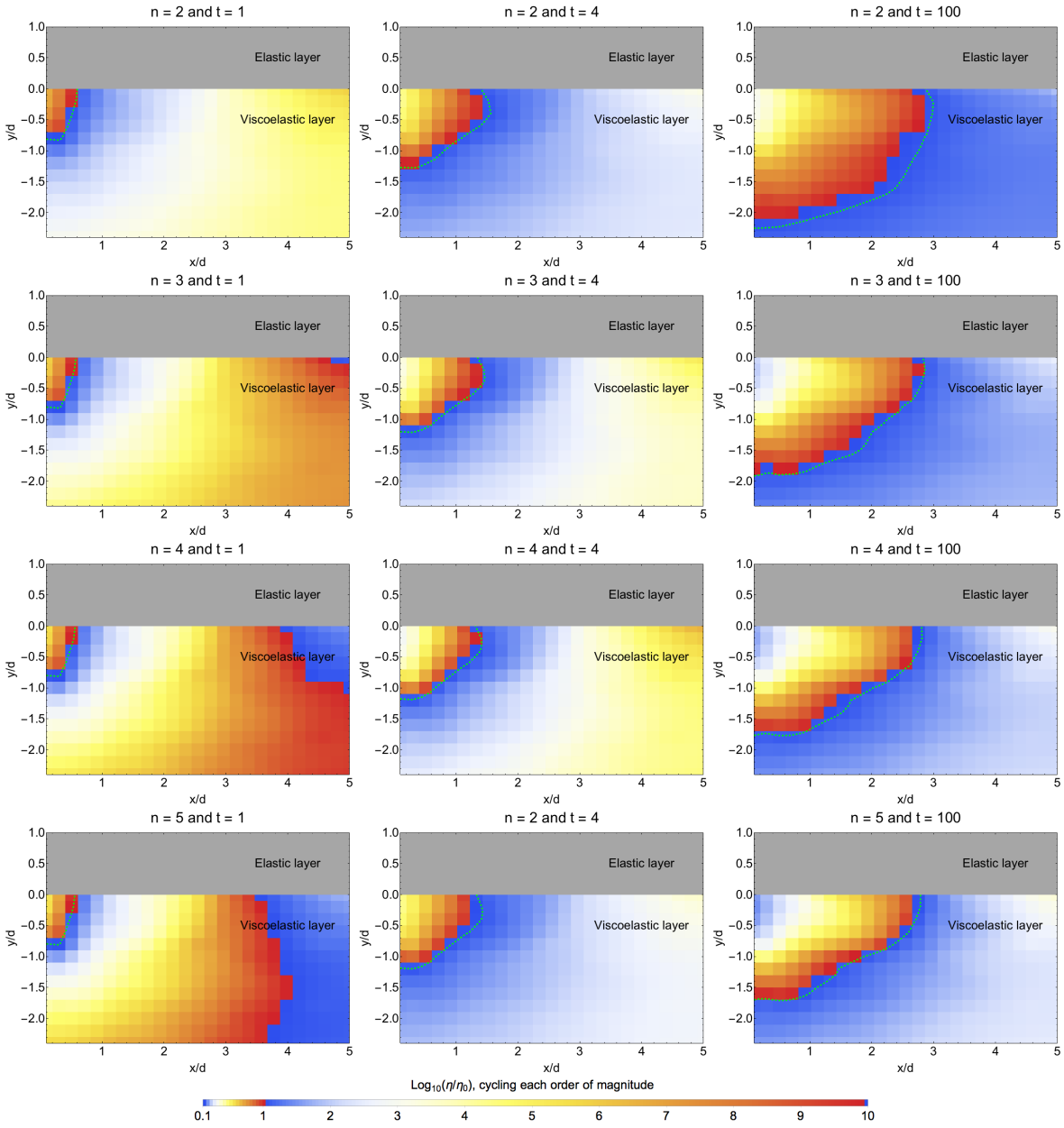


Figure 4.17: Viscosity structures for $n = 1, 2, 3, 4, 5$ where viscosity has been non-dimensionalised by equivalent linear viscosity η_0. Dashed green contour denotes $\eta = \eta_0$ and the elastic lid has been shaded gray.

4.6 Earthquake cycle with driving boundary conditions

The final increase in complexity for this model is repeated ruptures with driving boundary conditions as given in equations (4.12). Again, a full investigation of parameter space for this problem would necessitate incorporation of the ratio of earthquake repeat time to Maxwell time but an initial suite of calculations should reveal the key features of this model. For comparison with the previous section, I again chose a repeat time of one Maxwell time.

The models are run for up to 100 cycles and the velocity fields plotted in Figure 4.18. As with Section 4.5, I do not remove the elastic displacements from the solution file. Unlike the shear-stress free boundary condition, there is little difference in the post seismic motion for small t. At $t = 100$, the cycles have developed block motion in the elastic lid, and a zone of distributed shear forms in the viscoelastic layer. It should be noted that the width of the shear zone is no longer independent of n, and agrees with the findings of Chapter 3 where the width of the shear zone is inversely proportional to the stress exponent n.

Similarly, the viscosity fields illustrated in Figure 4.19 rapidly approach steady state after only a few cycles (< 10), and show a marked zone of weakness beneath the fault that is almost fully developed after 5 cycles. This region of weakness is now vertical in structure, and closely resembles the viscosity structures for a viscous shear zone with power law rheology shown in Figure 3.17. Though more work needs to be done to quantify the relationship between the Maxwell time and the repeat time, these results suggest that all strike-slip faults which have undergone more than ~ 10 ruptures will have developed shear zones on the down dip extension of the fault.

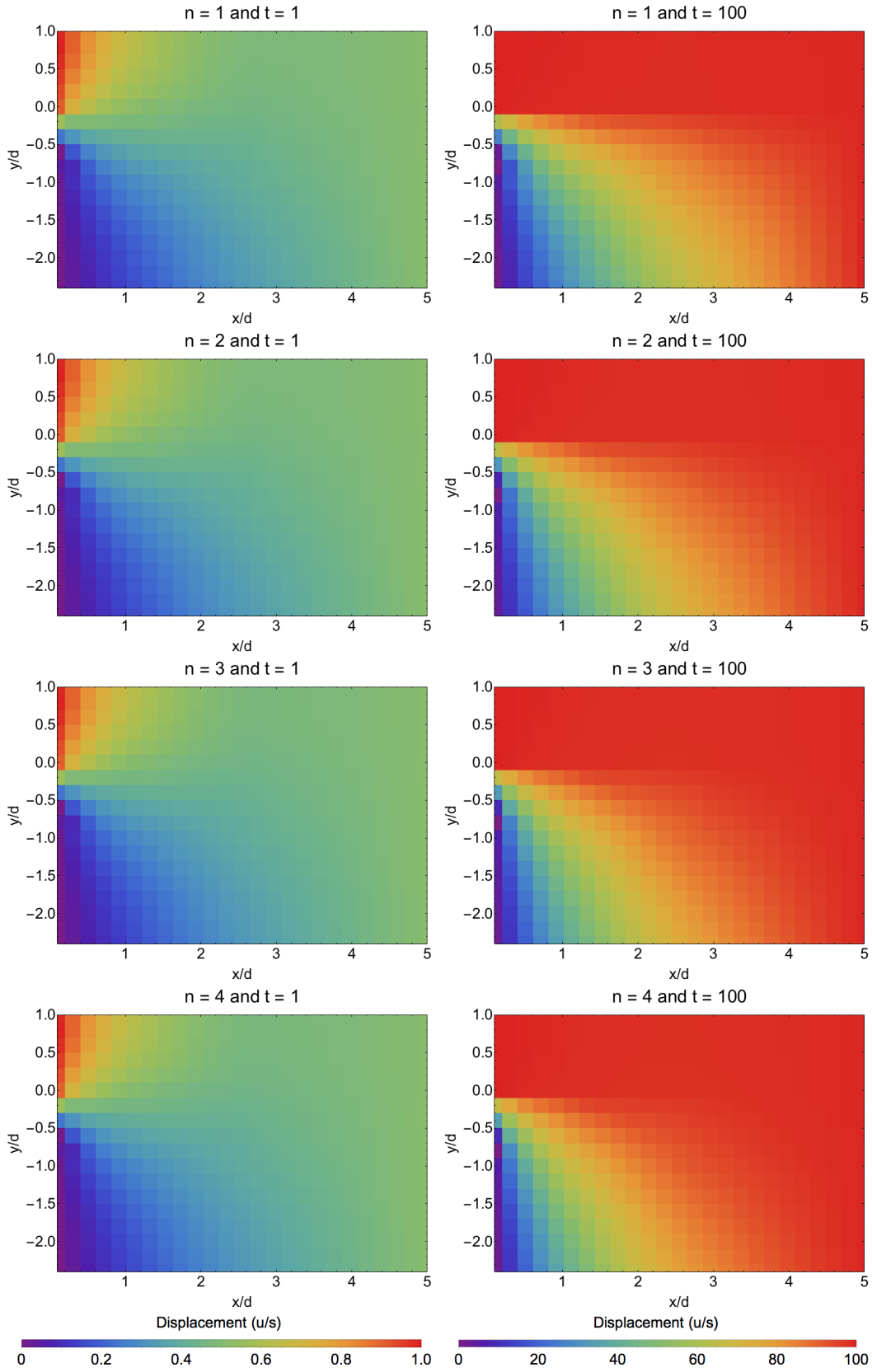


Figure 4.18: Fault parallel post-seismic displacements, including repeated earthquakes and associated co-seismic deformation. Displacements are nondimensionalised by slip on the fault s and vertical/horizontal distances by rupture depth d. Note the elastic lid is between $0 \leq y \leq 1$, with viscoelastic layer $y < 0$.

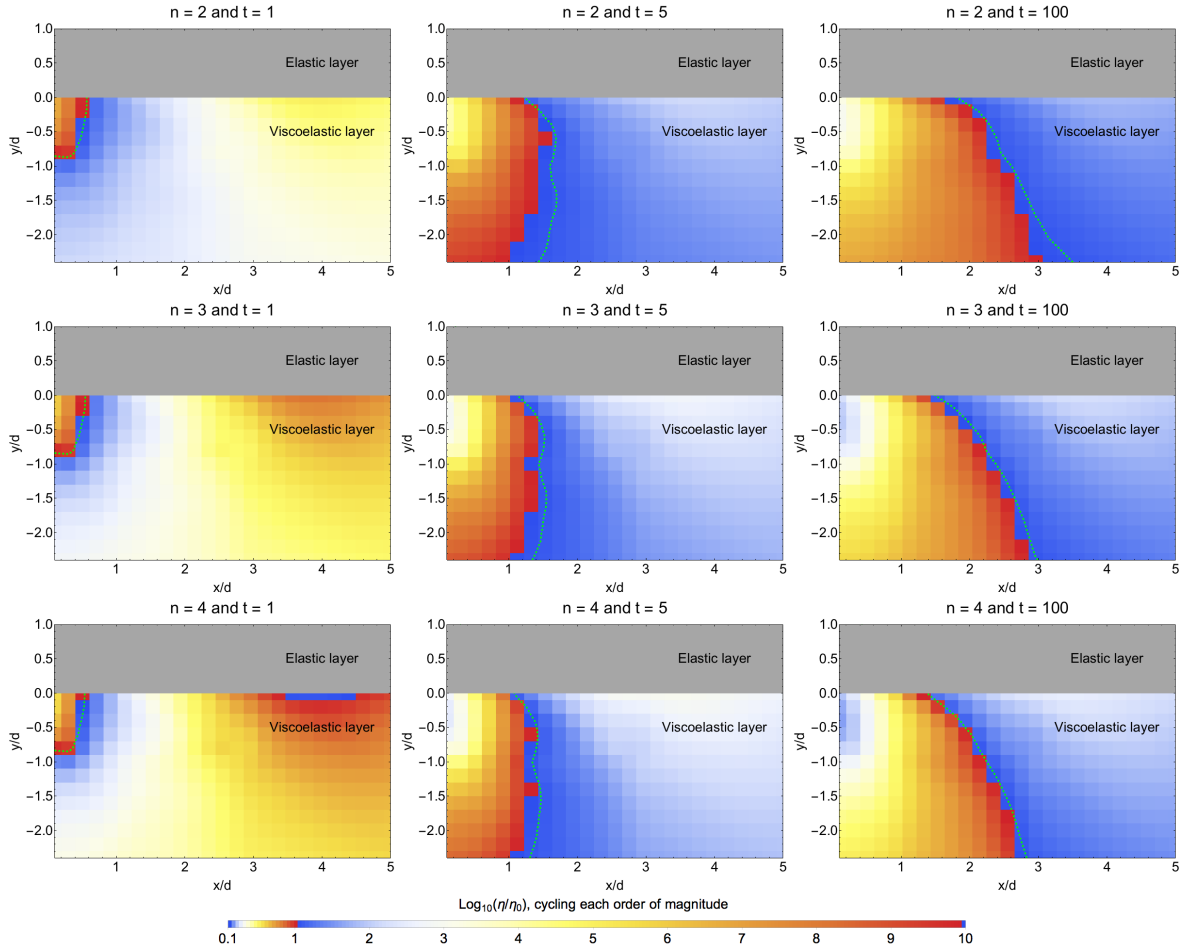


Figure 4.19: Viscosity structures for $n = 1, 2, 3, 4, 5$ where viscosity has been non-dimensionalised by equivalent linear viscosity η_0. Dashed green contour denotes $\eta = \eta_0$ and the elastic lid has been shaded gray.

4.7 Further work: Initial viscosity structure for a combined power law and linear rheology

I present here preliminary results on the scaling of viscosities for combined linear and non-linear creep mechanisms, such as diffusion and dislocation creep. The calculations are made at the instantaneous time $t = 0$ following a coseismic event, before any viscous relaxation of the stresses has taken place. I begin by assuming that the two creep mechanisms are governed by the same stress, so the total strain rate is given by the sum of the linear ($\dot{\epsilon}_l$) and non-linear ($\dot{\epsilon}_n$) mechanisms,

$$\dot{\epsilon} = \dot{\epsilon}_n + \dot{\epsilon}_l. \tag{4.13}$$

Each of these strain rates has an associated viscosity, such that an overall effective viscosity may be constructed as

$$\eta = \left(\frac{1}{\eta_n} + \frac{1}{\eta_l} \right)^{-1}. \tag{4.14}$$

The linear contribution in this example is treated as constant, such that $\eta_l = \eta_0$. The non-linear rheology has an effective viscosity given in terms of the deviatoric stress tensor

$$\eta_n = A(\sigma'_{ij}\sigma'_{ij})^{\frac{1-n}{2}}, \tag{4.15}$$

where deviatoric stress is dependent upon the elastic strain as

$$\sigma'_{ij} = 2\mu\epsilon^e_{ij}. \tag{4.16}$$

Combining equations (4.16) and (4.15) gives us the effective non-linear viscosity

$$\eta_n = A(8\mu^2\epsilon^e_{ij}\epsilon^e_{ij})^{\frac{1-n}{2}}. \tag{4.17}$$

The overall effective viscosity then comes from equations (4.17) and (4.14)

$$\eta = \left(\frac{1}{A} (8\mu^2 \epsilon^e_{ij} \epsilon^e_{ij})^{\frac{n-1}{2}} + \frac{1}{\eta_0} \right)^{-1}. \tag{4.18}$$

This may be rearranged and non-dimensionalised by η_0 such that $\eta' = \eta/\eta_0$ to give

$$\eta' = \left(B(4\pi^2 \epsilon^e_{ij} \epsilon^e_{ij})^{\frac{n-1}{2}} + 1 \right)^{-1}, \tag{4.19}$$

with

$$B = \frac{\eta_0}{A\pi^2} (2\mu^2)^{\frac{n-1}{2}}, \tag{4.20}$$

where B is defined here to simplify the elastic solution as the derivation proceeds. In order to calculate the effective viscosity following a coseismic rupture on a strike slip fault, I assume anti-plane conditions with slip on an infinite fault, thus $\epsilon_{xx} = \epsilon_{yy} = \epsilon_{zz} = \epsilon_{yz} = 0$. Using the elastic strains from (4.9) I may calculate $\epsilon^e_{ij}\epsilon^e_{ij}$,

$$\epsilon^e_{ij} \epsilon^e_{ij} = \frac{d^2 s^2}{4\pi^2 \left((d-z)^2 + y^2\right) \left((d+z)^2 + y^2\right)}. \tag{4.21}$$

The final effective viscosity is therefore given by (4.19) and (4.21),

$$\eta' = \left(B \left[\frac{d^2 s^2}{\left((d-z)^2 + y^2\right) \left((d+z)^2 + y^2\right)} \right]^{\frac{n-1}{2}} + 1 \right)^{-1}. \tag{4.22}$$

4.8 Comparison to results from *Yamasaki et al.* (2014)

The model presented in Section 4.7 can be compared to existing work by *Yamasaki et al.* (2014), who fit GPS data acquired before and after the 1999 Izmit and Duzce earthquakes. Their best-fitting model is that of an elastic lid over a viscoelastic layer with a weak zone of lower viscosity surrounding the base of the fault, parameters given in Table 4.1. In their study, two kinds of weak zones were modelled; blockwise constant

Length scale (L_0)	40km
Elastic lid (h)	$0.3L_0$
Fault depth (d)	$0.4L_0$
Model width (w)	$10L_0$
Model depth (L_0)	L_0
Weak zone width (Ω)	$0.6L_0$
Weak zone thickness (Θ)	$0.3L_0$
Fault slip (s)	(4.5 ± 1.5)m

Table 4.1: Best fit parameters from *Yamasaki et al.* (2014)

and a Gaussian-type structure. In both cases, a reduction of two orders of magnitude in the weak zone compared to the surrounding region were required to fit the data, illustrated in Figure 4.20.

To directly compare the effective viscosity structure from equation (4.22) to the results of *Yamasaki et al.* (2014), I compute the average viscosity within the weak zone (wz) and the stronger zone (sz) outside it using the integrals

$$I_{box} = \int_{y=-\frac{w}{2}}^{\frac{w}{2}} \int_{z=h}^{L_0} \eta' \, dy \, dz, \tag{4.23a}$$

$$I_{wz} = \int_{y=-\frac{\Omega}{2}}^{\frac{\Omega}{2}} \int_{z=h}^{h+\Theta} \eta' \, dy \, dz, \tag{4.23b}$$

and the areas

$$A_{wz} = 0.18L_0^2, \tag{4.24a}$$

$$A_{sz} = 6.82L_0^2, \tag{4.24b}$$

to give average viscosities for the two zones

$$\eta_{wz} = I_{wz}/A_{wz}, \tag{4.25a}$$

$$\eta_{sz} = (I_{box} - I_{wz})/A_{sz}. \tag{4.25b}$$

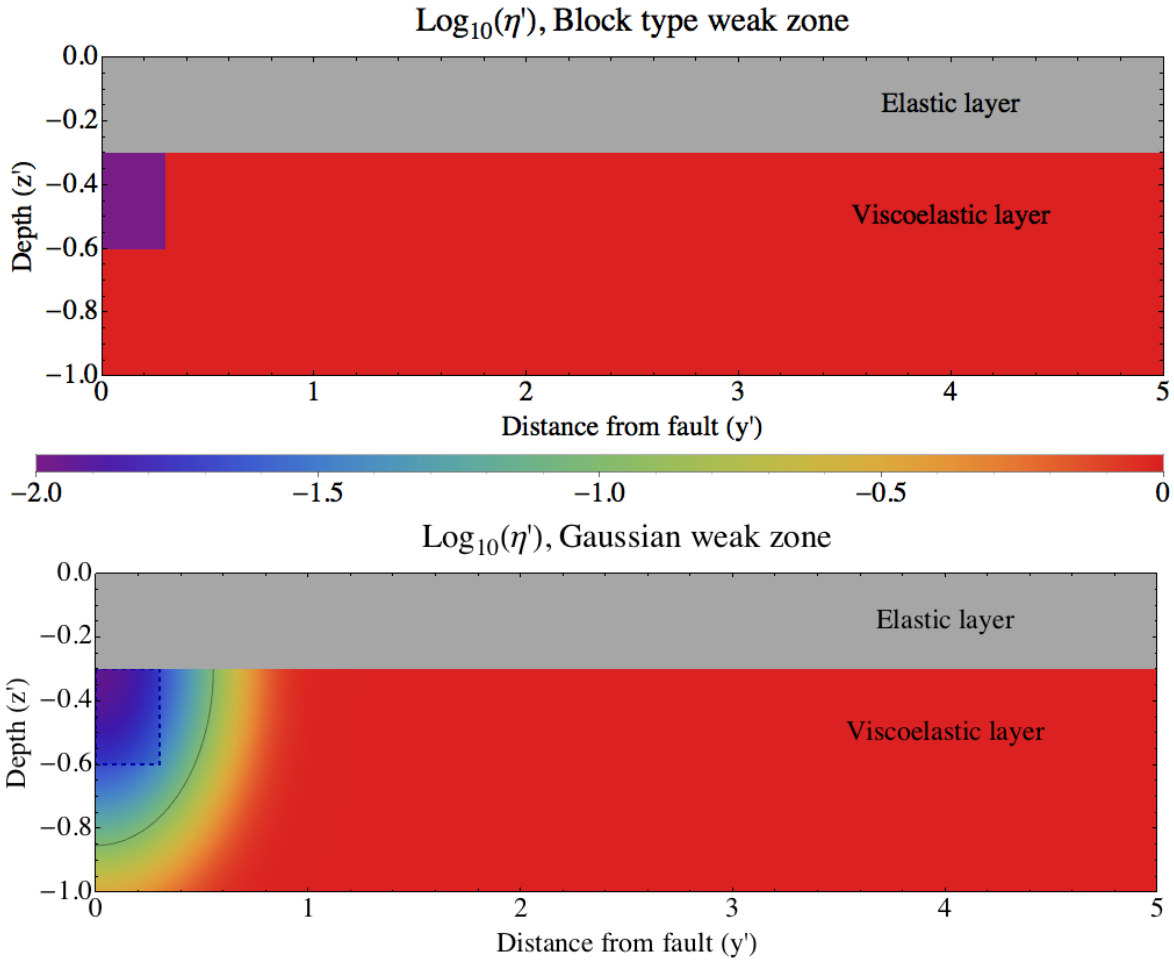


Figure 4.20: Viscosity variation in the viscoelastic layer, for the best fitting models in *Yamasaki et al.* (2014)

The majority of the parameters in equation (4.22) are specified in Table 4.1, such that I can then calculate the weakening factor $f_w = \log_{10}(\eta_{wz}/\eta_{sz})$ as a function of n and B (Figure 4.21). The results of *Yamasaki et al.* (2014) require a weakening factor of at least -2; equivalent scaling coefficients for integer values of $n \in [3, 10]$ are given in Table 4.2. Essentially the viscosity for the strong zone was a lower bound, and the governing factor for the post-seismic displacements was the relaxation time of the weak zone. The effective viscosity in the viscoelastic layer for a subset of the parameters in Table 4.2 are illustrated in Figures 4.22 and 4.23.

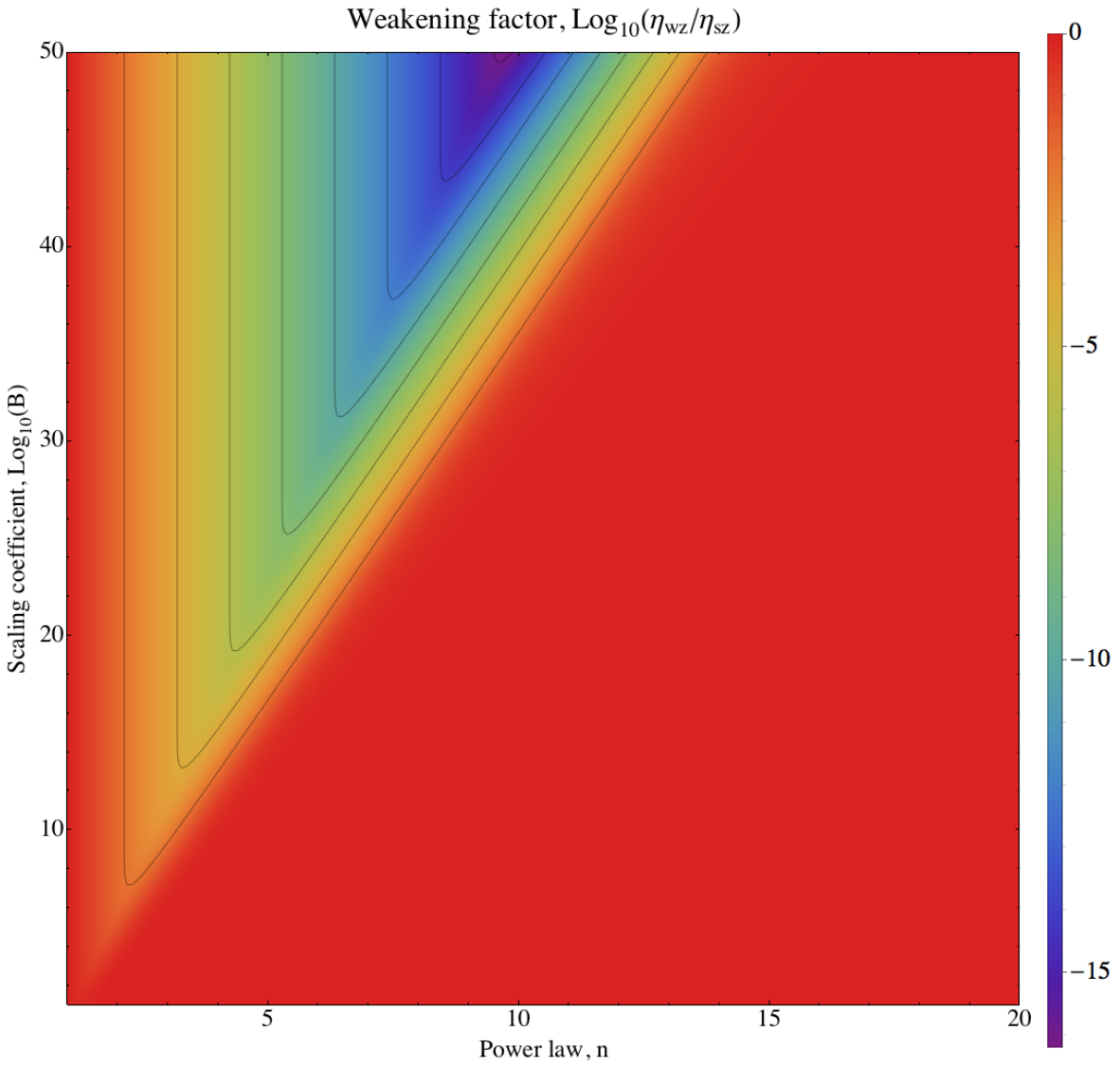


Figure 4.21: Post-seismic weakening factor, with contours at intervals of -2.

Power law (n)	Scaling coefficient ($\log_{10}(B)$)	Non-linear viscosity parameter ($\log_{10}(A)$)
3	9.5	10.99
4	13.1	10.07
5	16.8	9.50
6	20.55	9.12
7	24.3	8.84
8	28.1	8.63
9	31.9	8.46
10	35.75	8.31

Table 4.2: Parameter sets from combined linear and non-linear rheology to match results from *Yamasaki et al.* (2014). Non-linear viscosity parameter is calculated using equation (4.20) with $\mu = 3 \times 10^{10}$Pa and $\eta_0 = 2 \times 10^{20}$PaS

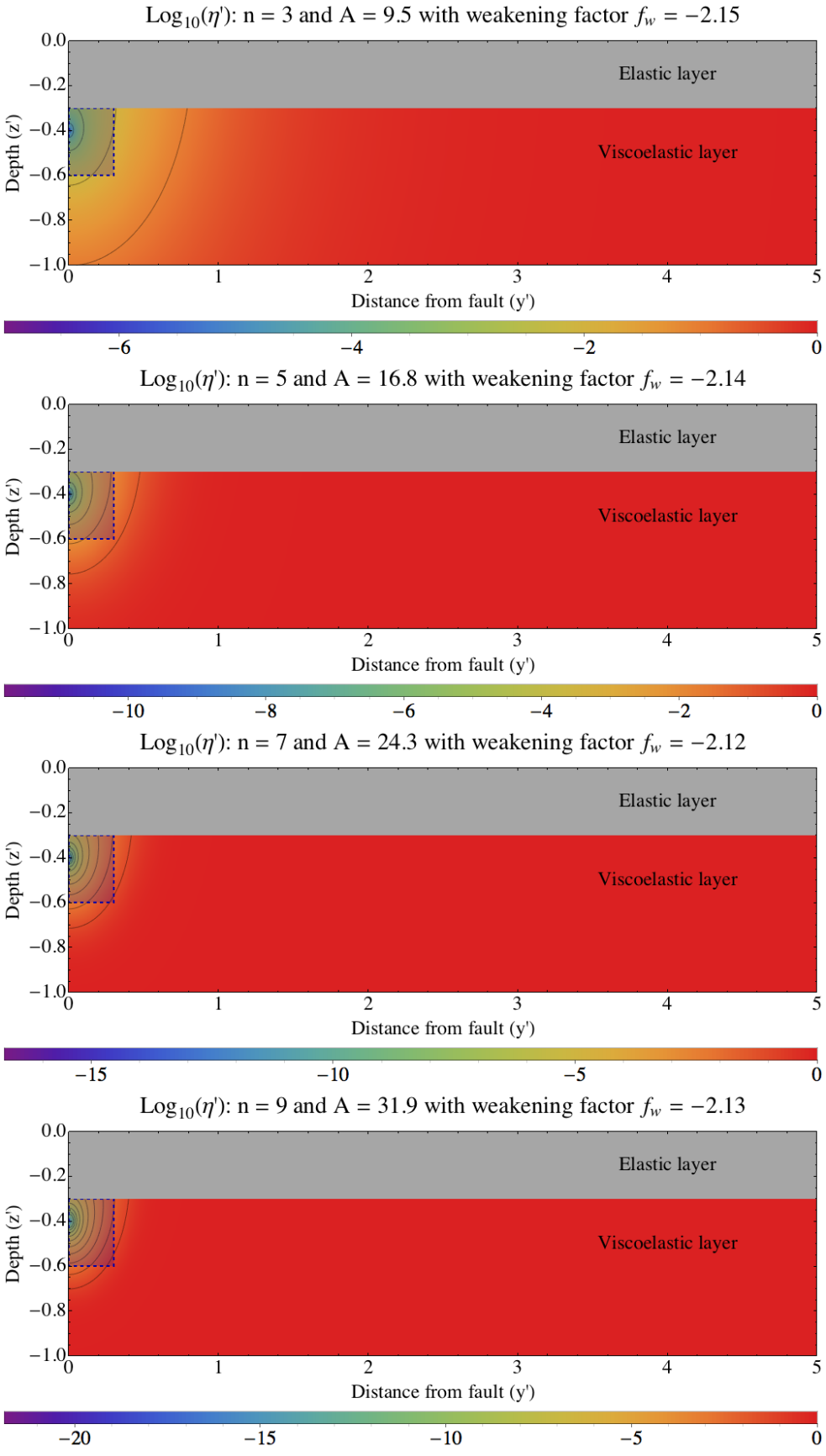


Figure 4.22: $log_{10}(\eta/\eta_0)$ in the viscoelastic layer for a combined rheology from equation 4.22.

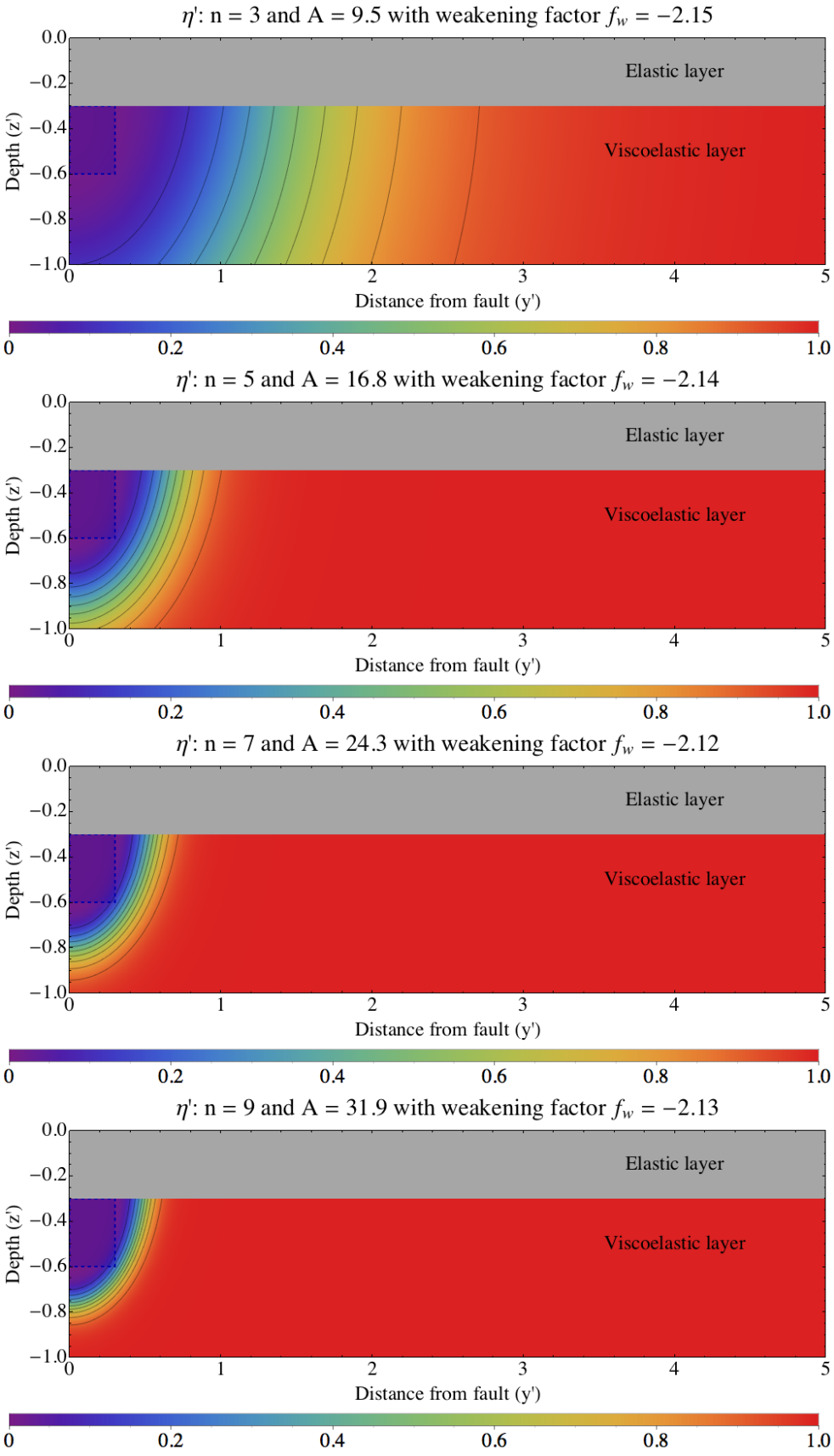


Figure 4.23: Viscosity variation in the viscoelastic layer from equation 4.22. Viscosity nondimensionalised by linear viscosity η_0.

4.9 Discussion

I have investigated the effect of two kinds of boundary conditions, free-slip and driving velocities, on post-seismic motions for both single and repeated ruptures. The calculations were performed using the 3D finite element code *oregano_ve* with periodic boundary conditions (implemented in Section 2.1) in the along-strike direction to simulate anti-plane conditions. Ruptures broke the elastic lid (thickness d) and drove displacements in a viscoelastic layer (thickness $20d$) with a non-linear rheology. For the case of a single rupture with free-slip boundary conditions, the post-seismic displacements are half that of the linear case when $n = 2$, and considerably lower for higher values of n. This is due to the very low strain-rates across the domain generating viscosities up to several orders of magnitude higher than the equivalent linear viscosity and suppressing deformation.

A single rupture with driving boundary conditions produces a very different result. Initially the viscosities are high and deformation is focused in the region of the base of the fault, but after around 4 Maxwell times the viscosity for the majority of the region has dropped to around $\eta \approx \eta_0$, except in a lobe directly beneath the fault where $\eta < \eta_0$. This reduction in viscosity allows a greater maximum post-seismic displacement for greater n, up to 20% larger for $n = 5$ compared to $n = 1$, once the system has fully relaxed, and the majority of this displacement is concentrated in the viscoelastic layer in the region close to the fault.

Multiple ruptures, or an earthquake cycle model, with free-slip boundary conditions show deformation patterns that vary little between $n = 1$ and $n = 5$ after 100 cycles. The sole exception to this is a slight linear ramp in the horizontal direction, which is more pronounced for greater n due to the higher viscosities in the far field suppressing relaxation. A zone of distributed shear develops beneath the fault, which exhibits little variation in width with n in contradiction to the results of Chapter 3. However, I would expect tectonic stresses to drive far field loading, which appears to

be a requirement for the localisation of shear. After the first few cycles, the viscosity structure beneath the fault develops a weak zone in the region surrounding the base of the fault. Again, the structure of this varies little with n, though higher values of n show elevated viscosities in the far field, contributing to the linear ramp that is visible in the displacements.

The fourth system I modelled numerically is that of multiple ruptures with far field loading conditions. These calculations have several properties that agree with the results of Chapter 3, including block motion in the elastic lid, narrowing of shear zone with increasing n, and the formation of a vertical zone of weakness aligned with the down-dip extension of the fault. The far field loading conditions produce a background viscosity that is relatively close to η_0, whilst the repeated ruptures ensure that a zone of weakness persists throughout the earthquake cycle.

An alternative model that may be capable of producing a relatively consistent background viscosity alongside localised reductions in viscosity would be a model combining two creep mechanisms, such as diffusion (a linear process) and dislocation (a non-linear process). I have made some initial investigations into how this model would behave in section 4.7, where I calculate the viscosity structure that would be present at $t = 0$, and then compare that to results from *Yamasaki et al.* (2014) in section 4.8. Qualitatively, these combined mechanisms could explain the transient zones of weakness in the immediate post-seismic displacements. Further work is required to implement modifications to *oregano_ve* implementing dual rheologies would allow us to quantify this phenomena.

Chapter 5

Viscoelastic modelling of subsidence and unrest at Santorini Volcano, Greece

This chapter represents work contributing to two articles, *Parks et al.* (2014) and *Watts et al.* (2015)[1].

5.1 Introduction

Santorini archipelago, situated in the South Aegean sea, consists of five islands which together comprise a volcanic caldera (Figure 5.1). For the past 360,000 years, the volcano has experienced major explosive eruptions every 10,000 to 30,000 years, separated by periods during which andesitic shields and dacite lava domes are built by multiple smaller effusive eruptions (*Druitt and Mellors* (1989)). The islands comprising the outer rim of the caldera (Thera, Therassia and Aspronisi) are the remnants of an older island, broken up during caldera-forming eruptions, the last of which occurred at ca. 1620 BC (the Minoan eruption). Since then, Santorini has been in a dome-forming

[1]InSAR processing and analysis conducted by M. Parks under the supervision of J. Biggs. GPS processing was undertaken by X. Papanikolaou and masked bathymetric difference maps produced by T. Watts. Lava-contraction and viscoelastic models were derived by J. Moore with guidance from P. England and T. Watts

phase, resulting in the growth of the two intra-caldera islands (Nea Kameni and Palea Kameni). The last major eruptions on the Kameni islands took place from 1866-1870 with minor eruptions from 1939-1941 and 1950.

Between 1950 and 2010 the volcano remained quiet, with no documented reports of seismic activity. Although reports of minor inflation were documented in the northern part of the caldera between 1994 and 2000 (*Stiros et al.* (2010); *Saltogianni and Stiros* (2012)) other studies (*Papageorgiou and Tzanis* (2010); *Foumelis et al.* (2013)) have argued that geodetic observations are not consistent with inflation of a shallow magma chamber. However, multiple measurements of surface deformation between 1992 and 2010 have confirmed a slow subsidence signal in the southwest region of Nea Kameni. The location and velocity of this signal has remained relatively stable throughout this period, with an observed subsidence rate of the order of 5 to 6 mm/yr (*Lagios and Parcharidis* (2005); *Papageorgiou and Tzanis* (2010); *Foumelis et al.* (2013)).

In January 2011, the volcano entered a period of unrest, characterised by the onset of detectable seismicity and caldera-wide uplift (*Newman et al.* (2012); *Parks et al.* (2012)). Figure 5.1 shows the location of volcano-tectonic (VT) earthquakes with local magnitude (M_L) greater than 2.0 which occurred within the caldera between October 2010 and September 2012 (Aristotle University of Thessaloniki, 2005-). The earthquakes are clustered close to the Kameni line, a NE-SW trending normal fault/fracture zone, which is thought to control vent locations for historic dome-forming eruptions on Nea Kameni (*Pyle and Elliott* (2006)). The unrest that began in late December 2010/January 2011 marked the first significant activity detected at Santorini since the end of the last eruption in 1950.

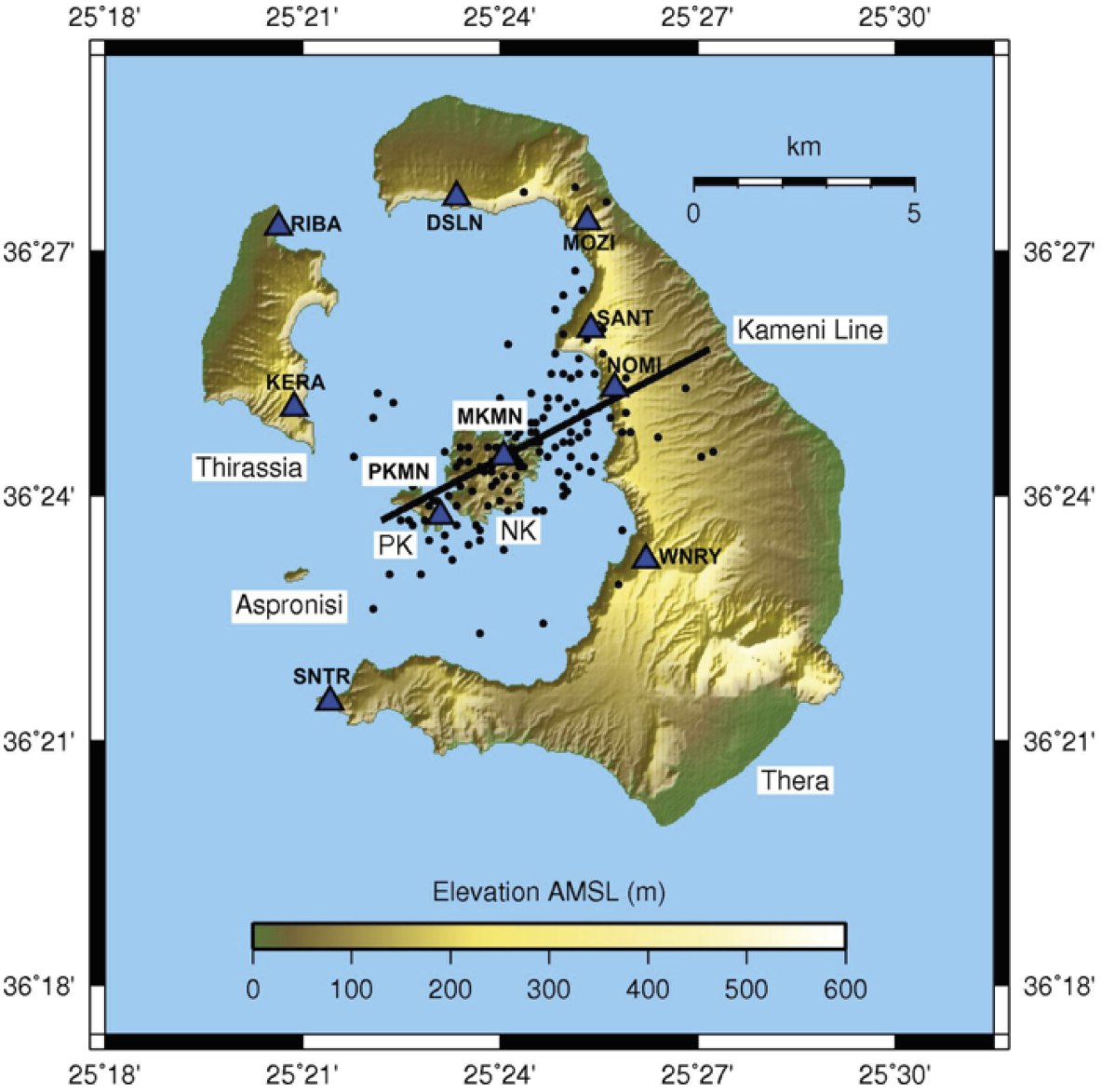


Figure 5.1: Map of Santorini, showing the location of recent seismic activity and continuous GPS sites. The black dots show epicentres of earthquakes greater than local magnitude 2.0 occurring from October 2010 to September 2012. Earthquakes were located by the Aristotle University of Thessaloniki (Aristotle University of Thessaloniki, 2005-). NK and PK label Nea Kameni and Palea Kameni respectively. cGPS sites are presented by the blue triangles. SNTR, WNRY, MKMN and DSLN were established and are operated by the University of Oxford and National Technical University of Athens. Sites KERA, RIBA, NOMI, PKMN and MOZI were established and are maintained by the Georgia Institute of Technology (*Newman et al.* (2012)). SANT is operated by the National and Kapodistrian University of Athens. Figure reproduced from *Parks et al.* (2014).

5.2 Slow subsidence from 1993-2010

InSAR acquisitions reveal a long-term subsidence trend in the south-west of Nea Kameni between 1993 and 2010, as shown in Figure 5.2. Scenarios that might explain the steady slow subsidence identified on Nea Kameni include: the cooling and contraction of an historic subsurface magma body, contraction of historic lava flows and loading by flows inducing viscoelastic relaxation of the substrate. Since there is no supporting evidence for a shallow deflating Mogi source beneath the southern portion of Nea Kameni (*Foumelis et al.* (2013)), while there is strong evidence for thermal contraction of and continued loading from historic lava flows (discussed in Section 5.3), we limit the modelling and discussion to the latter two scenarios and explore whether they can plausibly account for the observed signal.

5.2.1 Thermal contraction

The post-emplacement contraction of lava flows may result in observable surface deformation via two primary processes: (1) Repacking of clasts and compression of voids within underlying flows or strata and (2) Thermal contraction of lava flows. The loss of void space in underlying lava flows/strata is thought to occur rapidly following emplacement (*Stevens and Wadge* (2001)) so we do not consider (1) a likely on-going phenomenon for flows emplaced over 65 years ago, like those at Santorini. Thus, we address only (2) below.

The slow subsidence signal observed at Nea Kameni corresponds spatially to the locations of the 1866-1870 and 1939-1941 dacite lava flows (Figure 5.3a). First, I explore whether the observed subsidence on Nea Kameni between 1993 and 2010 may be related to the cooling and contraction of this lava pile by modelling each flow as a slab of thickness d, with temperature T (relative to ambient) and coefficient of thermal expansion α. If the individual flows are emplaced at time t before present, on a semi-infinite medium of thermal diffusivity κ (*Carslaw and Jaeger* (1959)), then the rate of

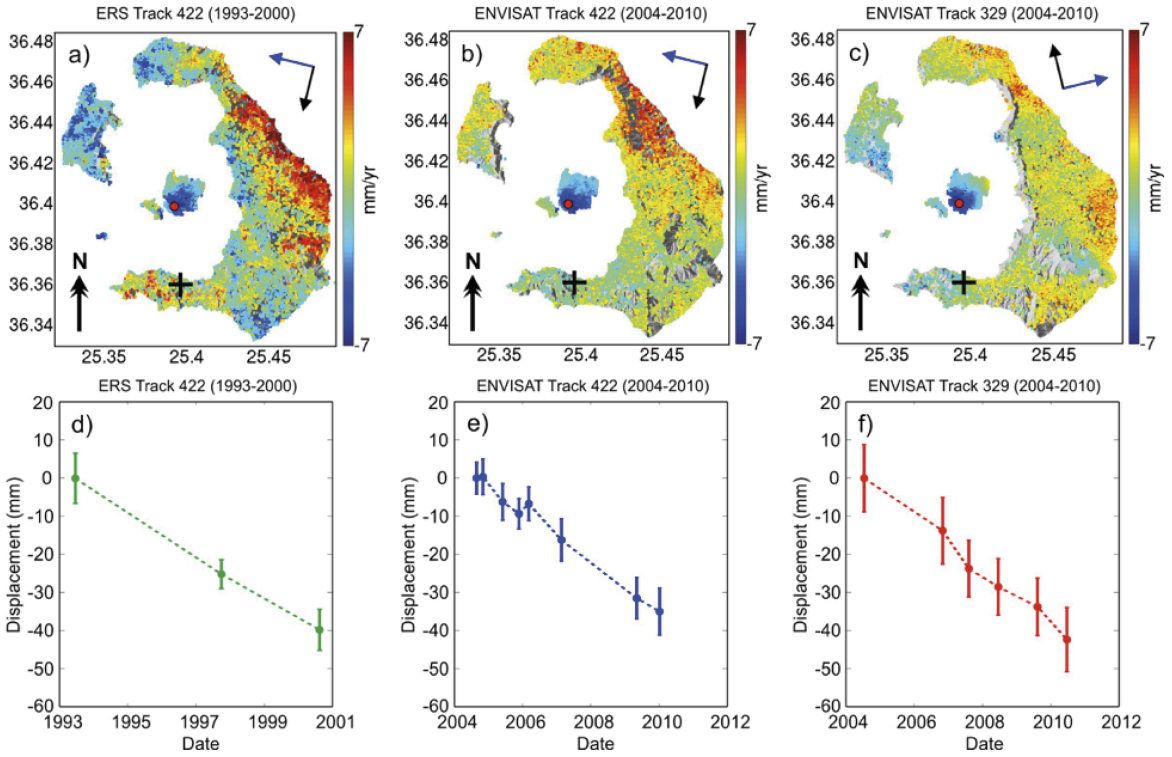


Figure 5.2: (a)-(c) Mean line-of-sight (LOS) velocities on Santorini from June 1993 to June 2010, derived using the StaMPS small baseline (SB) method. (a) Descending orbit data from ERS track 422 between June 1993 and August 2000. (b) Descending orbit data from ENVISAT track 422 between August 2004 and January 2010. (c) Ascending orbit data from ENVISAT track 329 between July 2004 and June 2010. (d-f) Time series of LOS displacements generated using SB interferograms from (a)-(c) respectively. The data point used to generate the time series is displayed as the red filled circle in figures (a-c). The dashed coloured lines in figures (d-f) display the average displacements within a 100 m radius of the red filled circle. The error bars represent the variation in displacements within the 100 m radius. The ERS and Envisat data were processed at 4 looks in range, 20 looks in azimuth and a 15-m DEM was used to remove topographic fringes. The look direction of the satellite is displayed by the blue arrow in figures (a)-(c). LOS velocity and displacement is negative away from the satellite for all interferograms shown. Figure reproduced from *Parks et al.* (2014).

contraction ω is

$$\omega = \frac{\alpha\kappa T(e^{-d^2/4\kappa t} - 1)}{\sqrt{\pi\kappa t}}. \tag{5.1}$$

Complex numerical models have been applied at other volcanoes to determine rates of lava flow cooling resulting from multiple thermodynamic processes (*Patrick* (2004)). However, models incorporating transient, short-term, post eruptive processes are only

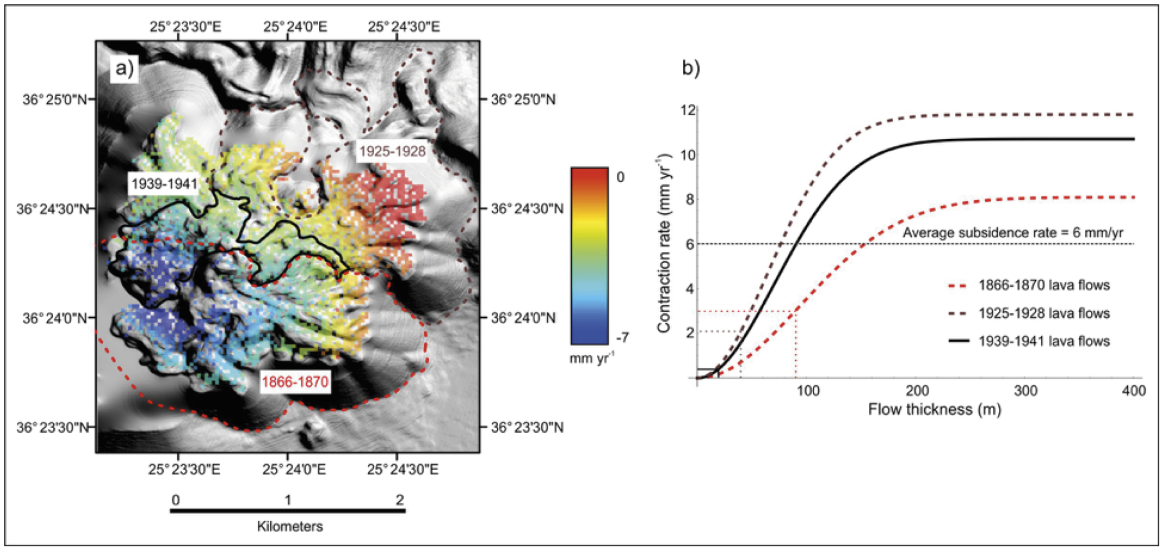


Figure 5.3: (a) Merged bathymetry and LiDAR grid after Nomikou et al. (2014), overlain with mean LOS displacements (mm/yr) computed using ascending ENVISAT interferograms from track 329. LOS displacement is negative away from the satellite. The red dashed polygon represents the outline of the 1866-1870 lava flow. The brown dashed polygon represents the outline of the 1925-1928 lava flow and the black polygon represents the outline of the 1939-1941 lava flow. The dashed lines represent the extrapolated boundaries of the 1866-1870 and 1925-1928 lava flows. The north-west part of the 1866-1870 flow is thought to reside beneath the 1939-1941 lava flows. (b) Modelled thermal contraction rate of historic lava flows (mm/yr), plotted as a function of lava flow thickness.

appropriate within the first two years of cooling, after which time thermal conductivity and porosity become the dominant factors governing heat loss (*Patrick* (2004)). Based on these observations and considering the extended period since lava extrusion, I do not see the need for a more complex model than the thermal contraction model presented here.

I computed the expected contraction rate as a function of flow thickness (Figure 5.3b) for the 1866-1870, 1925-1928 and 1939-1941 lava flows, using equation (5.1) and input variables: $\alpha = 3 \times 10^{-5}$ K^{-1}, $\kappa = 10^{-6}$ m^{2}s^{-1} and $T = 1223$ K (values from

Henderson and Henderson (2009)). The time since emplacement (t) is $\sim$ 140 years for the 1866-1870 flow, $\sim$ 80 years for the 1925-1928 flow and $\sim$ 65 years for the 1939-1941 flow. The difference in contraction rate between the three curves (displayed in Figure 5.3b) results from the different emplacement times. The average thickness of the 1886-1870 flow is approximately 90 m, which corresponds to $\sim$ 3 mm/yr of contraction. The average thickness of the 1939-1941 flow is $\sim$ 20 m (*Nomikou et al.* (2014)), which results in $<$ 1 mm/yr contraction; while the 40 m thick 1925-1928 flow (*Nomikou et al.* (2014)) would correspond to $<$ 2 mm/yr of contraction (Figure 5.3b). The 1866-1870 flow still has the largest continuing contraction rate due to its substantial thickness; this alone could account for $\sim$ 50% of the observed subsidence for the period 1993-2010. Although the combined rates from the three lava flows modelled (1866-1870; 1925-1928; 1939-1941) would add up to the 6 mm/yr of observed subsidence, the 2 mm/yr from the 1925-1928 flow cannot contribute to the subsidence observed in the southern portion of the island, as this flow was only extruded in the northwest of Nea Kameni (Figure 5.3a).

Since the observed subsidence exceeds that which may be due to thermal contraction alone, unless the thicknesses were double than current best estimates (*Nomikou et al.* (2014)), I now investigate other processes that may contribute to the signal. Subsidence on active volcanoes has previously been interpreted as loading from historic flows and induced viscoelastic relaxation of the substrate (*Ebmeier et al.* (2012); *Ofeigsson et al.* (2011); *Fournier et al.* (2010)), so I shall investigate this in Section 5.2.2.

5.2.2 Loading from historic lava flows and viscoelastic relaxation of the substrate

Since the 1866-1870 flow was the most voluminous of the historic lava flows extruded on Nea Kameni (*Pyle and Elliott* (2006); *Nomikou et al.* (2014)), I consider that loading from this flow and viscoelastic relaxation of the substrate may account for at least part of the observed subsidence signal. To test this, I modelled a cylindrical load placed on an elastic lid of thickness h_c, on a viscoelastic half-space (*Nakiboglu and Lambeck*

(1982); *England et al.* (2013)), where the displacement at radius r, $w(r)$, is given by

$$w(r) = Ah_0B_1 \int_0^\infty \frac{J_1\left(\frac{Au}{h_c}\right) J_0\left(\frac{ru}{h_c}\right) \left(1 - \frac{B_2 u e^{-\frac{tF_1}{\tau_{max}}}}{B_2 u + B_1 h_c + u^4}\right)}{B_1 h_c + u^4} du, \tag{5.2a}$$

$$F_1 = \frac{(B_1 h_c + u^4)}{(B_2 u + B_1 h_c + u^4)}, \tag{5.2b}$$

$$B_1 = \frac{6(1-\nu)\rho_c g}{\mu}, \tag{5.2c}$$

$$B_2 = 12(1-\nu). \tag{5.2d}$$

Other parameters may be found in Table 5.1. Based on the merged bathymetry and

A	Radius of load
h_0	Thickness of load
J_n	Bessel function of order n
τ_{max}	Relaxation timescale of viscoelastic half-space
h_c	Elastic lid thickness
ν	Poissons ratio
ρ_c	Density of crust
g	Acceleration due to gravity
μ	Shear modulus of crust

Table 5.1: Parameters for cylindrical load model.

LiDAR grid the load corresponding to the 1866-1870 flow has an approximate radius of $A = 1$km and height of $h_0 = 100$m. The load has a density of $\rho_c = 2500$kg/m^3 and the elastic lid has a density of 2800kg/m^3 and a Poissons ratio of 0.25. We assume a shear modulus of 5GPa and that loading commenced $\sim$ 140 years ago. The model is run for a range of elastic lid thickness (T_e) and substrate viscosities (η), with the peak displacements (at $r = 0$) plotted in Figures 5.4 and 5.5. The model (Figure 5.4) suggests that $> 60\%$ of the observed rate of subsidence at Nea Kameni may be induced by the loading of a 100 m thick lava flow, assuming a elastic lid thickness between 2-3 km. Note that whilst my model is for a half-space, in practice I only require this thickness of elastic lid for less than 3 km around the volcano, almost certainly due to the elevated geothermal there. For example, a viscosity of $\sim 5 \times 10^{18}$ Pa s and an elastic lid thickness of 3000m would result in an observed displacement rate of $\sim$ 4 mm/yr

at the present day. Figure 5.5 emphasises the tradeoff between elastic lid thickness and viscosity; in general there are a pair of viscosities that would give subsidence rates commensurate with the observations for any given elastic lid thickness. Additional data, presented in Section 5.3, will help me to discriminate between these cases.

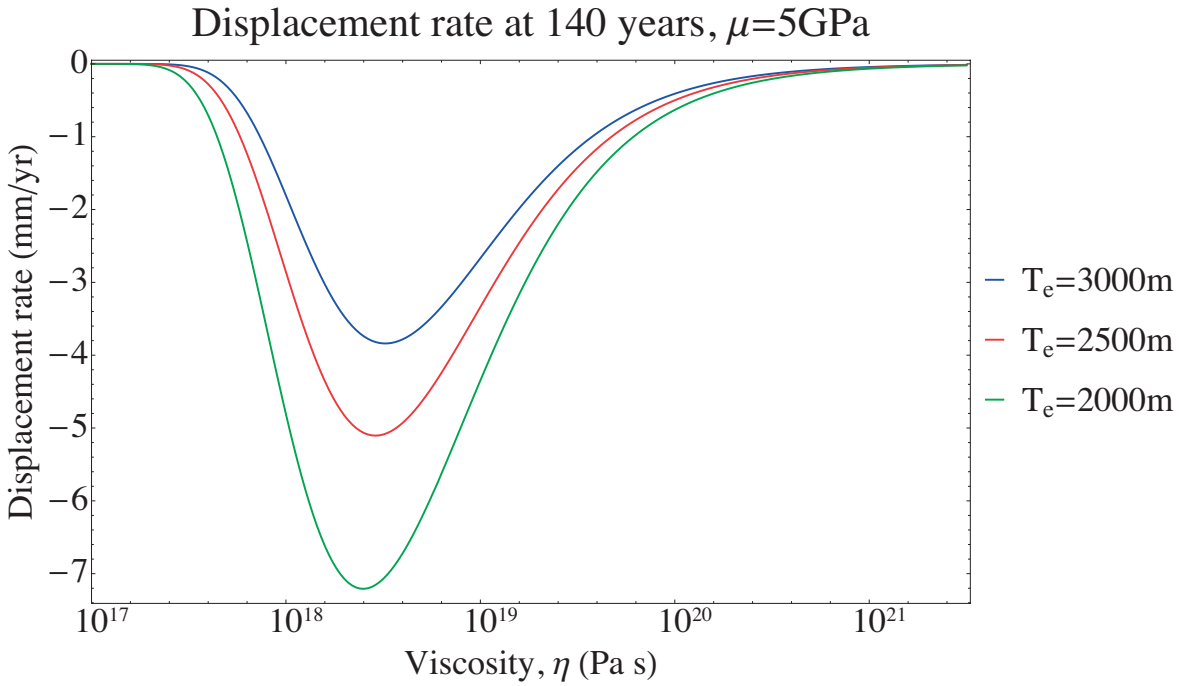


Figure 5.4: Viscoelastic model for a 100 m thick lava flow emplaced in 1866, using a shear modulus of 5 GPa. The different coloured contours represent the model using different values for the thickness of the elastic lid (T_e). For example a viscosity of 5×10^{18} Pa s and an elastic lid thickness of 3000 m would result in an observed displacement rate of ~ 4 mm/yr.

Given that the location of the subsidence signal agrees closely with the extent of the 1866-1870 lava flow, and significant seafloor subsidence occurred in the area of the extruded flows following emplacement (*Schmidt* (1868)), I suggest that a combination of thermal contraction and load-induced relaxation is sufficient to account for the observed subsidence. Similar mechanisms have been proposed by *Briole et al.* (1997) and *Stevens and Wadge* (2001) to explain post-eruptive deformation at Mt Etna (Italy) and by

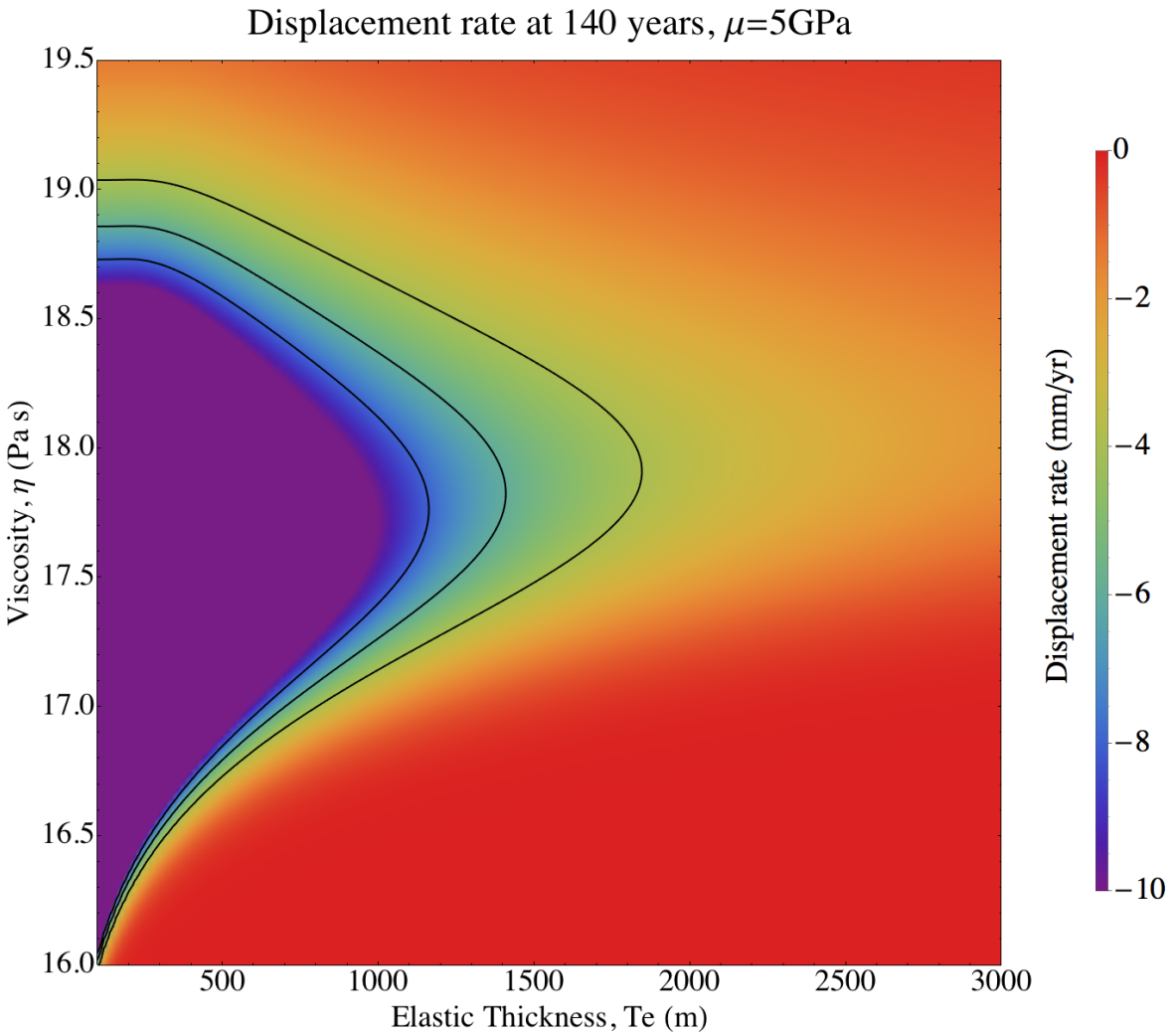


Figure 5.5: Viscoelastic model for a 100 m thick lava flow emplaced in 1866, using a shear modulus of 5 GPa. Displacement rate at 140 years as a function of elastic lid thickness T_e m and viscosity η Pa s.

Ofeigsson et al. (2011) at Hekla volcano (Iceland). It is important to note that the processes of cooling & contraction and loading are not mutually exclusive, and I would expect them to occur concurrently.

5.3 Historical bathymetric charts and the submarine evolution of Santorini

5.3.1 Introduction

During the late 1790s and early 1800s, the British Admiralty conducted a global campaign of seafloor depth measurements using disarmed war ships that had been converted to surveying ships. By the mid-1800s, more than 1900 individual charts of the coasts of western Europe, North and South America, West and East Indies, Australia and New Zealand, and the Pacific ocean islands had been published. Santorini was first surveyed in 1848 by Captain Graves R.N. (1802-1856) onboard HMS Volage, a 28 gun 6th rate ship of the line that was fitted out for survey work in late 1846 to early 1847. Unfortunately, the logs do not detail the sounding methods used, but there are a number of references to loss of leads and lines suggesting traditional depth methods. Positions are recorded in the logs and were determined mostly by bearings on prominent landmarks. The logs also record the occasional celestial measurement of latitude using sextants and dead-reckoning and ships chronometer measurements of longitude.

The charts based on Captain Graves's soundings are remarkably detailed. Figure 5.6 shows the 1848 Santorini map; this shows Nea Kameni when it comprised of two islands, Nea Kameni and Mikra Kameni, now joined by more recent flows. Soundings are recorded in fathoms (1f=1.82m). The map shows a narrow, shallow channel between Mikra Kameni and Nea Kameni and a wide, deep channel between Nea Kameni and Palea Kameni.

Figure 5.7 shows a bathymetric map based on the 1848 soundings, height and coastline data, reveals the main bathymetric features of the Santorini region in surprising detail. The North, South and West basins reach depths of up to 407m, 333m and 480m respectively, and the average difference in depth between the two basins is $\sim$ 80 m.



Figure 5.6: Royal Navy bathymetric map of Santorini produced by Captain Graves in 1848. Admiralty chart No. 2043.

The map also shows steep caldera walls (slopes up to 40°) and the elongate nature of the head of the northern entrance channel into the caldera. The morphology of the flanks of the volcano is not distinguished in the map, due to a lack of soundings close to the Kameni islands.

As the soundings from the 1848 bathymetry map do not lie on a regular grid, I have performed a series of statistical tests on the combined swath bathymetry data set in order to calculate the error introduced by gridding. To calculate this error, I took the distribution of spacing between soundings in the 1848 data set and used that distri-

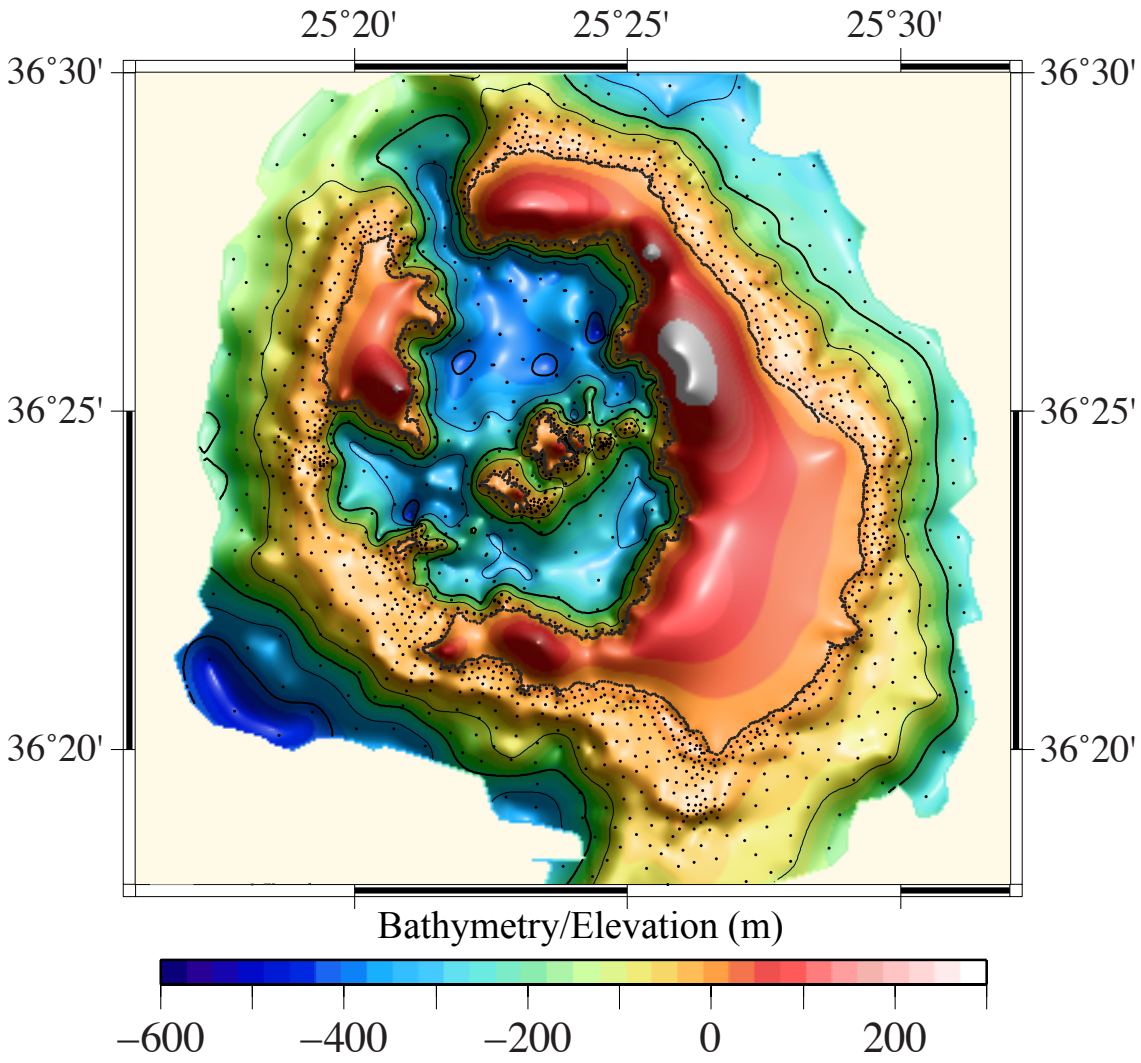


Figure 5.7: Bathymetry and topography map based on the 1848 soundings, coastline and height data. The map was constructed by digitizing the 1848 soundings, coastline (which was assumed to be at zero depth and height) and heights and smoothing and gridding the combined data set using GMTs blockmean and surface algorithms with a mean grid interval of 70 × 70 m. Contour interval = 100 m. 200 m contours are indicated by a thick black line. Figure reproduced from *Watts et al.* (2015).

bution to generate a set of points with similar spacing, and randomized geographical locations (Figure 5.8). The modern swath bathymetry data were then sampled at these locations and a new grid constructed. I difference the swath bathymetry and new grid and then plot the residual (Figure 5.9). I find that the greatest errors are located where there are large changes in topography/bathymetry, such as at the edge of the caldera. To quantify the error on the gridding, I calculate the Mean Absolute Deviation (MAD)

across this region for each model. After 100 simulations, the MAD for the entire area lies between 13.6 and 19.2 m, with a mean of 15.5 m. For the Kameni Islands region, the mean decreases to 8.6 m and I believe this to be a representative error associated with the spatial distribution of the 1848 soundings data.

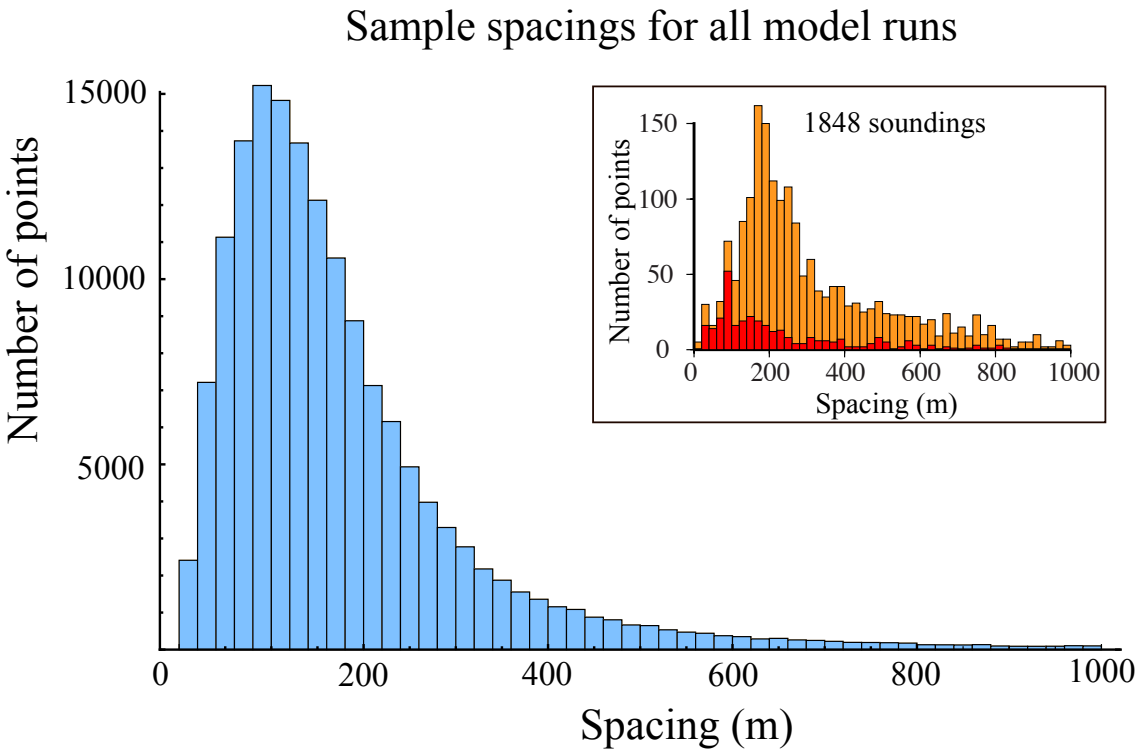


Figure 5.8: Plot of number of points and spacing used to generate the synthetic model which comprises a set of points with similar spacing distribution as the actual 1848 soundings and randomized geographical locations. Inset shows the number of points and spacing distribution of the actual 1848 soundings data. Brown-filled areas show the entire Santorini area. Red-filled areas show only the region of the Kameni Islands.

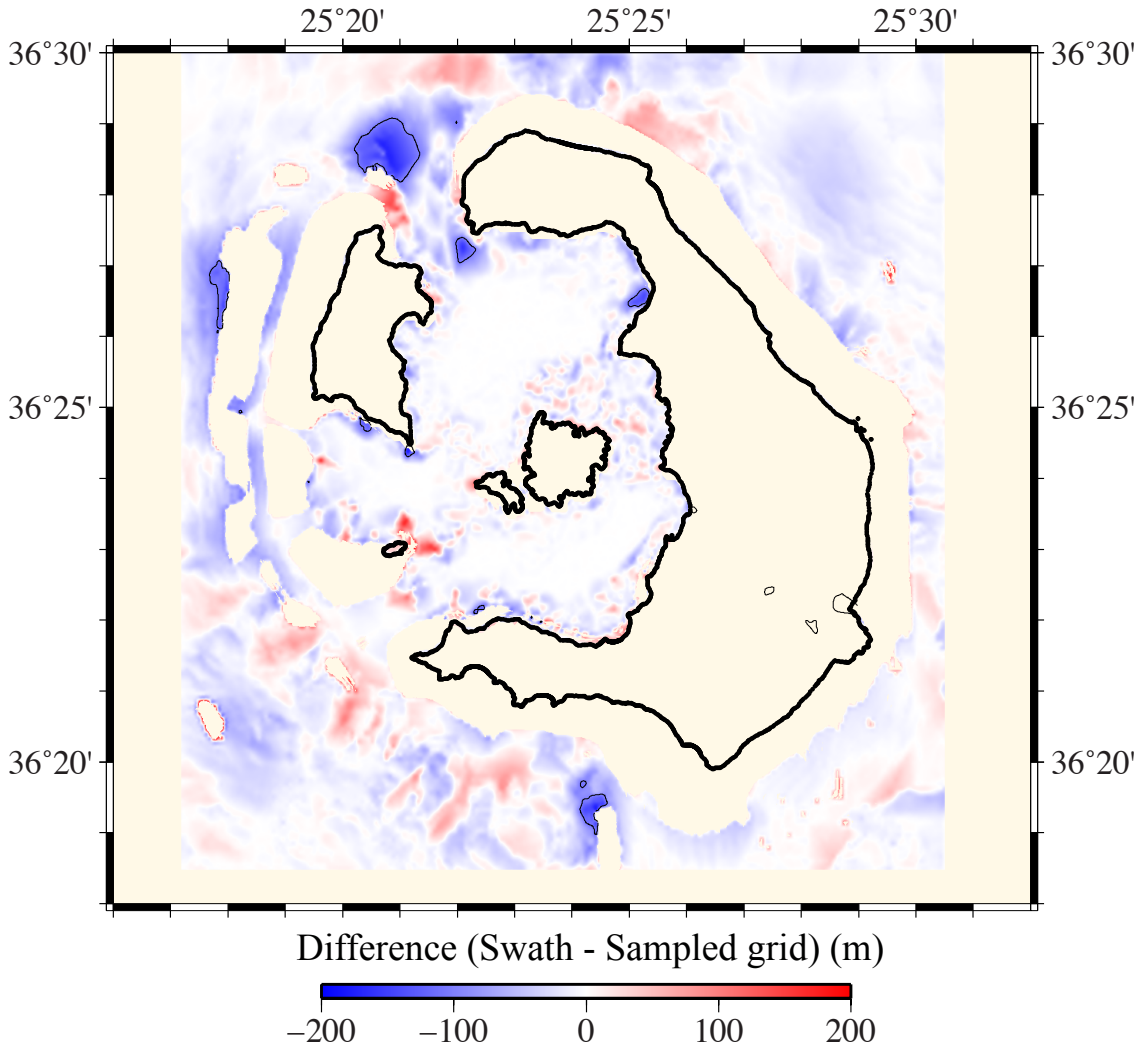


Figure 5.9: Map of the differences in seafloor depth between the observed swath bathymetry defined data set and the predicted depth based on the sample distribution shown in Figure A1 (blue-shaded area) and randomized geographical locations. The mean, standard deviation, and MAD of the difference data for this simulation is 25.0, 29.4, and 15.5 m respectively.

5.3.2 Collapse and the viscoelastic behaviour of the crust and mantle

Field-based observations (*Fouqué* (1879); *Leycester* (1850); *Brine* (1866); *Schmidt* (1868); *Washington* (1926)), together with the difference bathymetry maps in Figure 5.10(e, f and g), indicate that the Kameni Islands region is associated with distinct phases of shallowing and deepening of the seafloor since 1858. We have attributed

the shallowing to growth and the build-up of magmatic material (e.g. submarine lava flows) on the surface of the pre-existing seafloor. The origin of the deepening is not

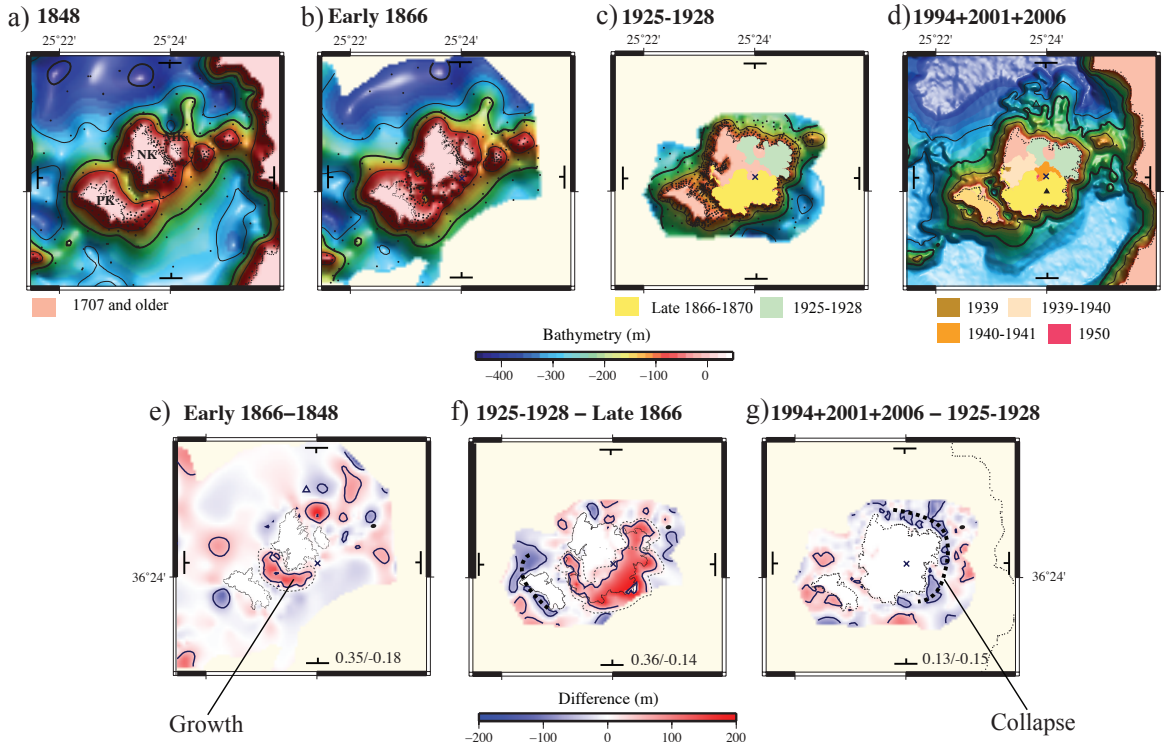


Figure 5.10: Map of the bathymetry, geology and differences in seafloor depth in the Kameni Islands region. a) Bathymetry in 1848. b) Bathymetry in early 1866. c) Bathymetry in 1925-1928 showing the extent of the 1866-1870 and 1925-1928 lavas. d) Modern bathymetry showing the extent of the 1939, 1939-1940, 1940-1941 and 1950 lavas. e) Differences in seafloor depths between 1848 and early 1866. Thin dotted lines show the mask used to isolate the volume of submarine lava associated with the early 1866 event. f) Differences in seafloor depths between late 1866 and 1925-1928. Thin dotted lines show the mask used to isolate the volume of submarine lava associated with the late 1866-1870 (southernmost polygon) and the 1925-1928 (northernmost polygon) events. Thick dashed lines show the minimum of the collapse region. g) Differences in seafloor depths between 1925-1928 and the modern bathymetry. Figure reproduced from *Watts et al.* (2015).

as clear; possible factors include large-scale slope failure, thermal cooling and magma deflation, or some combination of these. Large-scale slope failures are a major feature of oceanic islands where sliding on steep slopes generates debris flows with large (up to 1 km wide) blocks which, in turn, initiate turbidites that travel large distances across the seafloor (Hawaii - *Moore and Clague* (1989); Tenerife - *Watts and Masson* (1995)). Although submarine landslide deposits have been reported on the flanks of Santorini

east of Thera (*Bell et al.* (2013)), the difference bathymetry maps do not show the characteristic upslope head scar and deepening, or their associated downslope debris flow deposits and shallowing, typical of sector collapses on the stratovolcanic cones of other arc volcanoes (e.g. Monowai - *Watts et al.* (2012)). This is confirmed by the ROV dives carried out during 2010-2011 on board E/V Nautilus (*Nomikou et al.* (2012, 2013)), although there is evidence of some debris flow deposits and small, scattered, blocks all around the submarine edifice of Nea Kameni.

Post-eruption cooling and contraction of lava flows and deflation of a shallow magma chamber can both create subsidence. In Section 5.2.1 I showed that cooling and contraction since the 1866-1870 loading event on the Kameni Islands could explain up to 50% of the present day rate of subaerial subsidence of 6 mm/yr inferred from GPS/InSAR data, with the remainder due to viscoelastic relaxation, as modelled in Section 5.2.2. With the addition of the bathymetric data set, it seems unlikely that cooling and contraction alone could explain the magnitude of the observed submarine subsidence since 1866-1870, which totals up to 80 m, implying average subsidence rates of up to 550 mm/yr.

As discussed in Section 5.2.2, another possible explanation is some form of viscoelastic stress relaxation in the crust that underlies the Kameni Islands region. This phenomena arises in oceanic lithosphere because of its thermal cooling history and the strong dependence of viscosity on temperature (*Watts and Zhong* (2000)). During volcano loading, bending stresses in the hot lowermost low viscosity layer of the lithosphere migrate upwards into the cool, uppermost high viscosity layer. The thickness of the lithosphere that supports a volcanic load therefore changes with time from its initial short-term (seismic) thickness to its long-term (elastic) thickness. Evidence for this relaxation of load-induced bending stress is seen in the rapid subsidence of newly formed oceanic islands (*Watts and Zhong* (2000)) and the shift of the main depocentre towards the load which results in stratigraphic offlap in their flanking moats (*Collier*

and Watts (2001)).

In order to apply the viscoelastic model to the Kameni Islands, I have used the volcanic loads estimated from masking the difference maps in Figure 5.10(e, f and g) and emplaced them on a simple model of an elastic plate which overlies a viscoelastic half-space. I follow *Brotchie and Silvester* (1969); *Nakiboglu and Lambeck* (1982); *England et al.* (2013), who use a Green's function approach to solve the general equation for the flexure due to a concentrated load, equations (5.2), and calculate the subsidence due to the masked loads as a function of time and values of the thickness of an elastic plate and relaxation time of the underlying viscous substrate. I discretised the gridded load and calculated the viscoelastic response at the present day by evaluating the Green's functions using Mathematica 9.0 (*Wolfram Research* (2012)), Figure 5.11. I have been able to constrain the elastic lid thickness, T_e, and the viscosity of the substratum, η, that are supporting the load by comparing the predicted subsidence due to the 1866-1870 load to the observed total subsidence between 1866-1870 and the present day (blue shaded regions in Figs. 5.10e, f and g), together with the rates of subsidence inferred from InSAR data (Figure 5.2).

To calculate the best fitting parameters for T_e and η, I looked at the peak displacement after 140 years, and the displacement rate at that time. I required the solution to generate total subsidence of between 50 and 100m and have a modern rate of subsidence no greater than that inferred from the InSAR data, 6mm/yr. These two constraints proved sufficient to narrow down the range of parameters that fit the data, as illustrated in Figure 5.12.

Figure 5.13 shows in map form the subsidence and uplift expected at the present day for a volcanic load emplaced during 1866-1870 on an elastic plate ($T_e = 0.6$ km) that overlies a viscoelastic half-space ($\eta = 6.3 \pm 0.5 \times 10^{15}$ Pa s). These parameters best explain both the magnitude and wavelength of the bathymetric-derived subsidence

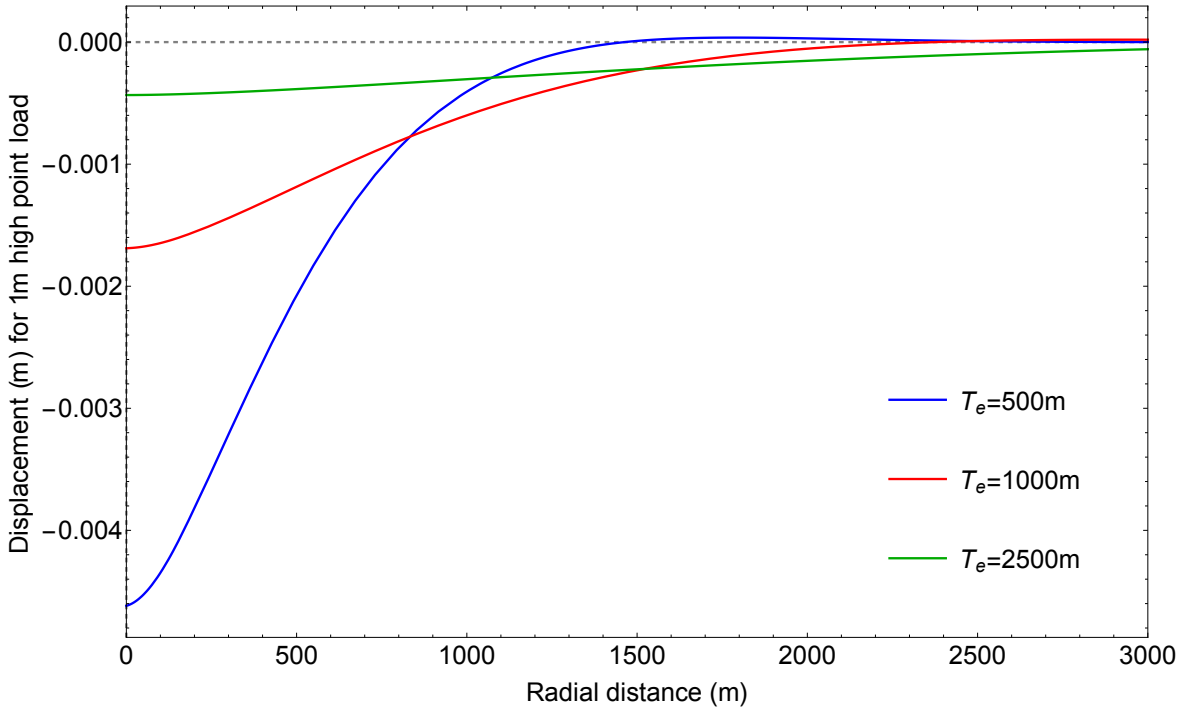


Figure 5.11: Viscoelastic Green's functions for displacement at 140 years of a point load 1m high as a function of radial distance from the load. Note that for $T_e = 500$m the solution exhibits a region of uplift around 1.5km from the source location.

and the GPS and InSAR data. In particular, they account for the subsidence observed from field observations of the southeast coast of Nea Kameni (*Fouqué* (1879)), the total subsidence of $\sim$ 80 m inferred from historical bathymetric charts offshore the southeast coast of Nea Kameni, Figure 5.10(f and g), the lack of evidence of any subsidence from the shoals (Figure 5.13), between Nea Kameni and Fira Port and the relatively slow rate of background subsidence in GPS and InSAR data of $\sim$ 6 mm/yr (Figure 5.3).

The best fitting parameters of T_e and η are reasonable given the setting of the Kameni Islands. The intra-caldera setting of an active arc suggests elevated temperatures, resulting in low elastic thickness and low viscosities of the order of values deduced at fast spreading mid-ocean ridge crests, where T_e and η values as low as 2.0 km (*Cochran* (1979)) have been reported. I note that the best T_e value of 0.6 km is actually less than the seismogenic thickness, T_s, which during 2011 and early 2012 was between 1 and 6 km along the Kameni line (*Newman et al.* (2012)). While both T_e and T_s are

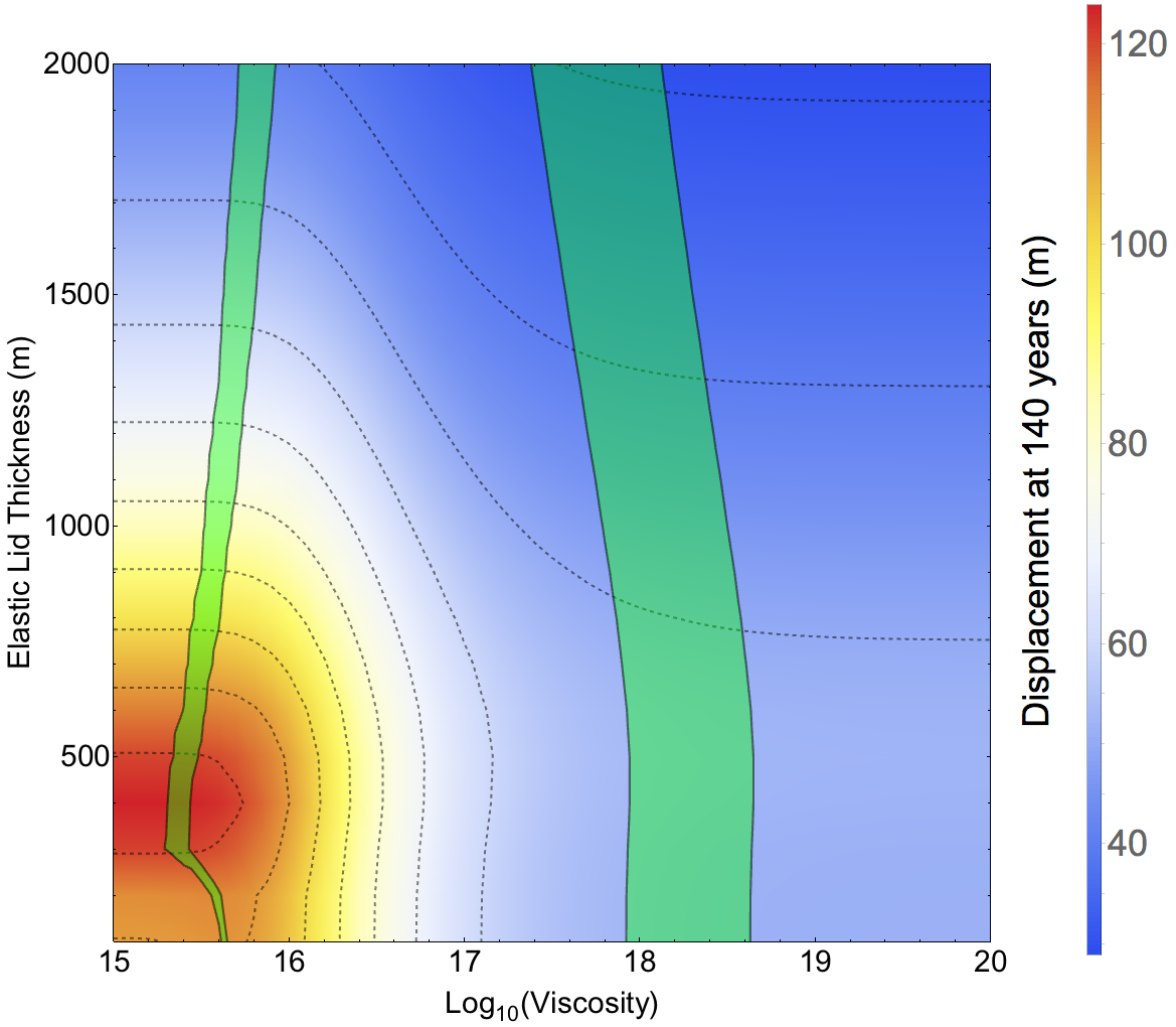


Figure 5.12: Displacements at 140 years for $\mu = 5$ GPa, $r = 1150$m, $h = 110$m. Dashed grey contours at 10m intervals. Green shaded region corresponds to displacements rates at present day between 2 and 10mm/yr to match InSAR observations of subsidence. Two acceptable regions of parameter space satisfy the InSAR, as in Figure 5.5.

proxies for strength, they are not the same: T_e reflects the integrated long-term elastic thickness of the lithosphere, whereas T_s reflects the depth to which earthquakes occur (*Watts and Burov* (2003)). Usually in oceanic lithosphere $T_e \sim T_s$, but both T_e less than T_s and T_e greater than T_s are possible because T_e can decrease with increasing curvature and bending stress and increase with the age of the lithosphere at the time of loading, in contrast to T_s which reflects local stress levels.

The relatively low viscosity in the half-space ($\sim 10^{16}$ Pa s) would require a locally elevated geotherm, which may be due to the magma plumbing system beneath

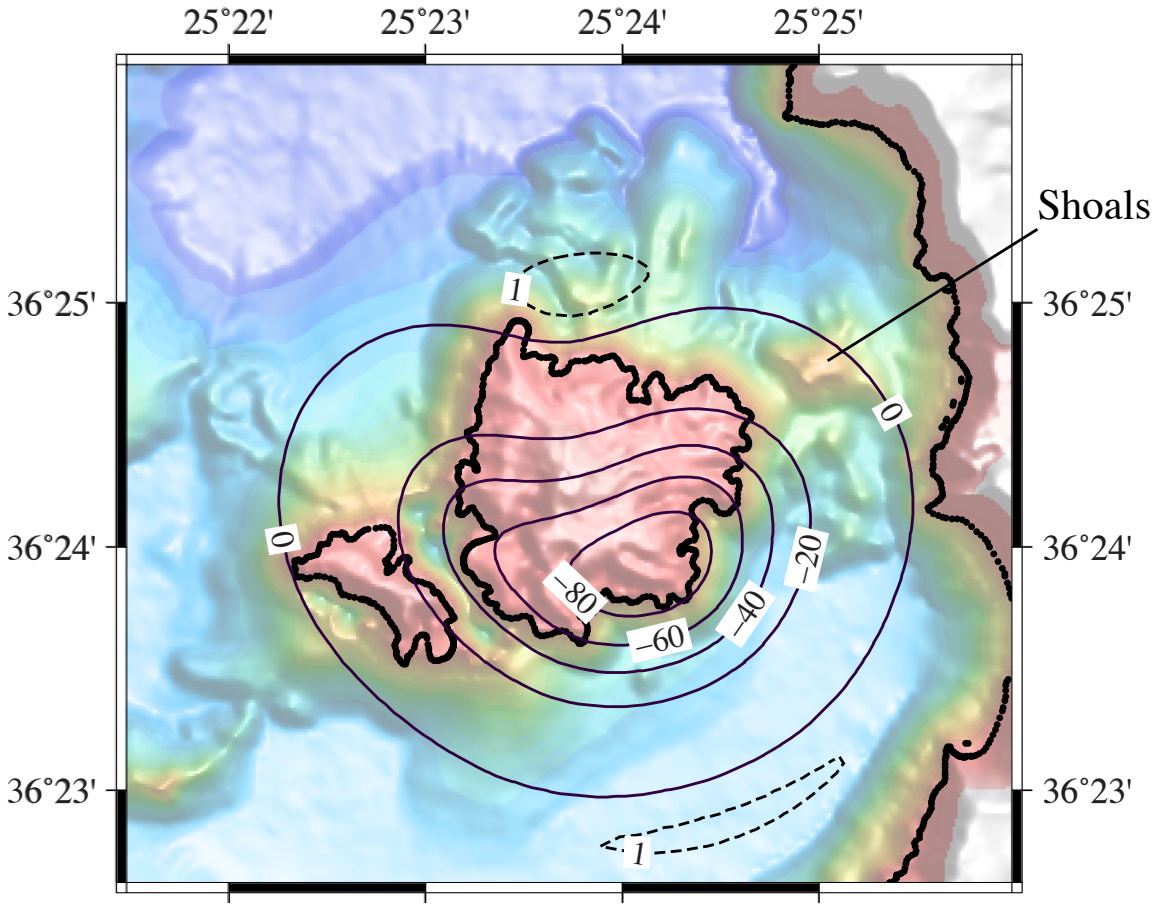


Figure 5.13: Bathymetry map of the Kameni Islands region showing the predicted subsidence and uplift at the present day assuming the submarine 1866-1870 volcanic load was emplaced on the surface of a thin elastic plate overlying a viscoelastic half-space. The subsidence and uplift have been computed for a shear modulus, μ, of 5.0×10^7 Pa, viscosity, η, of 6.3×10^{15} Pa s and elastic thickness 0.6 km. Black solid lines indicate subsidence (contour interval = -20 m). Black dashed lines indicate uplift (contour interval = +1 m).

the caldera, so I constructed a thermal model. The model has cylindrical symmetry and consists of a magma chamber buried at 5km depth held at constant temperature (900°C), with constant surface temperature (20°C) and background linear geothermal gradient (20°C/km). The resulting thermal structure can be seen in Figure 5.14a, where the green line is the required elastic thickness. If I assume a rheological flow law, such as for wet quartzite (*Hirth et al.* (2001)), then the viscosity structure follows as in Figure 5.14b. This viscosity structure differs from my model in that it is depth dependent, while I assume a constant viscosity for the loading. To compare these mod-

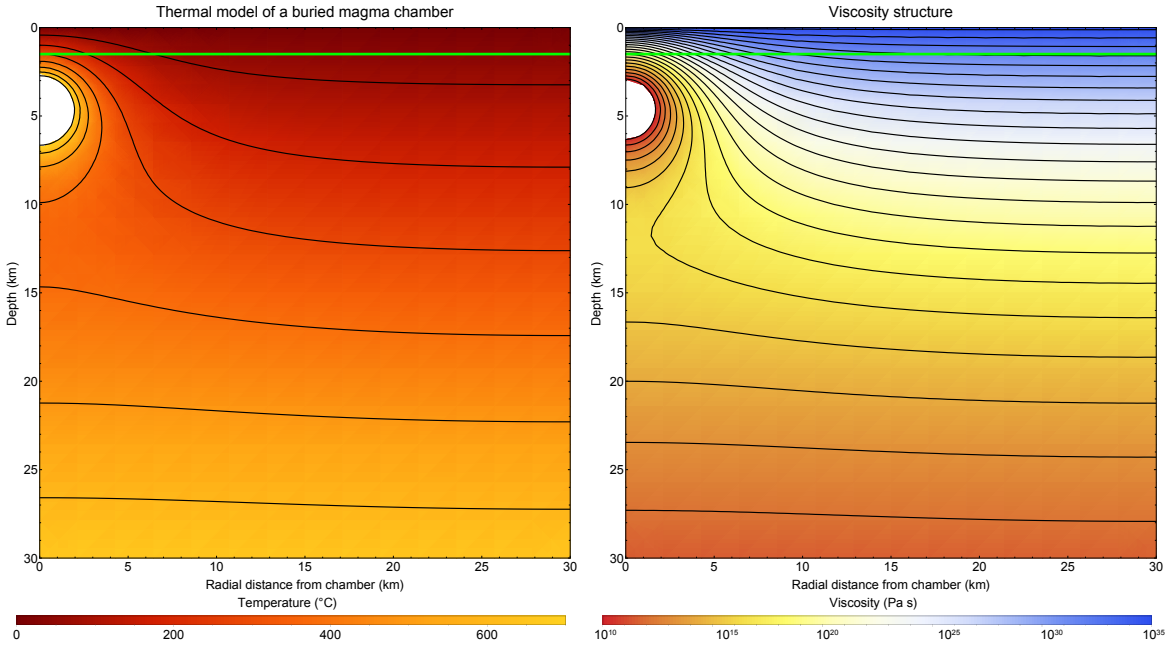


Figure 5.14: Models for temperature and viscosity structure in the vicinity of a magma chamber with radius and depth of 2 and 5 km respectively. (a) Temperature structure. Contours=100° C interval. Surface temperature = 20° C. Bottom temperature ∼ 650° C. (b) Viscosity structure assuming a wet quartz rheology. Contour interval = one order of magnitude. Surface viscosity 10^{35} Pa s.

els, I depth integrate the viscosity from the base of the elastic lid to a depth of 30km in order to calculate the mean viscosity as a function of radial distance from the chamber (Figure 5.15). The model suggests that an effective half-space viscosity of less than 10^{16} Pa s may be achieved up to a radial distance of 5km from the magma chamber. It is also interesting to note that the thermal model predicts the existence of a low viscosity shell of material surrounding the magma chamber itself.

A best fit viscosity, η, of 6.3×10^{15} Pa s implies a relatively short relaxation time, τ , of the order of $\sim$ 4 a, assuming a shear modulus, μ, of 5.0×10^{7} Pa ($\tau = \eta/\mu$). As Figure 5.16 shows (red solid line), most of the subsidence would therefore have occurred soon after the 1866-1870 loading event. This rapid subsidence is due to the fact that the lithosphere appears as a relatively strong elastic plate at short-timescales, then weakens as viscous stresses increase, resulting in increased subsidence. Figure 5.16 shows that while the total predicted subsidence is consistent with observations based

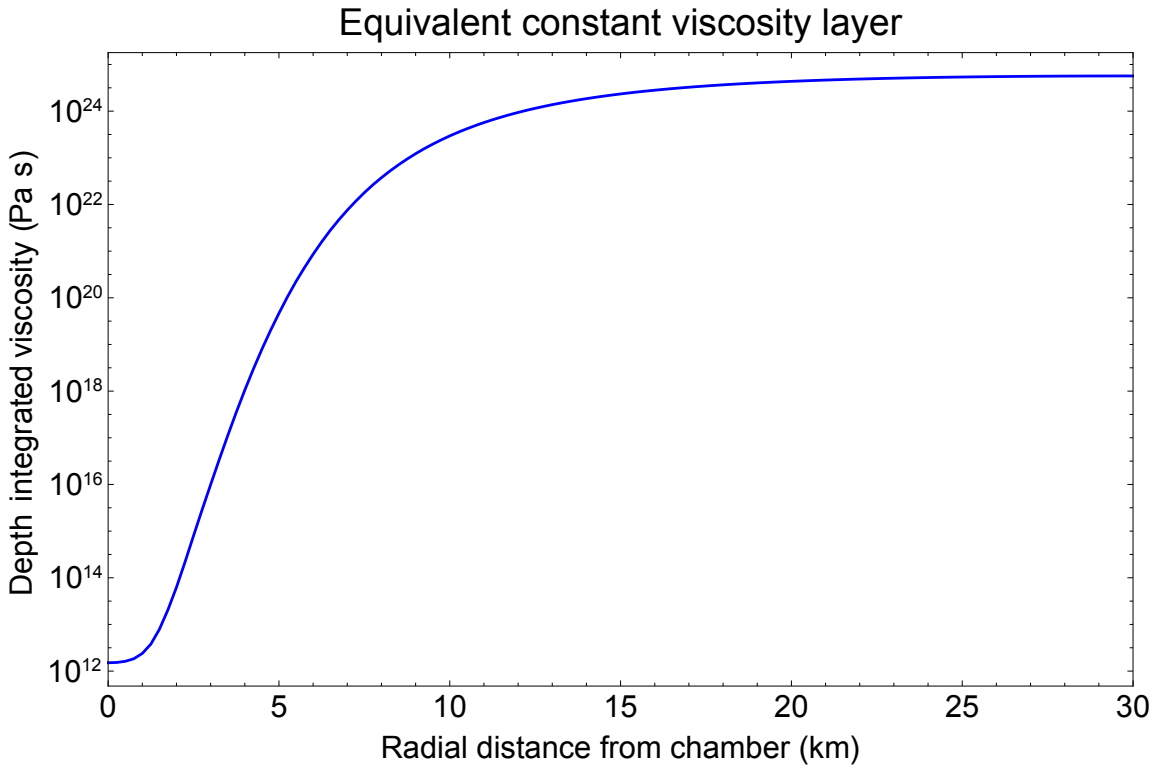


Figure 5.15: Plot of the depth integrated viscosity against radial distance from the centre of the magma chamber for the temperature and viscosity structure illustrated in Figure 5.14b. The plot shows that depth integrated viscosities of up to 10^{16} Pa s are representative of the region within a radial distance of up to 5 km from the magma chamber.

on the historical bathymetric charts, the proportion of the subsidence before and after 1925-1928 differs (Figure 5.10(f,g)), with too much subsidence predicted between 1866-1870 and 1925-1928 and too little predicted between 1925-1928 and the present day.

In order to better account for the observations, I have calculated the subsidence using the same half-space viscosity as before, but for two loads instead of one: one emplaced during 1866-1870 and the other during 1925-1928, Figure 5.10. The subsidence due to the two loads is illustrated in the blue solid line Figure 5.16. While the total subsidence is similar to the one load case, the temporal distribution of subsidence is different, with a similar amount of subsidence before 1925-1928 as after. The only difference in the best fitting parameters between the one and two load cases is that in

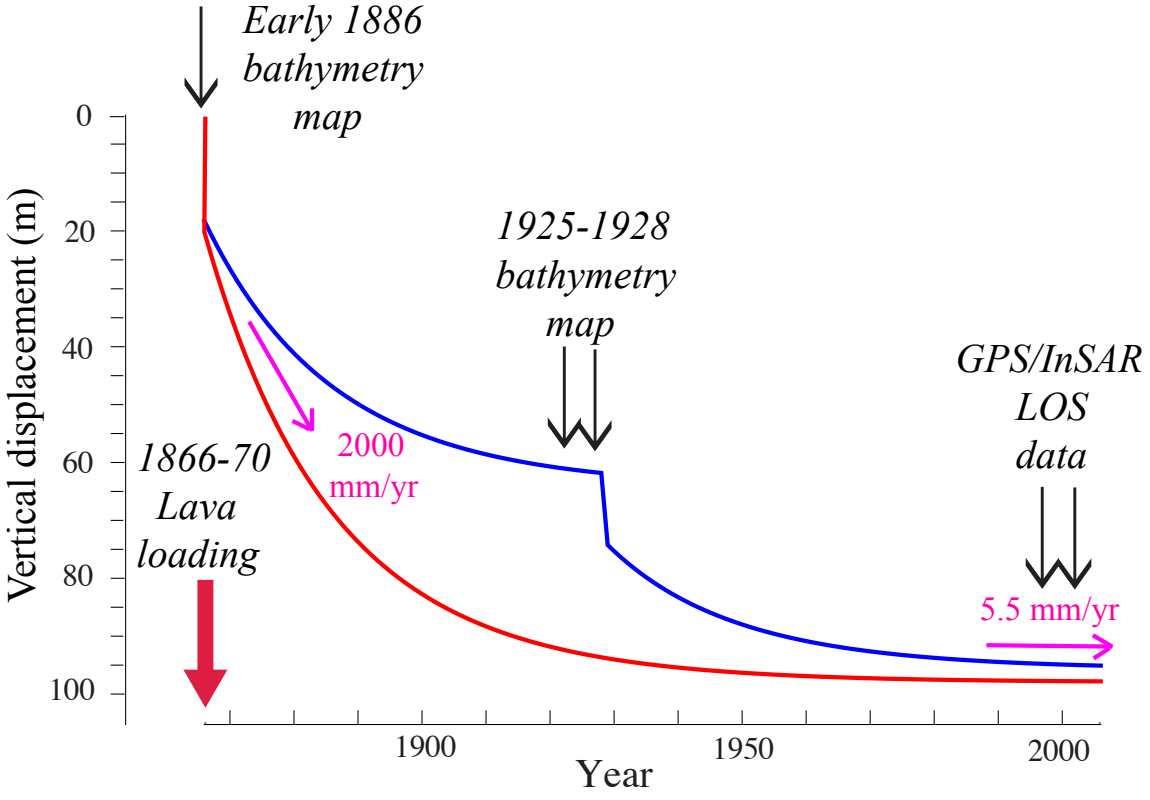


Figure 5.16: Plot of the maximum subsidence vs. year. The subsidence has been computed for the same values of shear modulus and viscosity as assumed in Figure 5.13. The red solid line shows the case of a single loading event at 1866-1870 while the blue solid line shows the case of a double loading event at 1866-1870 and 1925-1928. The elastic thickness that best accounted for the magnitude and wavelength of the observed subsidence was 0.6 km for the single loading event and 1.0 km for the double loading event. The plot shows that while both loading cases generally account for the rate of subsidence inferred by the GPS/InSAR data, only the double load case is consistent with the relative proportion of subsidence observed between 1866-1870 and 1925-1928 and between the 1925-1928 and the modern survey.

order to still explain the magnitude of the total observed subsidence, the fit including emplacement of the second load requires a thicker elastic lid, with $T_e = 1.0$ km.

5.4 Caldera-wide uplift and viscoelastic modelling

5.4.1 Introduction, joint inversion etc

Following the subsidence observed between June 1993 and June 2010, discussed in Section 5.2, Santorini entered a phase of unrest in January 2011 and persisted until around

October 2012. This period of unrest was characterised by caldera-wide inflation and was punctuated with volcano-tectonic earthquake swarms. Starting with the observed surface displacements a joint inversion technique, outlined by *Biggs et al.* (2010) and adapted from *Berardino and Fornaro* (2002), was used to convert cGPS and InSAR observations to a sub-surface volume change, by treating the displacements as arising from a point source pressure variation at depth within an elastic crust. Unlike standard joint inversion techniques, this method provides a time series of the change in source volume through time, which is more useful for interpreting changes in the behaviour of a volcano over a defined time period. I use a simple Mogi source, Section 1.3, to convert the displacement time series to volume change, then estimate the incremental volume change of the Mogi source ΔV. The primary strength of this method is that it enables us to jointly invert InSAR measurements from multiple tracks with different look angles and 3 component GPS measurements from multiple sites. The InSAR provides high spatial resolution with low temporal resolution, whilst the GPS has low spatial resolution and high temporal resolution, thus a joint inversion provide both good spatial and temporal resolution. The incremental volumes are then integrated to provide a time series of volume change, shown in Figure 5.17. More detailed information regarding both the inversion technique and error estimation may be found in *Biggs et al.* (2010).

Prior to running the inversion I wrote a Matlab script to resample the small baseline (SB) interferograms onto a grid with a spacing of 200 m up to an inner radius of 4 km from the source, and a coarser spacing of 400 m outside this area. This increased the number of sample points close to the source and reduced the number of sample points in the inversion that are more likely to be affected by atmospheric delay on the eastern edge of Thera. For each dataset, we searched a three-dimensional grid of 200 m spacing in x, y and z to determine the optimal source location which minimised the misfit between the data and the model. The error on the total volume estimates for each of the inversions is the cumulative sum of the weighted root mean square (WRMS) residual for individual observations (computed during the inversion). The

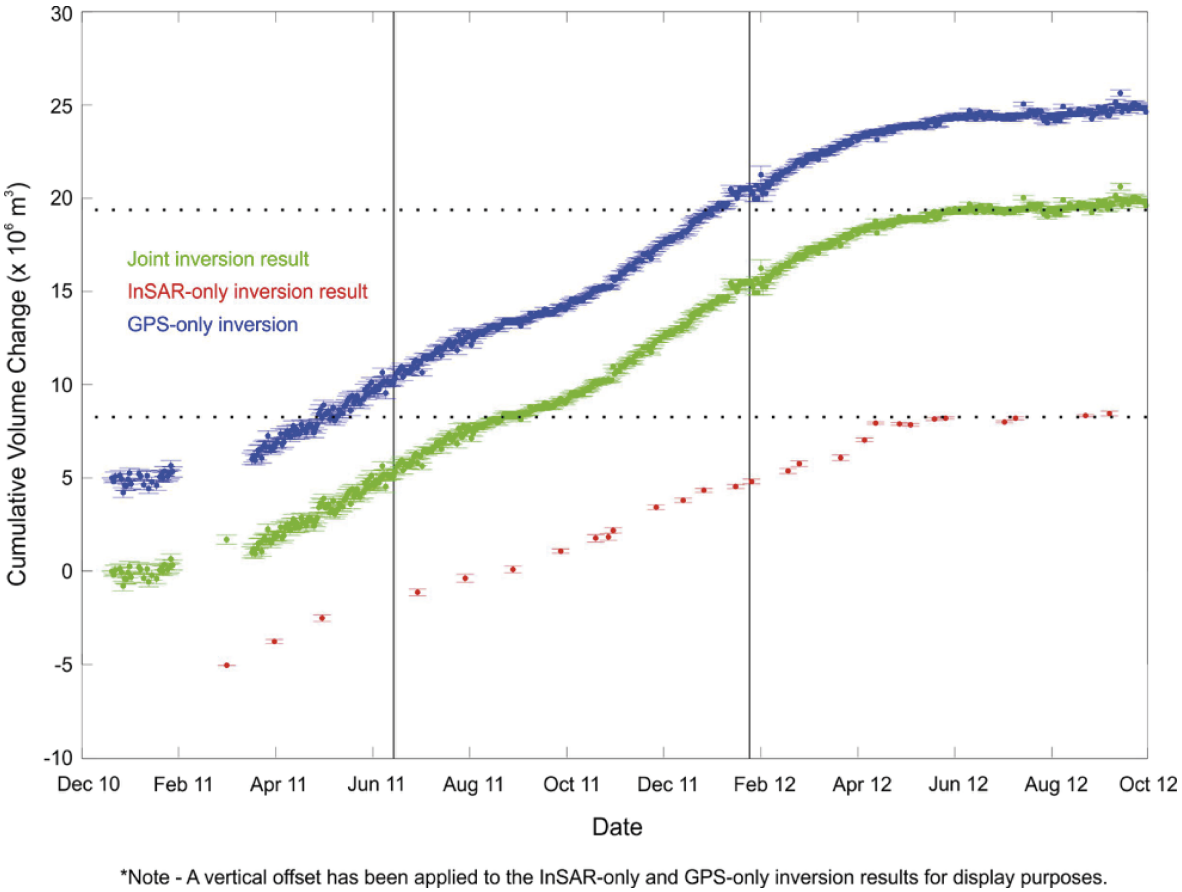


Figure 5.17: Temporal evolution of volume change derived from the inversion of cGPS measurements for 10 sites, from December 2010 to October 2012 (blue), of SB InSAR measurements from March 2011 to September 2012 (red) and Joint inversion result covering the period December 2010 to October 2012 (green). A vertical offset of $\pm 5 \times 10^6$ m^3 has been applied to the GPS-only and InSAR-only inversion results respectively for display purposes. Figure reproduced from *Parks et al.* (2014).

optimum depth was located using the minimum WRMS derived during the inversion. The bounds on the depth estimate were derived using the volume change error estimates and the trade-off plot between depth and volume, shown in Figure 5.18.

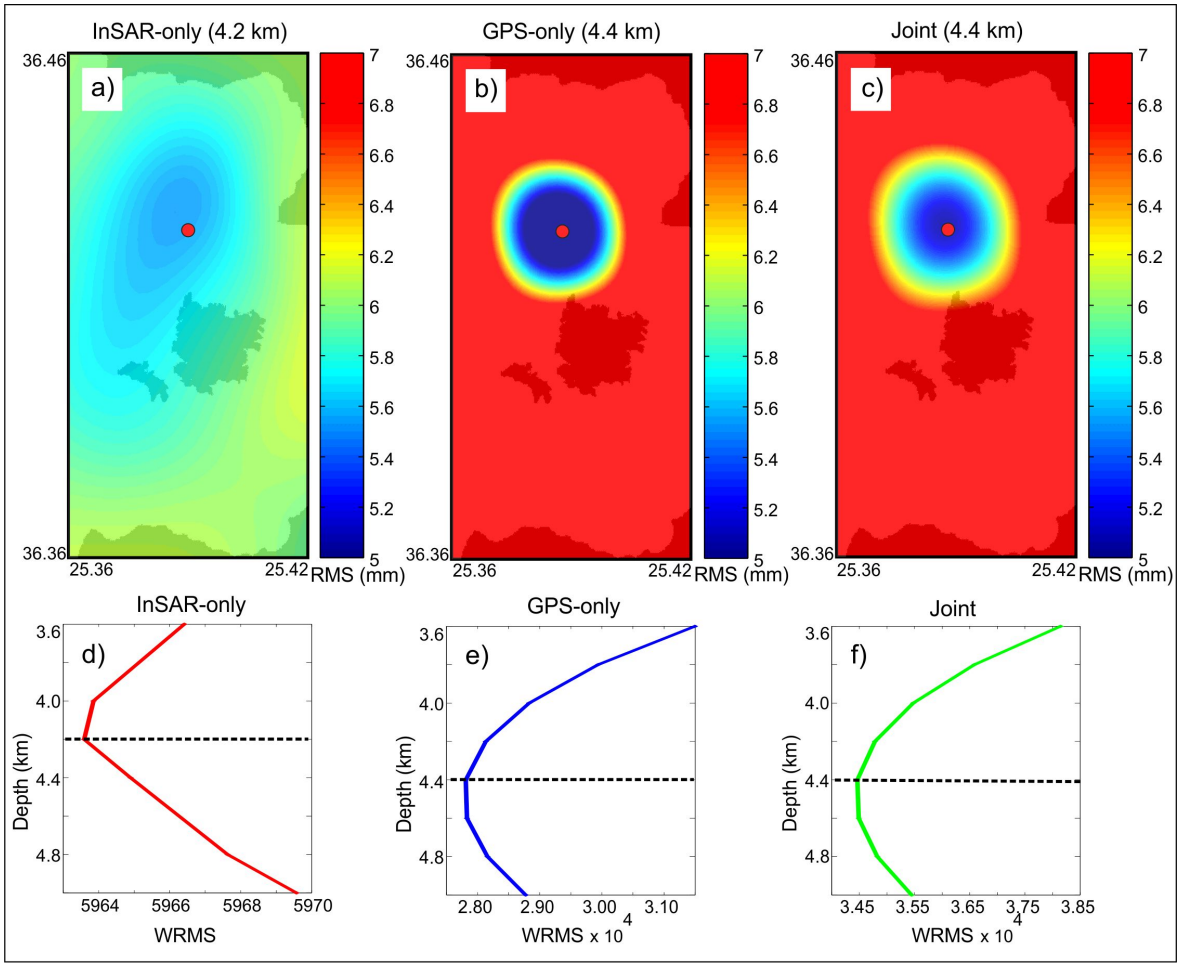


Figure 5.18: Source location plots. (a & d) Best-fit source location derived from the InSAR-only volumetric inversion. (b & e) Best-fit source location derived from the GPS-only volumetric inversion. (c & f) Best-fit source location derived from the Joint volumetric inversion. The red filled circle in (a)-(c), represents the best-fit geographic location of the source of the deformation as determined in this study. Figure reproduced from *Parks et al.* (2014).

5.4.2 Viscoelastic Modelling

Field evidence and thermal models demonstrate that thermal aureoles develop in the host rock surrounding magma intrusions (*Gasparini et al.* (1981)). There is strong evidence for the existence of a thermal or metamorphic aureole beneath the Kameni islands, both from basement rock inclusions in Kameni lavas (*Nicholls* (1971)) and from the carbon-isotopic compositions of Kameni CO_2 emissions (*Parks et al.* (2013); *Tassi et al.* (2013)). Temperatures in the thermal aureole and the outer margin of the

magma chamber may exceed the brittle-ductile transition, potentially to a sufficient extent to allow ductile flow and stress relaxation in the vicinity of the magma chamber. I therefore investigate whether a viscoelastic rheology may be invoked to explain the pattern of increase and subsequent decay in cumulative volume change observed twice during the period of unrest. The increase in the cumulative volume change appears to decay exponentially, a signal that can be indicative of viscoelastic stress relaxation (as seen in the temporal plots from Chapter 4).

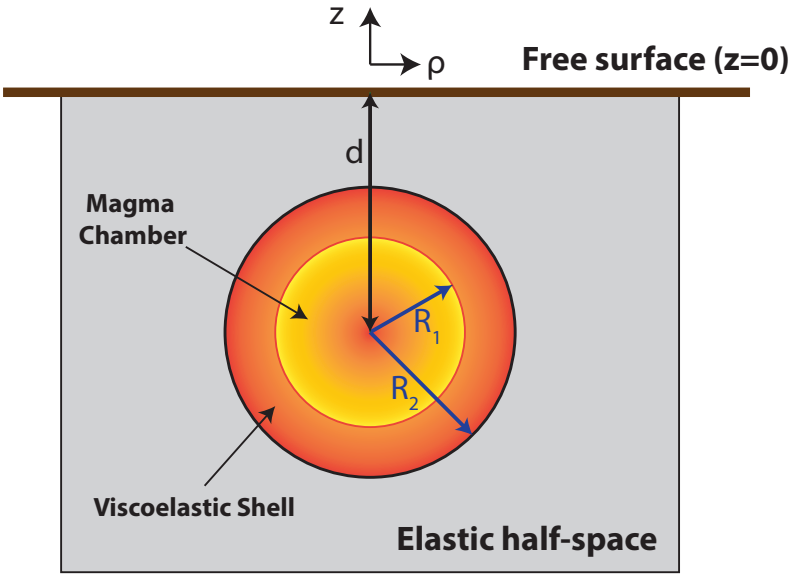


Figure 5.19: A spherical magma chamber, inner radius R_1 surrounded by a viscoelastic shell, outer radius R_2, embedded in an elastic half-space at a depth d beneath the free surface.

I modelled the observed temporal variation in volume change (derived from the joint inversion) associated with both pulses in terms of the ductile deformation of a concentric viscoelastic shell surrounding a shallow magma chamber, driven by two separate increases in pressure, which I interpret as pulses of melt supply. My viscoelastic model consists of a magma chamber located at a depth, d, beneath the surface, with a radius R_1. The magma chamber is surrounded by a viscoelastic shell of radius of $R_2 - R_1$ and rigidity μ, embedded in an elastic half-space, as illustrated in Figure 5.19.

Several studies at other active volcanoes have used viscoelastic modelling to better constrain source parameters responsible for observed deformation (*Bonafede* (1986); *Dragoni and Magnanensi* (1989); *Newman et al.* (2001); *Hickey et al.* (2013)). *Bonafede* (1986) were the first to explore the effect of the anelastic properties of the Earth's crust in relation to volcanic deformation. These authors presented analytical expressions to model the deformation produced by a pressure source embedded in a viscoelastic half-space. *Dragoni and Magnanensi* (1989) expanded on this study by considering a magma chamber surrounded by a viscoelastic shell embedded in an infinite medium (full-space). The analytical solutions of *Dragoni and Magnanensi* (1989) are very useful in determining the variation in the shape of the deformation pattern over time; however the output may not be directly compared with GPS or InSAR measurements. This is because the elastic full-space solution would underestimate the observed deformation (by a factor of 3 if using a Poissons ratio of 0.25) due to the exclusion of the free-surface (numerical calculations in *Newman et al.* (2001)).

More recent studies (*Newman et al.* (2001); *Masterlark et al.* (2010); *Hickey et al.* (2013)) have used a finite-element approach to model viscoelastic deformation within an elastic half-space. Although the finite-element method enables modelling of magma chambers with increased complexity, these models are often time-consuming, computationally intensive and difficult to validate. *Segall* (2010) presented two analytical solutions to model vertical displacements due to a pressurised sphere in an elastic half-space, based on a modification of *Dragoni and Magnanensi* (1989) original full-space solution. The first of the solutions in *Segall* (2010) models the pressure change within the magma chamber as an instantaneous step increase (Heavyside function) whereas the second models the pressure change as a gradual increase followed by an exponential decay (*Segall* (2010), pp 243-248). Although I applied both solutions in an attempt to model the variation in volume change and associated deformation observed at Santorini between December 2010 and October 2012, the solutions did not provide a good fit to the observations. This suggests that the pressure increase within the shallow magma

chamber at Santorini was not instantaneous, nor did it gradually decay with time. Here I present a new analytical solution to model the variation in volume change within a shallow magma chamber surrounded by a viscoelastic shell considering an alternate pressure history.

Following the derivation of *Dragoni and Magnanensi* (1989), after *Segall* (2010) I solve for a pressurised spherical chamber surrounded by a viscoelastic shell, illustrated in Figure 5.19. For parameters, subscript 1 corresponds to the viscoelastic shell and subscript 2 corresponds to the surrounding region. The elastic solution for a pressurised sphere in a shell ($R_1 \leq r \leq R_2$) with shear modulus μ_1 embedded in a half space ($r \geq R_2$) with shear modulus μ_2 is, modified from Section 1.3,

$$u_r(R_1 \leq r \leq R_2) = A\frac{r}{3} + \frac{B}{r^2}, \tag{5.3a}$$

$$u_r(r \geq R_2) = \frac{C}{r^2}, \tag{5.3b}$$

$$\sigma_{rr}(R_1 \leq r \leq R_2) = K_1 A - \frac{4\mu_1 B}{r^3}, \tag{5.3c}$$

$$\sigma_{rr}(r \geq R_2) = -\frac{4\mu_2 C}{r^3}. \tag{5.3d}$$

Equating the stresses and displacements on the interface and taking the stress at the inner boundary to be equal to the pressure ($\sigma(r = R_1) = -p$) allows us to solve for the unknowns

$$A = -3pR_1^3(\mu_1 - \mu_2)/D, \tag{5.4a}$$

$$B = -pR_1^3R_2^3(3K_1 + 4\mu_2)/4D, \tag{5.4b}$$

$$C = -pR_1^3R_2^3(3K_1 + 4\mu_1)/4D, \tag{5.4c}$$

$$D = 3K_1R_1^3(\mu_1 - \mu_2) - \mu_1R_2^3(3K_1 + 4\mu_2). \tag{5.4d}$$

The correspondence principle is then utilised, defining a Maxwell rheology in the shell

surrounding the magma chamber embedded in an elastic infinite media where

$$\mu_1 = \frac{s}{\mu^{-1}s + \eta^{-1}}, \tag{5.5a}$$

$$\mu_2 = \mu, \tag{5.5b}$$

$$p = \bar{p}(s). \tag{5.5c}$$

Substituting equations 5.5 into equations 5.4 gives

$$\bar{A}(s) = -\frac{\bar{p}(s)}{2\eta}\left(\frac{1-2\nu}{1-\nu}\right)\left(\frac{R_1}{R_2}\right)^3 \bar{D}^{-1}(s), \tag{5.6a}$$

$$\bar{B}(s) = \frac{\bar{p}(s)R_1^3}{4\mu}\left(s + \frac{\mu}{\eta}\right) D^{-1}(s), \tag{5.6b}$$

$$\bar{C}(s) = \frac{\bar{p}(s)R_1^3}{4\mu}\left[s + \frac{\mu(1+\nu)}{3\eta(1-\nu)}\right] \bar{D}^{-1}(s), \tag{5.6c}$$

$$\bar{D}(s) = s + \frac{\mu(1+\nu)R_1^3}{3\eta(1-\nu)R_2^3}. \tag{5.6d}$$

Note that at this stage the pressure history in Laplace transform space, $\bar{p}(s)$ remains undefined. The solutions are thus given by

$$u_r(R_1 \leq r \leq R_2) = \mathcal{L}^{-1}\left\{\bar{A}(s)\right\}\frac{r}{3} + \frac{\mathcal{L}^{-1}\left\{\bar{B}(s)\right\}}{r^2}, \tag{5.7a}$$

$$u_r(r \geq R_2) = \frac{\mathcal{L}^{-1}\left\{\bar{C}(s)\right\}}{r^2}, \tag{5.7b}$$

where the new coefficients are the inverse Laplace transforms of equations 5.6.

The radial displacements are currently relative to the centre of the magma chamber in an infinite media and I must now consider a magma chamber in a half space, centered on $z = d$. Converting from spherical polar co-ordinates (r, θ, ϕ) to cylindrical polar coordinates $(\hat{\rho}, \theta, z)$ gives

$$u_z^{(s)}(\rho, z = 0, t) = \mathcal{L}^{-1}\left\{\bar{C}(s)\right\}\frac{1}{d^2(1+\rho^2)^{3/2}}, \tag{5.8a}$$

$$u_\rho^{(s)}(\rho, z = 0, t) = \mathcal{L}^{-1}\left\{\bar{C}(s)\right\}\frac{\rho}{d^2(1+\rho^2)^{3/2}}, \tag{5.8b}$$

for the vertical and radial displacements on the surface $z = 0$, where the superscript (s) indicates that this is displacement due to the source in a full space. The radial coordinate has been non-dimensionalised by the depth of the source ($\rho = \hat{\rho}/d$). However, this leaves us with non-zero surface stresses, when the $z = 0$ surface must be stress free. Using the method of images, after *McTigue* (1987), we can apply equal and opposite tractions to the surface of the elastic half space to obtain a solution accurate to $O(R_2/d)^3$. The displacements generated by the applied surface tractions are

$$u_z^{(i)}(\rho, z = 0, t) = \mathcal{L}^{-1}\left\{\bar{C}(s)\right\} \frac{(3 - 4\nu)}{d^2(1 + \rho^2)^{3/2}}, \tag{5.9a}$$

$$u_\rho^{(i)}(\rho, z = 0, t) = \mathcal{L}^{-1}\left\{\bar{C}(s)\right\} \frac{(3 - 4\nu)\rho}{d^2(1 + \rho^2)^{3/2}}. \tag{5.9b}$$

It should be noted that the free surface correction introduces a factor of $4(1-\nu)$, which agrees with the numerical findings of *Newman et al.* (2001). The total displacement is given by the sum of equations (5.8) and (5.9), $u = u^{(s)} + u^{(i)}$,

$$u_z(\rho, z = 0, t) = \mathcal{L}^{-1}\left\{\bar{C}(s)\right\} \frac{4(1 - \nu)}{d^2(1 + \rho^2)^{3/2}}, \tag{5.10a}$$

$$u_\rho(\rho, z = 0, t) = \mathcal{L}^{-1}\left\{\bar{C}(s)\right\} \frac{4(1 - \nu)\rho}{d^2(1 + \rho^2)^{3/2}}. \tag{5.10b}$$

Drawing analogies with the elastic Mogi model with a time dependent volume change,

$$u_z = \frac{(1 - \nu)\Delta V(t)}{\pi d^2} \frac{1}{(1 + \rho^2)^{3/2}}, \tag{5.11a}$$

$$u_\rho = \frac{(1 - \nu)\Delta V(t)}{\pi d^2} \frac{\rho}{(1 + \rho^2)^{3/2}}, \tag{5.11b}$$

I can derive an equivalent time dependant volume change for our viscoelastic relaxation model,

$$\Delta V(t) = 4\pi\mathcal{L}^{-1}\left\{\bar{C}(s)\right\}. \tag{5.12}$$

Equation (5.12) is still in terms of the inverse Laplace transform of equation (5.6). This

transform can be calculated analytically for a number of pressure histories $p(t)$. Table 5.2 gives a sample of 3 pressure functions and the associated $\bar{p}(s)$.

$p(t)$	$\bar{p}(s)$
$p_0 H(t)$	$\frac{1}{s}$
$p_0(t + (t_1 - t)H(t - t_1))/t_1$	$p_0(1 - e^{-st_1})/(s^2 t_1)$
$p_0(1 - \exp(-t/t_s))$	$p_0(s^{-1} - (s + t^{-1})^{-1})$

Table 5.2: Pressure functions, where $H(t)$ is the Heavyside function.

The pressure history described by the second function in Table 5.2 is the novel calculation that we are most interested in, as it describes a linear increase in pressure corresponding to an injection of magma at a constant rate that shuts off at a time t_1. Calculating $\mathcal{L}^{-1}\left\{\bar{C}(s)\right\}$ with $\bar{p}(s) = p_0(1 - e^{-st_1})/(s^2 t_1)$ gives us:

$$\bar{C}(s) = \frac{p_0 R_1^3 \left(1 - e^{-st_1}\right)\left(\frac{\mu(\nu+1)}{3\eta(1-\nu)} + s\right)}{4\mu s^2 t_1 \left(\frac{\mu(\nu+1)R_1^3}{3\eta(1-\nu)R_2^3} + s\right)} \tag{5.13}$$

Taking the inverse Laplace transform.

$$\mathcal{L}^{-1}\left\{\bar{C}(s)\right\} = \frac{p_0 R_2^3}{4\mu^2(\nu+1)t_1 R_1^3}\Bigg[R_1^3 f(t, t_r, \eta, \nu) + R_1^3\mu(1+\nu)t - R_2^3 f(t, t_R, \eta, \nu)$$

$$+ (R_2^3 f(t - t_1, t_R, \eta, \nu) - R_1^3 f(t - t_1, t_R, \eta, \nu) - R_1^3(t - t_1)\mu(1+\nu))H(t - t_1)\Bigg], \tag{5.14a}$$

with

$$f(t, t_R, \eta, \nu) = 3\eta(1-\nu)\left(1 - e^{-\frac{t}{t_R}}\right), \tag{5.14b}$$

$$t_R = \frac{3\eta(1-\nu)R_2^3}{\mu(1+\nu)R_1^3}. \tag{5.14c}$$

Equations (5.14) can then be substituted into equation (5.12) and simplified to give

the equivalent volume change for an elastic Mogi source, which I used to fit the data,

$$\Delta V(t) = \frac{\alpha t_R}{t_1}\left[\hat{R}^3\left(\frac{t}{t_R} + e^{-\frac{t}{t_R}} - 1\right) + 1 - e^{-\frac{t}{t_R}} ... \right.$$
$$\left. - H(t-t_1)\left(\hat{R}^3\left(\frac{t-t_1}{t_R} - 1\right) + 1 + \left(\hat{R}^3 - 1\right)e^{(t_1-t)/t_R}\right)\right], \tag{5.15a}$$

with

$$\hat{R} = R_2/R_1, \tag{5.15b}$$
$$t_R = \frac{3\eta(1-\nu)R_2^3}{\mu(1+\nu)R_1^3}, \tag{5.15c}$$
$$\alpha = \frac{p_0\pi R_1^3}{\mu}. \tag{5.15d}$$

It should be noted that this model has the same underlying assumptions of the Mogi model, *Mogi* (1958), and produces displacements that are accurate to order $(R_2/d)^3$. In the case of Santorini, this introduces an error of approximately 1%. Given that this is smaller than the sources of error in both the InSAR and GPS datasets, I choose to neglect it. I suggest that the observed variations in seismicity and the rate of volume change of the shallow magma chamber (Figure 5.20) are intrinsically connected and result from two separate phases of increasing pressure within the shallow reservoir. I then transform the final equation so the output is given in terms of volume change rather than deformation, using an equivalence relation to the time dependent Mogi model. That allows me to use our best fitting Mogi source, with depth of 4.5km from the joint inversion, and formulate the viscoelastic model in terms of effective volume change.

My pressure function corresponds to two consecutive ramp functions, which represents a linearly increasing pressure to the shallow chamber between times t_0 and t_1, a fixed pressure from t_1 to t_2, linearly increasing pressure between t_2 and t_3 and a constant pressure from t_3 onwards (Figure 5.20). If I assume a fixed radius then the pressure change within the shallow magma chamber may be represented in terms of volume change as in Section 1.3, and the time dependent volume change is given by

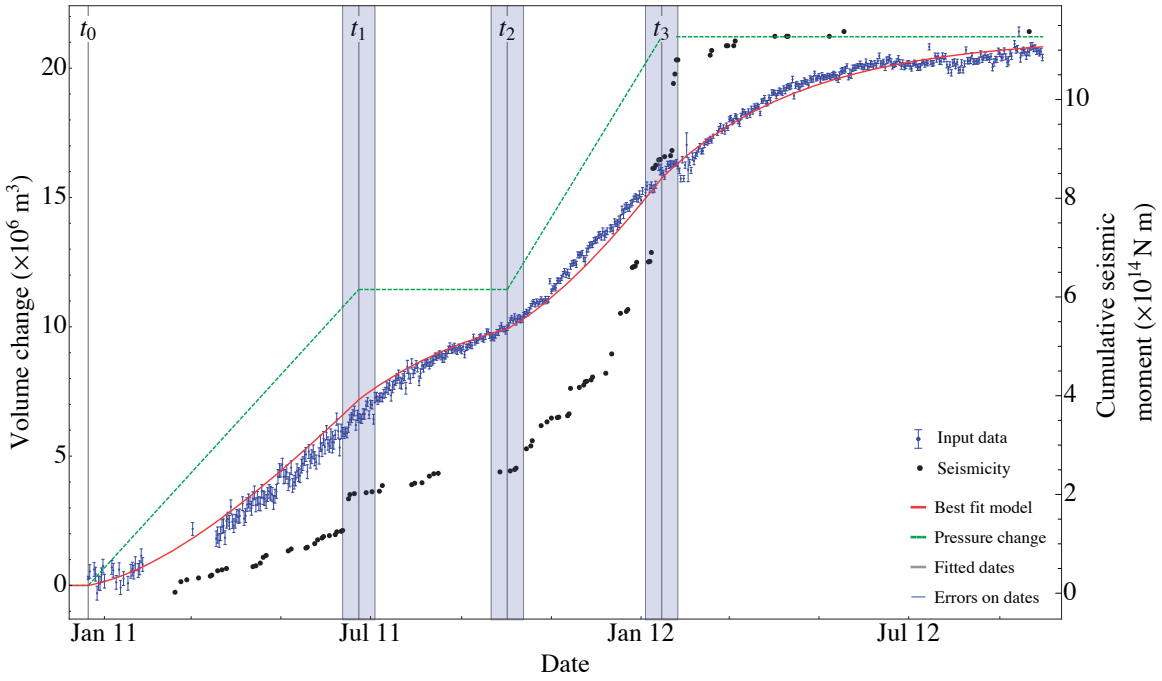


Figure 5.20: Comparison between the viscoelastic relaxation model (red), joint inversion result (blue) and the cumulative seismic moment released during VT earthquakes $> M_L 2.0$ occurring within the caldera (black dots). The green lines represent the normalised pressure model, displaying a linearly increasing pressure from t_0 to t_1 and from t_2 to t_3. In between t_1 and t_2 and after t_3 the pressure within the shallow magma chamber is held constant.

equation (5.15).

The predicted volume change is fitted to the data, weighted by the errors between consecutive observations using a least squares minimisation routine in Mathematica 9.0 (*Wolfram Research* (2012)). I solve for the ratio of the outer to inner radius (R_2/R_1), the characteristic relaxation time (t_R), the shut-off times (t_1 and t_3), the onset time for the second pressure pulse (t_2) and a scaling coefficient α, which is a combination of the pressure increase within the magma chamber and the chamber radius (R_1). The only constraints I apply to the minimisation are $t_3 > t_2 > t_1$, the initial guesses are given in Table 5.3, and the best fit is plotted in Figure 5.20.

The best-fit parameters minimising the misfit between the volume change derived from the joint inversion and the viscoelastic model are displayed in Table 5.4. The

Relaxation timescale (t_r)	70 days
Total volume change from first pulse (α_1)	Max($\Delta V/2$)
Total volume change from second pulse (α_2)	Max($\Delta V/2$)
Ratio of shell to chamber radii ($R = R_2/R_1$)	1.5
Cessation of first pulse (t_1)	150 days
Start of second pulse (t_2)	300 days
Cessation of second pulse (t_3)	400 days

Table 5.3: Initial guesses for least squares minimisation to fit viscoelastic shell model.

pressure and radius terms in the Mogi approximation are coupled (*Mogi* (1958)) and it is not possible to individually determine either parameter. However, a trade-off curve between pressure change and radius may be used to determine a range of physically viable parameters. The computed trade-off curve is displayed in Figure 5.21 and can be used to calculate the chamber size for any given pressure change. For example, if the overpressure in the shallow chamber was increased to 10 MPa this would dictate a chamber radius (R_1) of $\sim$ 1200 m and an outer radius (R_2) of $\sim$ 2000 m). The effective viscosity of the shell was calculated using the relaxation timescale (Table 5.4) and equation (5.15c); specifying both the shear modulus ($\mu = 5$ GPa) and Poisson's ratio ($\nu = 0.25$).

η(Pa s)	$5.2 \pm 0.5 \times 10^{15}$
R_2/R_1	1.73 ± 0.14
t_0	20$^{\text{th}}$ December 2010
t_1	23$^{\text{rd}}$ June 2011
t_2	7$^{\text{th}}$ October 2011
t_3	15$^{\text{th}}$ January 2012
t_R (days)	101 ± 4
ΔV_1 (m^3)	$(11.56 \pm 0.14) \times 10^6$
ΔV_2 (m^3)	$(9.73 \pm 0.10) \times 10^6$

Table 5.4: Best-fit parameters derived from the viscoelastic model with 99% confidence errors. η is the viscosity of the viscoelastic shell surrounding the magma chamber. R_2/R_1 is the ratio of the outer-inner radius. t_0 and t_2 are the onset times of the increasing pressure pulses. The shut-off times are t_1 and t_3. t_R is the relaxation timescale.

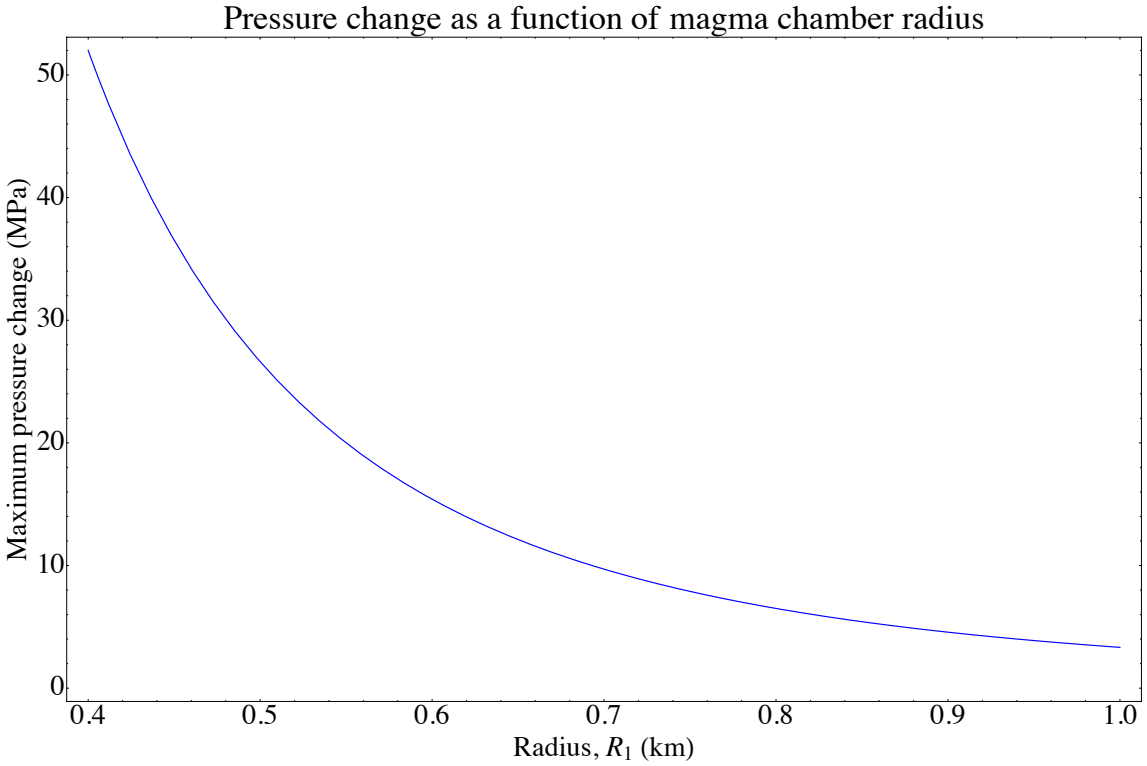


Figure 5.21: Trade-off curve for the viscoelastic shell model for pressure change in the chamber p_0 and chamber radius R_1.

The best-fit model provides a good match to the joint inversion result and the derived shell viscosity ($5.2 \pm 0.5 \times 10^{15}$ Pa s) and R_2/R_1 ratio (1.73) are within the expected range, considering values reported from other studies (*Newman et al.* (2001) and references therein). The derived shut-off times (t_1 and t_3) for the pressure pulses also provide a good fit to the observed cumulative seismicity (Figure 5.20); coinciding with a lull in earthquake activity, following a spike in cumulative seismic moment. Between t_1 and t_2, and after t_3, the pressure within the magma chamber is held constant (this may be achieved via a number of processesone of which could involve vesiculation and concurrent degassing), but the apparent cumulative volume change continues to increase at an exponentially decaying rate.

This viscoelastic response explains the lag between the cessation in earthquake activity and the flattening of the cumulative volume change curve shown in Figures 5.17

and 5.20. The VT earthquakes cease following the removal of the source of increasing pressure, but the viscoelastic shell surrounding the magma chamber acts as a buffer, resulting in a delayed decline in deformation driven by the pressure shut-off.
The timing of the second pressure shut off (t_3) corresponds to a significant earthquake swarm at the end of January 2012, which was followed by an almost complete cessation in micro-seismic activity in the centre of the caldera.

5.5 Conclusions

Prior to the start of 2011, a slow subsidence signal persisted at Nea Kameni with no evidence from InSAR of any uplift since 1993 or from triangulation since 1955 (*Parks et al.* (2012)). My modelling suggests that this subsidence could be accounted for by a combination of thermal contraction and viscoelastic relaxation of the substrate associated with recent lava flows. There is a strong correspondence between the signal and the outline of the 1866-1870 lava flow, the most voluminous and spatially extensive historical flow. The observed subsidence rate between 1993-2010 may be accounted for by modelling both thermal contraction and viscoelastic relaxation.

Comparisons of historical bathymetric charts and modern swath bathymetry shows the seafloor in the vicinity of the Kameni Islands has in some regions shallowed by up to $\sim$ 200m and deepened in others by up to $\sim$ 100m, which I attribute to the emplacement of lava on the seafloor during the 1866-1870 eruption driving viscoelastic relaxation of the substrate. My model explains both the seafloor deepening over the past 140 years and the modern rates of subsidence derived from InSAR observations between 1993 and 2010. The best fitting model for the historic and modern data sets incorporates both the 1866-1870 and 1925-1928 loads, emplaced on the surface of an elastic lid of thickness 1.0 km overlying a viscoelastic half-space with viscosity 6.3×10^{15} Pa s, or relaxation timescale of $\sim$ 4 years. The subsidence model only requires these low viscosities inside the caldera, which I propose is due to the elevated thermal structure

generated. Thermal modelling suggests that the required viscosities can be produced by a magma chamber at 4.5km depth.

A new phase of unrest commenced at Santorini volcano in January 2011, characterised by the onset of micro-seismic activity focussed along the Kameni line and caldera wide uplift. A joint inversion of InSAR and GPS data was performed to determine the temporal evolution of source parameters by converting both InSAR and GPS observations into a time series of sub-surface volume change. The best fitting parameters derived from the combined InSAR-GPS inversions are in good agreement with other studies (*Newman et al.* (2012); *Parks et al.* (2012)) and suggest that the point pressure source is centred north of Nea Kameni at a depth of ~ 4 km beneath the surface and has an effective volume of $\sim 14-23\times10^6$ m^3. The time series of sub-surface volume change has identified two separate pressure pulses during the period of unrest. A lag response is also identified between the lulls in micro-seismicity activity within the caldera and the exponential decay observed in the rate of sub-surface volume change several months later.

I propose that the two pulses observed are associated with episodes of magma supply to a shallow reservoir and that the resulting pressure increase within this reservoir is responsible for the VT earthquake swarms focussed along the pre-existing zone of weakness, the Kameni line. I have modified the model of the shallow magma chamber beneath Santorini volcano by considering the magma chamber as a sphere inside a viscoelastic shell (representing a ductile aureole) in order to account for the lag between the seismicity and shape of the volume change time series. The thermal modelling used to explain the relatively low viscosities required for the subsidence model also produces a localised shell of relatively low viscosity material around the chamber itself. It is currently not possible to determine when or if a new pulse of magma will be supplied to the existing shallow chamber, or indeed if the next pulse will bypass this chamber completely and be intruded at an alternate location. However, this information does

provide additional insight into the subsurface rheology and the temporal evolution of magma supply during periods of unrest at Santorini. The next phase of unrest will provide important constraints on the time period between these rapid pulses. Continuous deformation, seismicity and gas monitoring are essential to identify the onset of renewed unrest.

Chapter 6

Discussion and Conclusions

In this thesis I have set out to quantify strength in the continents by investigating deformation associated with the earthquake cycle, inflation of magma chambers beneath volcanoes and changes in surface loads. I have investigated timescales from short (months to years) to intermediate (hundreds to thousands of years), as in Figure 1.1. Results of these models have important implications for our understanding of large-scale continental deformation, in addition to both seismic and volcanic hazard assessment.

I have taken the existing finite element code *oregano_ve* and modified it to include periodic boundary conditions, so that I may investigate anti-plane problems. Once the code was modified, I developed a suite of benchmarking tests to ensure it was operating correctly. I derived a series of novel analytical solutions for simple shear of a rectangular block with viscosity structures to match those in *oregano_ve*, and a series of novel results for viscoelastic relaxation of imposed harmonic tractions and displacements. I examined the effect of a harmonic shear stress or displacement condition at the surface of a 2D plane, with motions assumed to be anti-plane. I found the time constant associated with the relaxation was dependent on the wavelength of the forcing stress or displacement condition (though the two cases did not behave in the same fashion).

Short wavelength components of imposed shear stresses relax more quickly than longer wavelength components and have smaller maximum amplitude. Physically this

would mean that components of stress with relatively short wavelengths will decay with the classic maxwell relaxation time, but longer wavelength stress fields can persist for much greater lengths of time. The reverse is true for displacement conditions, where longer wavelength displacements decay more rapidly. My interpretation of this dichotomy is that the stresses for short wavelength components diffuse outwards more rapidly than longer wavelength components, whilst short wavelength displacements are filtered by the elastic lid and are therefore restricted in how rapidly they can diffuse, until $w \gg T_e$. Multiple mechanisms have been involved in the literature to explain the observed variability in decay of short versus long wavelength signals (*Hetland and Hager* (2005, 2006); *Barbot et al.* (2008); *Yamasaki and Houseman* (2012b)) and this study shows that multiple timescales may be involved with a relatively simple model of an elastic lid over a maxwell viscoelastic half space. This wavelength dependence for a Maxwell rheology is also evident in Chapter 5, where the relaxation time for a viscoelastic shell surrounding a magma chamber depends on the radius of the chamber and the thickness of the shell.

When looking at longer timescale processes, such as the localisation of shear at depth beneath a strike slip fault over many earthquake cycles, I find that the key control is the dependence of the viscosity on the geothermal gradient. For a purely depth dependent rheology the range of parameters consistent with continental lithologies all result in shear zones of less than 6km for continental crust less than 40km thick. A non-linear rheology with uniform properties can generate shear zones of $\sim$ 11km; narrower than that of a constant viscosity layer (18km). Simple scaling relationships are derived which can be used to predict shear zone widths for given parameter combinations, these relationships agree with the numerical results for continental rheologies to within 5%. The results of *Yamasaki et al.* (2014) indicate that a region of low viscosity beneath the rupturing fault is required to fit both pre-seismic and post-seismic rates of deformation on the North Anatolian Fault, and our results suggest that a non-linear rheology plus thermomechnical coupling could produce such a reduction in viscosity in this region. The narrowest shear zones from our model include contributions from thermomechan-

ical coupling, a non-linear rheology, and the depth dependence of viscosity to produce shear zones of 1.7 - 3.5km for a wide range of parameter choices, significantly narrower than the 17km shear zone produced by a constant viscosity layer.

I have also investigated the effect of two kinds of boundary conditions, free-slip and driving velocities, on post-seismic motions for both single and repeated ruptures. I find that for the case of a single rupture with free-slip boundary conditions, the post-seismic displacements are half that of the linear case when $n = 2$, and considerably lower for higher values of n. This is due to the very low strain-rates across the domain generating viscosities up to several orders of magnitude higher than the equivalent linear viscosity and suppressing deformation. A single rupture with driving boundary conditions produces a very different result. The driving boundary produces a constant strain rate across the fault and the resultant reduction in viscosity allows a higher maximum post-seismic displacement for higher n, up to 20% greater for $n = 5$ compared to $n = 1$, once the system has fully relaxed, and the majority of this displacement is concentrated in the viscoelastic layer in the region close to the fault. Multiple ruptures, or an earthquake cycle model, with free-slip boundary conditions show deformation patterns that vary little between $n = 1$ and $n = 5$ after 100 cycles. A zone of distributed shear develops beneath the fault, which exhibits little variation in width with n in contradiction to the results of Chapter 3. However, including far field loading reproduces the narrowing of the shear zone with n seen in Chapter 3.

In the final Chapter of this Thesis I investigated deformation measured at Santorini volcano, Greece, between 1848 and 2012. The geodetic measurements incorporated into the modelling ranged from historic bathymetric charts to InSAR and GPS measurements. The 164 years covered were generally quiescent, punctuated by relatively brief periods of unrest - the most effusive of which were the 1866-1870 eruptions. Comparisons of historical bathymetric charts and modern swath bathymetry shows the seafloor in the vicinity of the Kameni Islands has in some regions shallowed by up to $\sim$ 200m and deepened in others by up to $\sim$ 100m, which I attribute to the emplacement of

lava on the seafloor during the 1866-1870 eruption driving viscoelastic relaxation of the substrate. My model explains both the seafloor deepening over the past 140 years and the modern rates of subsidence derived from InSAR observations between 1993 and 2010.

A new phase of unrest commenced at Santorini volcano in January 2011, characterised by the onset of micro-seismic activity focussed along the Kameni line and caldera wide uplift. A joint inversion of InSAR and GPS data was performed to determine the temporal evolution of source parameters by converting both InSAR and GPS observations into a time series of sub-surface volume change. The time series of sub-surface volume change has identified two separate pressure pulses during the period of unrest. I propose that the two pulses observed are associated with episodes of magma supply to a shallow reservoir and that the resulting pressure increase within this reservoir is responsible for the VT earthquake swarms focussed along the pre-existing zone of weakness, the Kameni line. I have produced a model of the shallow magma chamber beneath Santorini volcano by considering the magma chamber as a sphere inside a viscoelastic shell (representing a ductile aureole) in order to account for the lag between the seismicity and shape of the volume change time series. The thermal modelling used to explain the relatively low viscosities required for the subsidence model also produces a localised shell of relatively low viscosity material around the chamber itself. As with the results of Chapter 2, I find that a Maxwell rheology is capable of having a wavelength dependant relaxation timescale, where the relaxation time for this model depends both on the radius of the magma chamber and the thickness of the deforming shell.

In this Thesis I have demonstrated that:

- analytical solutions exist for simple shear of a viscoelastic media with depth dependent and non-linear rheologies.
- deformation associated with a Maxwell rheology does not necessarily have to decay with one characteristic Maxwell time η/μ, but may exhibit a timescale

dependent on the wavelength of the imposed deformation or stress field. It can even exhibit relaxation over multiple timescales if the deformation or stress field can be decomposed as a Fourier series.

- the localisation of shear at depth in a viscous shear zone arises directly from the dependence of the viscosity on the geothermal gradient, with secondary contributions from non-linear rheologies and thermo-mechanical coupling.

- simple scaling relationships can be used to predict the width of a shear zone for any given crustal thickness and lithological properties, and all known lithological properties commensurate with continental rocks produce narrow zones of shear.

- post-seismic deformation driven by a single rupture in an elastic lid above a viscoelastic half space with non-linear constitutive relation only produces displacements similar to those observed with inclusion of far field loading conditions.

- a full earthquake cycle model with non-linear rheology reproduces the viscosity structures observed in the time averaged earthquake cycle model implemented to investigate viscous shear zones.

- a model incorporating diffusion and dislocation creep mechanisms which would have a consistent background viscosity, but exhibit marked reductions in viscosities and thus higher rates of post-seismic deformation in the immediate vicinity of the fault following a co-seismic rupture.

- subsidence between 1848 and 2010 at Santorini Volcano, Greece, is a result of the eruptions of 1866-1870 driving viscoelastic relaxation of the lower crust with elastic lid thickness 1km and effective viscosity 6.3×10^{15} Pa s.

- a model for the viscoelastic relaxation of the thermal aureole surrounding a magma chamber during, and subsequent to, a period of volume change is proposed which explains the period of unrest observed at Santorini between 2011 and 2012 with a best fitting viscosity for the shell of 4.7×10^{15} Pa s.

- I propose that two pulses inferred from geodetic observations are associated with episodes of magma supply to a shallow reservoir, with the initiation and cessation of supply correlating well with the independently observed occurrence of swarms of volcano-tectonic earthquakes.

There are questions that remain as to the importance of thermo-mechanical coupling in post-seismic deformation, in addition to dual rheologies incorporating linear and non-linear mechanisms. However, I have demonstrated that long term deformation is primarily governed by the dependence of viscosity on the geothermal gradient, and multiple relaxation timescales may be achieved with simple analytical models.

6.1 Further Work

The overarching objective is to reconcile the variance in apparent crustal strength with the timescale of observation. There is much that may contribute to this; several avenues of investigation are discussed here. Incorporating multiple creep mechanisms, dynamic recrystallisation and grain size dependence into a consistent earthquake cycle theory may explain both transient rates of deformation and the formation of long term crustal structures. Comprehensive InSAR and GPS data sets are available for the 1997 Manyi *Ryder et al.* (2007); *Yamasaki and Houseman* (2012a) and 1999 Izmit earthquakes *Yamasaki et al.* (2014); I may collaborate with Dr Tadashi Yamasaki and others in fitting these datasets, whilst ensuring my model remains consistent with laboratory experiments.

Investigation of the effect of thermo-mechanical coupling on post-seismic deformation will require the incorporation of a finite element heat equation solver into *oregano_ve*. In addition, the ability to have multiple creep mechanisms operating simultaneously will allow me to investigate the interplay between linear and non-linear physical mechanisms across the whole earthquake cycle, potentially explaining the rapid post-seismic transients and the relatively well understood inter-seismic deformations.

Secondly, adding the ability to model normal loads and centres of dilatation to *oregano_ve* will allow me to investigate more general loading and volcano problems with complex rheologies. I may also extend my theoretical formulation for viscoelastic relaxation of a magma chamber thermal aureole to incorporate temperature-dependence.

Finally, *oregano_ve* is only capable of implementing periodic conditions in the z-direction, which strongly limits the effectiveness of the parallelisation routines as the problem cannot be efficiently sliced along a long axis for distribution to the computational nodes. Periodicity in x and y would require a more comprehensive rewrite of the matrix assembly routines, but would then allow us to evaluate problems such as an infinite array of infinitely long strike slip faults, as well as more efficient calculations.

Ultimately we need to work towards constructing a consistent theoretical framework encompassing earthquake cycle problems, volcanic deformation, and normal loading to explain the variance of crustal strength with timescale of observation. Understanding how and why the strength of the continents varies on different timescales will facilitate better earthquake and volcanic hazard assessment and the potential impact on human populations.

Bibliography

Barbot, S., and Y. Fialko (2010), A unified continuum representation of post-seismic relaxation mechanisms: semi-analytic models of afterslip, poroelastic rebound and viscoelastic flow, *Geophysical Journal International*, *182*(3), 1124–1140, doi:10.1111/j.1365-246X.2010.04678.x.

Barbot, S., Y. Hamiel, and Y. Fialko (2008), Space geodetic investigation of the coseismic and postseismic deformation due to the 2003 M w 7.2 Altai earthquake: Implications for the local lithospheric rheology, *Journal of Geophysical Research*, *113*(B3), B03,403, doi:10.1029/2007JB005063.

Bell, K. L. C., S. N. Carey, P. Nomikou, H. Sigurdsson, and D. Sakellariou (2013), Submarine evidence of a debris avalanche deposit on the eastern slope of Santorini volcano, Greece, *Tectonophysics*, *597-598*, 147–160, doi:10.1016/j.tecto.2012.05.006.

Berardino, P., and G. Fornaro (2002), A new algorithm for surface deformation monitoring based on small baseline differential SAR interferograms, *Geoscience and Remote Sensing*, *40*(11), 2375–2383.

Biegel, R. L., C. G. Sammis, and J. H. Dieterich (1989), The frictional properties of a simulated gouge having a fractal particle distribution, *Journal of Structural Geology*, *11*(7), 827–846, doi:10.1016/0191-8141(89)90101-6.

Biggs, J., Z. Lu, T. Fournier, and J. T. Freymueller (2010), Magma flux at Okmok Volcano, Alaska, from a joint inversion of continuous GPS, campaign GPS, and interferometric synthetic aperture radar, *Journal of Geophysical Research*, *115*(B12), B12,401, doi:10.1029/2010JB007577.

Blenkinsop, T. G. (1991), Cataclasis and processes of particle size reduction, *Pure and Applied Geophysics PAGEOPH*, *136*(1), 59–86, doi:10.1007/BF00878888.

Bonafede, M. (1986), Displacement and stress fields produced by a centre of dilation and by a pressure source in a viscoelastic half-space: application to the study of ground, *Geophysical Journal International*.

Brine, L. (1866), Documents relating to the Recent Volcanic Disturbances in the Neighbourhood of Santorino, *Quarterly Journal of the Geological Society*, pp. 318–320.

Briole, P., D. Massonnet, and C. Delacourt (1997), Post-eruptive deformation associated with the 1986-87 and 1989 lava flows of Etna detected by radar interferometry, *Geophysical Research Letters*, *24*(1), 37–40, doi:10.1029/96GL03705.

Brotchie, J., and R. Silvester (1969), On Crustal Flexure, *Journal of Geophysical Research*, *74*(22).

Bürgmann, R., and G. Dresen (2008), Rheology of the Lower Crust and Upper Mantle: Evidence from Rock Mechanics, Geodesy, and Field Observations, *Annual Review of Earth and Planetary Sciences*, *36*(1), 531–567, doi:10.1146/annurev.earth.36.031207.124326.

Carslaw, and Jaeger (1959), *Conduction of Heat in Solids*.

Cochran, J. (1979), An Analysis of Isostasy in the World's Oceans 2. Midocean ridge crests, *Journal of Geophysical Research: Solid Earth*, *84*.

Collier, J. S., and a. B. Watts (2001), Lithospheric response to volcanic loading by the Canary Islands: constraints from seismic reflection data in their flexural moat, *Geophysical Journal International*, *147*(3), 660–676, doi:10.1046/j.0956-540x.2001.01506.x.

Dragoni, M., and C. Magnanensi (1989), Displacement and stress produced by a pressurized, spherical magma chamber, surrounded by a viscoelastic shell, *Physics of the Earth and Planetary Interiors*, *56*(3-4), 316–328, doi:10.1016/0031-9201(89)90166-0.

Druitt, T., and R. Mellors (1989), Explosive volcanism on Santorini, Greece, *Geological Magazine*, *126*(2), 95–126.

Duretz, T., S. M. Schmalholz, Y. Y. Podladchikov, and D. a. Yuen (2014), Physics-controlled thickness of shear zones caused by viscous heating: implications for crustal shear localization, *Geophysical Research Letters*, pp. n/a–n/a, doi:10.1002/2014GL060438.

Ebmeier, S., J. Biggs, T. Mather, J. Elliott, G. Wadge, and F. Amelung (2012), Measuring large topographic change with InSAR: Lava thicknesses, extrusion rate and subsidence rate at Santiaguito volcano, Guatemala, *Earth and Planetary Science Letters*, *335-336*, 216–225, doi:10.1016/j.epsl.2012.04.027.

England, P., G. Houseman, and L. Sonder (1985), Length Scales for Continental Deformation in Convergent, Divergent, and Strike-Slip Environments: Analytical and Approximate Solutions for a Thin Viscous Sheet Model, *Journal of Geophysical Research*, *90*(7), 3551–3557.

England, P. C., R. T. Walker, B. Fu, and M. a. Floyd (2013), A bound on the viscosity of the Tibetan crust from the horizontality of palaeolake shorelines, *Earth and Planetary Science Letters*, *375*, 44–56, doi:10.1016/j.epsl.2013.05.001.

Foumelis, M., E. Trasatti, E. Papageorgiou, S. Stramondo, and I. Parcharidis (2013), Monitoring Santorini volcano (Greece) breathing from space, *Geophysical Journal International*, *193*(1), 161–170, doi:10.1093/gji/ggs135.

Fouqué, F. A. (1879), Santorin et ses Eruptions, *Libraire de L'Académie de Médecine, Paris*, p. 440.

Fournier, T. J., M. E. Pritchard, and S. N. Riddick (2010), Duration, magnitude, and frequency of subaerial volcano deformation events: New results from Latin America using InSAR and a global synthesis, *Geochemistry, Geophysics, Geosystems*, *11*(1), n/a–n/a, doi:10.1029/2009GC002558.

Gasparini, P., M. S. M. Mantovani, and R. Scandone (1981), A thermal model of the magma reservoir feeding Plinian eruptions at Mt. Vesuvius (Italy), *Bulletin Volcanologique*, *44*(3), 317–326, doi:10.1007/BF02600567.

Goodwin, L. B., and H.-R. Wenk (1995), Development of phyllonite from granodiorite: Mechanisms of grain-size reduction in the Santa Rosa mylonite zone, California, *Journal of Structural Geology*, *17*(5), 689–707, doi:10.1016/0191-8141(94)00093-F.

Gueydan, F., and J. Précigout (2014), Modes of continental rifting as a function of ductile strain localization in the lithospheric mantle, *Tectonophysics*, *612-613*, 18–25, doi:10.1016/j.tecto.2013.11.029.

Henderson, P., and G. M. Henderson (2009), *The Cambridge Handbook of Earth Science Data*, Cambridge University Press.

Hetland, E. A., and B. H. Hager (2005), Postseismic and interseismic displacements near a strike-slip fault: A two-dimensional theory for general linear viscoelastic rheologies, *Journal of Geophysical Research*, *110*(B10), B10,401, doi:10.1029/2005JB003689.

Hetland, E. A., and B. H. Hager (2006), Interseismic strain accumulation: Spin-up, cycle invariance, and irregular rupture sequences, *Geochemistry, Geophysics, Geosystems*, *7*(5), n/a–n/a, doi:10.1029/2005GC001087.

Hickey, J., J. Gottsmann, and R. del Potro (2013), The large-scale surface uplift in the Altiplano-Puna region of Bolivia: A parametric study of source characteristics and crustal rheology using finite element analysis, *Geochemistry, Geophysics, Geosystems*, *14*(3), 540–555, doi:10.1002/ggge.20057.

Hirth, G., and D. Kohlstedt (2003), Rheology of the Upper Mantle and the Mantle Wedge : A view from the experimentalists, *Inside the subduction Factory*, (4), 1–24.

Hirth, G., C. Teyssier, and J. Dunlap (2001), An evaluation of quartzite flow laws based on comparisons between experimentally and naturally deformed rocks, *International Journal of Earth Sciences*, *90*(1), 77–87, doi:10.1007/s005310000152.

Lagios, E., and I. Parcharidis (2005), Ground deformation monitoring of the Santorini volcano using satellite radar interferometry, *Recent Advances in Space Technologies*.

Leycester, E. (1850), Some account of the volcanic group of santorin or Thera, once called Calliste, or the Most Beautiful, *Journal of the Royal Geographical Society of London*, *20*(1850), 1–38.

Marone, C., and C. Scholz (1989), Particle-size distribution and microstructures within simulated fault gouge, *Journal of Structural Geology*, *11*(7), 799–814, doi:10.1016/0191-8141(89)90099-0.

Masterlark, T., M. Haney, H. Dickinson, T. Fournier, and C. Searcy (2010), Rheologic and structural controls on the deformation of Okmok volcano, Alaska: FEMs, InSAR, and ambient noise tomography, *Journal of Geophysical Research*, *115*(B2), B02,409, doi:10.1029/2009JB006324.

McTigue, D. F. (1987), Elastic stress and deformation near a finite spherical magma body: Resolution of the point source paradox, *Journal of Geophysical Research*, *92*(B12), 12,931, doi:10.1029/JB092iB12p12931.

Mogi, K. (1958), Relations between the Eruptions of Various Volcanoes and the Deformations of the Ground Surfaces around them.

Moore, J., and D. Clague (1989), Prodigious Submarine Landslides on the Hawaiian Ridge, *Journal of Geophysical Research: Solid Earth*, *94*(89).

Moore, J. D. P., and B. Parsons (2015), Scaling of viscous shear zones with depth dependent viscosity and power law stress-strain rate dependence, *Geophysical Journal International*.

Nakiboglu, S., and K. Lambeck (1982), A study of the Earth's response to surface loading with application to Lake Bonneville, *Geophysical Journal International*.

Newman, A. V., T. Dixon, G. Ofoegbu, and J. Dixon (2001), Geodetic and seismic constraints on recent activity at Long Valley Caldera, California: evidence for viscoelastic rheology, *Journal of Volcanology and Geothermal Research*, *105*(3), 183–206, doi:10.1016/S0377-0273(00)00255-9.

Newman, A. V., S. Stiros, L. Feng, P. Psimoulis, F. Moschas, V. Saltogianni, Y. Jiang, C. Papazachos, D. Panagiotopoulos, E. Karagianni, and D. Vamvakaris (2012), Recent geodetic unrest at Santorini Caldera, Greece, *Geophysical Research Letters*, *39*(6), n/a–n/a, doi:10.1029/2012GL051286.

Nicholls, I. (1971), Calcareous Inclusions in Lavas and Agglomerates of Santorini Volcano, *Contributions to Mineralogy and Petrology*, *276*, 261–276.

Nomikou, P., S. Carey, D. Papanikolaou, K. Croff Bell, D. Sakellariou, M. Alexandri, and K. Bejelou (2012), Submarine volcanoes of the Kolumbo volcanic zone NE of Santorini Caldera, Greece, *Global and Planetary Change*, *90-91*, 135–151, doi:10.1016/j.gloplacha.2012.01.001.

Nomikou, P., D. Papanikolaou, M. Alexandri, D. Sakellariou, and G. Rousakis (2013), Submarine volcanoes along the Aegean volcanic arc, *Tectonophysics*, *597-598*, 123–146, doi:10.1016/j.tecto.2012.10.001.

Nomikou, P., M. Parks, D. Papanikolaou, D. Pyle, T. Mather, S. Carey, a.B. Watts, M. Paulatto, M. Kalnins, I. Livanos, K. Bejelou, E. Simou, and I. Perros (2014), The emergence and growth of a submarine volcano: The Kameni islands, Santorini (Greece), *GeoResJ*, *1-2*, 8–18, doi:10.1016/j.grj.2014.02.002.

Norris, R. J., and A. F. Cooper (2003), Very high strains recorded in mylonites along the Alpine Fault, New Zealand: implications for the deep structure of plate boundary faults, *Journal of Structural Geology*, *25*(12), 2141–2157, doi:10.1016/S0191-8141(03)00045-2.

Nur, A., and G. Mavko (1974), Postseismic Viscoelastic Rebound, *Science*, pp. 0–3.

Ofeigsson, B. G., A. Hooper, F. Sigmundsson, E. Sturkell, and R. Grapenthin (2011), Deep magma storage at Hekla volcano, Iceland, revealed by InSAR time series analysis, *Journal of Geophysical Research*, *116*(B5), B05,401, doi:10.1029/2010JB007576.

Papageorgiou, E., and A. Tzanis (2010), DGPS and magnetotelluric constraints on the contemporary tectonics of the Santorini volcanic complex, Greece, *Bulletin of the Geological Society of Greece*.

Parks, M., J. D. P. Moore, P. Xanthos, J. Biggs, T. Mather, D. Pyle, C. Raptakis, and Parad (2014), From quiescence to unrest 20 years of satellite geodetic measurements at Santorini volcano, Greece, *Journal of Geophysical Research: Solid Earth*, *in review*.

Parks, M. M., J. Biggs, P. England, T. a. Mather, P. Nomikou, K. Palamartchouk, X. Papanikolaou, D. Paradissis, B. Parsons, D. M. Pyle, C. Raptakis, and V. Zacharis (2012), Evolution of Santorini Volcano dominated by episodic and rapid fluxes of melt from depth, *Nature Geoscience*, *5*(10), 749–754, doi:10.1038/ngeo1562.

Parks, M. M., S. Caliro, G. Chiodini, D. M. Pyle, T. a. Mather, K. Berlo, M. Edmonds, J. Biggs, P. Nomikou, and C. Raptakis (2013), Distinguishing contributions to diffuse CO2 emissions in volcanic areas from magmatic degassing and thermal decarbonation using soil gas 222Rnδ13C systematics: Application to Santorini volcano, Greece, *Earth and Planetary Science Letters*, *377-378*, 180–190, doi:10.1016/j.epsl.2013.06.046.

Patrick, M. R. (2004), Numerical modeling of lava flow cooling applied to the 1997 Okmok eruption: Approach and analysis, *Journal of Geophysical Research*, *109*(B3), B03,202, doi:10.1029/2003JB002537.

Poirier, J. (1980), Shear localization and shear instability in materials in the ductile field, *Journal of Structural Geology*, *2*(1-2), 135–142, doi:10.1016/0191-8141(80)90043-7.

Pollitz, F. F. (2005), Transient rheology of the upper mantle beneath central Alaska inferred from the crustal velocity field following the 2002 Denali earthquake, *Journal of Geophysical Research*, *110*(B8), B08,407, doi:10.1029/2005JB003672.

Pyle, D. M., and J. R. Elliott (2006), Quantitative morphology, recent evolution, and future activity of the Kameni Islands volcano, Santorini, Greece, *Geosphere*, *2*(5), 253, doi:10.1130/GES00028.1.

Ray, S. K. (1999), Transformation of cataclastically deformed rocks to pseudotachylyte by pervasion of frictional melt: inferences from clast-size analysis, *Tectonophysics*, *301*(3-4), 283–304, doi:10.1016/S0040-1951(98)00229-7.

Reid, H. (1910), The Mechanics of the Earthquake, The California Earthquake of April 18, 1906,, *Carnegie institution of Washington*.

Rutter, E. (1999), On the relationship between the formation of shear zones and the form of the flow law for rocks undergoing dynamic recrystallization, *Tectonophysics*, *303*(1-4), 147–158, doi:10.1016/S0040-1951(98)00261-3.

Ryder, I., B. Parsons, T. J. Wright, and G. J. Funning (2007), Post-seismic motion following the 1997 Manyi (Tibet) earthquake: InSAR observations and modelling, *Geophysical Journal International*, *169*(3), 1009–1027, doi:10.1111/j.1365-246X.2006.03312.x.

Saltogianni, V., and S. C. Stiros (2012), Modeling the Mogi magma source centre of the Santorini (Thera) volcano, Aegean Sea, Greece, 19941999, based on a numerical-topological approach, *Studia Geophysica et Geodaetica*, *56*(4), 1037–1062, doi:10.1007/s11200-012-0408-z.

Sammis, C. G., R. H. Osborne, J. L. Anderson, M. Banerdt, and P. White (1986), Self-similar cataclasis in the formation of fault gouge, *Pure and Applied Geophysics*, *124*(1-2), 53–78, doi:10.1007/BF00875719.

Savage, J., and R. Burford (1973), Geodetic Determination of Relative Plate Motion in Central California, *Journal of Geophysical Research*, *78*(5), 832–845.

Savage, J. C., and W. H. Prescott (1978), Asthenosphere readjustment and the earthquake cycle, *Journal of Geophysical Research*, *83*(B7), 3369, doi:10.1029/JB083iB07p03369.

Savage, J. C., W. H. Prescott, U. S. G. Survey, and M. Park (1978), Asthenosphere Readjustment and the Earthquake Cycle, *Journal of Geophysical Research*, *83*(B7), 3369.

Schmidt, J. S. J. (1868), On the eruption of the Kaimeni of Santorin, doi:10.1144/GSL.JGS.1868.024.01-02.47.

Scholz, C. (1998), Earthquakes and friction laws, *Nature*, pp. 37–42.

Segall, P. (2010), *Earthquake and Volcano Deformation*, Princeton University Press.

Stevens, N., and G. Wadge (2001), Surface movements of emplaced lava flows measured by synthetic aperture radar interferometry, *Journal of Geophysical Research: Solid Earth*, *106*.

Stiros, S. C., P. Psimoulis, G. Vougioukalakis, and M. Fyticas (2010), Geodetic evidence and modeling of a slow, small-scale inflation episode in the Thera (Santorini) volcano caldera, Aegean Sea, *Tectonophysics*, *494*(3-4), 180–190, doi:10.1016/j.tecto.2010.09.015.

Takeuchi, C. S., and Y. Fialko (2012), Dynamic models of interseismic deformation and stress transfer from plate motion to continental transform faults, *Journal of Geophysical Research*, *117*(B5), 1–16, doi:10.1029/2011JB009056.

Takeuchi, C. S., and Y. Fialko (2013), On the effects of thermally weakened ductile shear zones on postseismic deformation, *Journal of Geophysical Research: Solid Earth*, *118*(12), 6295–6310, doi:10.1002/2013JB010215.

Tassi, F., O. Vaselli, C. B. Papazachos, L. Giannini, G. Chiodini, G. E. Vougioukalakis, E. Karagianni, D. Vamvakaris, and D. Panagiotopoulos (2013), Geochemical and isotopic changes in the fumarolic and submerged gas discharges during the 20112012 unrest at Santorini caldera (Greece), *Bulletin of Volcanology*, *75*(4), 711, doi:10.1007/s00445-013-0711-8.

Thatcher, W., and P. England (1998), Ductile shear zones beneath strikeslip faults: Implications for the thermomechanics of the San Andreas fault zone, *Journal of Geophysical Research*, *103*, 891–905.

Washington, H. (1926), Santorini eruption of 1925, *Geological Society of America Bulletin*, *16*(349), 9–10.

Watts, A., and E. Burov (2003), Lithospheric strength and its relationship to the elastic and seismogenic layer thickness, *Earth and Planetary Science Letters*, *213*(1-2), 113–131, doi:10.1016/S0012-821X(03)00289-9.

Watts, A., and D. Masson (1995), A giant landslide on the north flank of Tenerife, Canary Islands, *Journal of Geophysical Research: Solid Earth*, *100*(95).

Watts, A., and S. Zhong (2000), Observations of fexure and the rheology of oceanic lithosphere, *Geophysical Journal International*.

Watts, A., S. Zhong, and J. Hunter (2013), The Behavior of the Lithosphere on Seismic to Geologic Timescales, *Annual Review of Earth and Planetary Sciences*, *41*(1), 443–468, doi:10.1146/annurev-earth-042711-105457.

Watts, A. B., C. Peirce, I. Grevemeyer, M. Paulatto, W. Stratford, D. Bassett, J. a. Hunter, L. M. Kalnins, and C. E. J. de Ronde (2012), Rapid rates of growth and collapse of Monowai submarine volcano in the Kermadec Arc, *Nature Geoscience*, *5*(7), 510–515, doi:10.1038/ngeo1473.

Watts, A. B., P. Nomikou, J. D. P. Moore, M. Parks, and A. M (2015), Historical bathymetric charts and the evolution of Santorini submarine volcano, Greece, *G-cubed*.

White, S., S. Burrows, and J. Carreras (1980), On mylonites in ductile shear zones, *Journal of Structural Geology*, *2*(1).

Wilson, C. K., C. H. Jones, P. Molnar, A. F. Sheehan, and O. S. Boyd (2004), Distributed deformation in the lower crust and upper mantle beneath a continental strike-slip fault zone: Marlborough fault system, South Island, New Zealand, *Geology*, *32*(10), 837, doi:10.1130/G20657.1.

Wolfram Research, I. (2012), *Mathematica*, version 9. ed., Wolfram Research, Inc., Champaign, Illinois.

Yamasaki, T., and G. a. Houseman (2012a), The crustal viscosity gradient measured from post-seismic deformation: A case study of the 1997 Manyi (Tibet) earthquake, *Earth and Planetary Science Letters*, *351-352*, 105–114, doi:10.1016/j.epsl.2012.07.030.

Yamasaki, T., and G. a. Houseman (2012b), The signature of depth-dependent viscosity structure in post-seismic deformation, *Geophysical Journal International*, *190*(2), 769–784, doi:10.1111/j.1365-246X.2012.05534.x.

Yamasaki, T., T. Wright, and G. Houseman (2014), Weak ductile shear zone beneath a major strike-slip fault: inferences from earthquake cycle model constrained by geodetic observations of the western North Anatolian Fault Zone, *Journal of Geophysical Research*, (0), doi:10.1002/2013JB010347.

Yuen, D., L. Fleitout, G. Schubert, and C. Froidevaux (1978), Shear deformation zones along major transform faults and subducting slabs, *Geophysical Journal of the Royal Astronomical Society*.

Appendix A

oregano_ve code

A.1 parconjgr.f

```
C=======================================================================
C
      SUBROUTINE ADDMATVEC3(RES,SU,SL,ISU,ISL,NPROCS,RANK,NDFREE,NSEND,
     &                  PBC)
C     ISU and ISL are index arrays indicating the entries in the
C     RES vector that correspond to degrees of freedom at the
C     processor boundaries. Contain displacement and pressure parts

      implicit double precision (a-h,o-z)
      INCLUDE 'mpif.h'
      DOUBLE PRECISION SU,SL,RES
      DIMENSION SU(NSEND)
      DIMENSION SL(NSEND)
      DIMENSION ISU(NSEND)
      DIMENSION ISL(NSEND)
      DIMENSION RES(NDFREE)

C     Added flag for periodic boundary conditions, PBC
      INTEGER RANK,ERCODE,RQUEST2,RQUEST3,PBC
      INTEGER STAT1(MPI_STATUS_SIZE)
      INTEGER STAT2(MPI_STATUS_SIZE)
      INTEGER STAT3(MPI_STATUS_SIZE)
      INTEGER STAT4(MPI_STATUS_SIZE)
C=============================================================JAMES EDIT
    ↪ FOR PERIODIC BCS
C     INITIALISE SU and SL
C        store contributions to SU from this processor's
C        boundary nodes. Top surface.
         DO I=1,NSEND
            SU(I)=RES(ISU(I))
         ENDDO
```

```
31       IF (RANK.NE.NPROCS) THEN !if not top processor
32  C       send values of RES to processor above
33          CALL MPI_ISEND(SU,NSEND,MPI_DOUBLE_PRECISION,
34       .          RANK,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
35       ENDIF
36       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
             ↪ check flag matches pbc in z, 3
37          IF (RANK.EQ.NPROCS) THEN !if top processor
38  C          send values of RES to bottom processor
39          CALL MPI_ISEND(SU,NSEND,MPI_DOUBLE_PRECISION,
40       .          0,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
41          ENDIF
42       ENDIF
43
44       IF (RANK.NE.1) THEN !if not bottom processor
45  C       receive values of RES from processor below
46          CALL MPI_RECV(SL,NSEND,MPI_DOUBLE_PRECISION,
47       .          RANK-2,0,MPI_COMM_WORLD,STAT1,ERCODE)
48
49  C       add received values from processor below to
50  C       the RES vector
51          DO I=1,NSEND
52             RES(ISL(I))=RES(ISL(I))+SL(I)
53          ENDDO
54       ENDIF
55       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
             ↪ check flag matches pbc in z, 3
56          IF (RANK.EQ.1) THEN !if bottom processor
57  C          receive values of RES from top processor
58          CALL MPI_RECV(SL,NSEND,MPI_DOUBLE_PRECISION,
59       .          NPROCS-1,0,MPI_COMM_WORLD,STAT1,ERCODE)
60  c          add received values from processor below to
61  C          the RES vector
62             DO I=1,NSEND
63                RES(ISL(I))=RES(ISL(I))+SL(I)
64             ENDDO
65          ENDIF
66
67       ENDIF
68       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
69          CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
70       ENDIF
71       IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN
72          IF(RANK.NE.NPROCS)CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
73       ENDIF
74
75  C-----------------------------------------------------------------------
76  C       store contributions to RES from this processor's
```

```
77 C        boundary nodes. Bottom surface
78       DO I=1,NSEND
79          SL(I)=RES(ISL(I))
80       ENDDO
81       IF (RANK.NE.1) THEN !if not bottom processor
82 C        send values of RES to processor below
83          CALL MPI_ISEND(SL,NSEND,MPI_DOUBLE_PRECISION,
84      .         RANK-2,1,MPI_COMM_WORLD,RQUEST2,ERCODE)
85       END IF
86       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
             ↪ check flag matches pbc in z, 3
87          IF (RANK.EQ.1) THEN !if bottom processor
88 C           send values of RES to top processor
89          CALL MPI_ISEND(SL,NSEND,MPI_DOUBLE_PRECISION,
90      .         NPROCS-1,1,MPI_COMM_WORLD,RQUEST2,ERCODE)
91          ENDIF
92       ENDIF
93       IF (RANK.NE.NPROCS) THEN !if not top processor
94 C        receive values of RES from processor above
95          CALL MPI_RECV(SU,NSEND,MPI_DOUBLE_PRECISION,
96      .        RANK,1,MPI_COMM_WORLD,STAT4,ERCODE)
97 C        set RES vector entries to values received
98 C        from processor above
99          DO I=1,NSEND
100            RES(ISU(I))=SU(I)
101         ENDDO
102      END IF
103      IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
             ↪ check flag matches pbc in z, 3
104         IF (RANK.EQ.NPROCS) THEN !if top processor
105 C          receive values of RES from bottom processor
106         CALL MPI_RECV(SU,NSEND,MPI_DOUBLE_PRECISION,
107     .        0,1,MPI_COMM_WORLD,STAT4,ERCODE)
108 C          set RES vector entries to values received
109 C          from processor above
110            DO I=1,NSEND
111               RES(ISU(I))=SU(I)
112            ENDDO
113         END IF
114      ENDIF
115      IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
116         CALL MPI_WAIT(RQUEST2,STAT2,ERCODE)
117      ENDIF
118      IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN
119         IF (RANK.NE.1)CALL MPI_WAIT(RQUEST2,STAT2,ERCODE)
120      ENDIF
121      RETURN
122      END
```

```
123 C=====================================================================
124       SUBROUTINE ADDPROD(D3,NPROCS,RANK,NN3,NN,NNT3,
125      &                   STK7,STK8,STK9,IK7,NK7,D1,SL789,SU789,
126      &                   KL789,KU789,D3L,D3U,PBC)
127 C     add the 2*STK*STK for boundary entries
128       implicit double precision (a-h,o-z)
129       INCLUDE 'mpif.h'
130       DOUBLE PRECISION SU789,SL789,D3,D3L,D3U,D1
131       DIMENSION SU789(19*NN)
132       DIMENSION SL789(19*NN)
133       DIMENSION D3(NN3),D1(3*NNT3)
134       DIMENSION STK7(NK7),STK8(NK7),STK9(NK7)
135       DIMENSION IK7(NK7)
136       DIMENSION KL789(19,NN),KU789(19,NN)
137       DIMENSION D3L(NN),D3U(NN)
138 
139       INTEGER RANK,ERCODE,RQUEST2,RQUEST3,PBC
140       INTEGER STAT1(MPI_STATUS_SIZE)
141       INTEGER STAT2(MPI_STATUS_SIZE)
142       INTEGER STAT3(MPI_STATUS_SIZE)
143       INTEGER STAT4(MPI_STATUS_SIZE)
144 C=====================================JAMES EDIT FOR PBC
145 C     add missing contributions to D3 from processor below
146 C        store contributions to D3 from this processor's
147 C        top surface.
148          DO I=1,NN
149             D3U(I)=D3(I+NN3-NN)
150          ENDDO
151       IF (RANK.NE.NPROCS) THEN !if not top processor
152 C        send values of D3U to processor above
153          CALL MPI_ISEND(D3U,NN,MPI_DOUBLE_PRECISION,
154      .           RANK,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
155       END IF
156 
157        IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
             ↪ check flag matches pbc in z, 3
158          IF (RANK.EQ.NPROCS) THEN !if top processor
159 C           send values of D3U to bottom processor
160          CALL MPI_ISEND(D3U,NN,MPI_DOUBLE_PRECISION,
161      .           0,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
162          END IF
163       ENDIF
164 
165 
166       IF (RANK.NE.1) THEN !if not bottom processor
167 C        receive values of RES from processor below
168          CALL MPI_RECV(D3L,NN,MPI_DOUBLE_PRECISION,
169      .          RANK-2,0,MPI_COMM_WORLD,STAT1,ERCODE)
```

```
170
171 C        add received values from processor below to
172 C        the D3 vector
173          DO I=1,NN
174             D3(I)=D3(I)+D3L(I)
175          ENDDO
176       ENDIF
177       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
178          IF (RANK.EQ.1) THEN !if bottom processor
179 C           receive values of RES from top processor
180          CALL MPI_RECV(D3L,NN,MPI_DOUBLE_PRECISION,
181      .         NPROCS-1,0,MPI_COMM_WORLD,STAT1,ERCODE)
182
183 C           add received values from top processor to
184 C           the D3 vector
185             DO I=1,NN
186                D3(I)=D3(I)+D3L(I)
187             ENDDO
188          ENDIF
189       ENDIF
190
191       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
192          CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
193       ENDIF
194       IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN !
              ↪ check flag matches pbc in z, 3
195          IF(RANK.NE.NPROCS)CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
196       ENDIF
197 C-----------------------------------------------------------------------
198 C     add 2*STKu*STKl for STK7, STK8 and STK9
199 C-----------------------------------------------------------------------
200 C     STK7
201 C     assign values to SU789
202       IF(RANK.NE.NPROCS) THEN !if not top processor
203          DO I=1,NN
204             DO J=1,19
205                SU789((I-1)*19+J)=0.0d0
206                IK789=KU789(J,I) !entry in STK vector
207                IF(IK789.GT.0)SU789((I-1)*19+J)=STK7(IK789)
208             ENDDO
209          ENDDO
210
211 C        send values of STK7 to processor above
212          CALL MPI_ISEND(SU789,NN*19,MPI_DOUBLE_PRECISION,
213      .          RANK,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
214       END IF
215
```

```
216       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
217          IF(RANK.EQ.NPROCS) THEN !if top processor
218             DO I=1,NN
219                DO J=1,19
220                   SU789((I-1)*19+J)=0.0d0
221                   IK789=KU789(J,I) !entry in STK vector
222                   IF(IK789.GT.0)SU789((I-1)*19+J)=STK7(IK789)
223                ENDDO
224             ENDDO
225
226 C           send values of STK7 to bottom processor
227          CALL MPI_ISEND(SU789,NN*19,MPI_DOUBLE_PRECISION,
228      .           0,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
229          END IF
230       ENDIF
231
232       IF (RANK.NE.1) THEN !if not bottom processor
233 C        receive values of RES from processor below
234          CALL MPI_RECV(SL789,NN*19,MPI_DOUBLE_PRECISION,
235      .           RANK-2,0,MPI_COMM_WORLD,STAT1,ERCODE)
236
237 C        multiply received values from processor below with
238 C        appropriate entries in STK7 on this processor
239          DO I=1,NN
240             DO J=1,19
241                IKL789=KL789(J,I) !entry in STK vector
242                IF(IKL789.GT.0)THEN
243                   IROW=IK7(IKL789)  !row number
244                   D3(I)=D3(I)+2.0*D1(IROW)*SL789((I-1)*19+J)*
245      :                 STK7(IKL789)
246                ENDIF
247             ENDDO
248          ENDDO
249       END IF
250       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
251          IF (RANK.EQ.1) THEN !if bottom processor
252 C           receive values of RES from top processor
253          CALL MPI_RECV(SL789,NN*19,MPI_DOUBLE_PRECISION,
254      .           NPROCS-1,0,MPI_COMM_WORLD,STAT1,ERCODE)
255
256 C           multiply received values from top processor with
257 C           appropriate entries in STK7 on this processor
258             DO I=1,NN
259                DO J=1,19
260                   IKL789=KL789(J,I) !entry in STK vector
261                   IF(IKL789.GT.0)THEN
```

```
262                     IROW=IK7(IKL789)  !row number
263                   D3(I)=D3(I)+2.0*D1(IROW)*SL789((I-1)*19+J)*
264     :                   STK7(IKL789)
265                   ENDIF
266                 ENDDO
267               ENDDO
268             END IF
269           ENDIF
270           IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
271             CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
272           ENDIF
273           IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN !
                  ↪ check flag matches pbc in z, 3
274             IF(RANK.NE.NPROCS)CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
275           ENDIF
276
277   C       STK8
278   C       assign values to SU789
279           IF(RANK.NE.NPROCS) THEN !if not top processor
280             DO I=1,NN
281               DO J=1,19
282                 SU789((I-1)*19+J)=0.0d0
283                 IK789=KU789(J,I) !entry in STK vector
284                 IF (IK789.GT.0)SU789((I-1)*19+J)=STK8(IK789)
285               ENDDO
286             ENDDO
287
288   C         send values of STK7 to processor above
289             CALL MPI_ISEND(SU789,NN*19,MPI_DOUBLE_PRECISION,
290         .           RANK,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
291           END IF
292           IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
                  ↪ check flag matches pbc in z, 3
293             IF(RANK.EQ.NPROCS) THEN !if top processor
294               DO I=1,NN
295                 DO J=1,19
296                   SU789((I-1)*19+J)=0.0d0
297                   IK789=KU789(J,I) !entry in STK vector
298                   IF (IK789.GT.0)SU789((I-1)*19+J)=STK8(IK789)
299                 ENDDO
300               ENDDO
301
302   C           send values of STK7 to bottom processor
303             CALL MPI_ISEND(SU789,NN*19,MPI_DOUBLE_PRECISION,
304         .           0,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
305             END IF
306           ENDIF
307           IF (RANK.NE.1) THEN !if not bottom processor
```

```
C        receive values of RES from processor below
         CALL MPI_RECV(SL789,NN*19,MPI_DOUBLE_PRECISION,
     .           RANK-2,0,MPI_COMM_WORLD,STAT1,ERCODE)

C        multiply received values from processor below with
C        appropriate entries in STK8 on this processor
         DO I=1,NN
            DO J=1,19
               IKL789=KL789(J,I) !entry in STK vector
               IF(IKL789.GT.0)THEN
                  IROW=IK7(IKL789)  !row number
                  D3(I)=D3(I)+2.0d0*D1(IROW+NNT3)*SL789((I-1)*19+J)*
     &                 STK8(IKL789)
               ENDIF
            ENDDO
         ENDDO
       END IF
       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
           ↪ check flag matches pbc in z, 3
         IF (RANK.EQ.1) THEN !if bottom processor
C           receive values of RES from top processor
         CALL MPI_RECV(SL789,NN*19,MPI_DOUBLE_PRECISION,
     .           NPROCS-1,0,MPI_COMM_WORLD,STAT1,ERCODE)

C           multiply received values from top processor with
C           appropriate entries in STK8 on this processor
            DO I=1,NN
               DO J=1,19
                  IKL789=KL789(J,I) !entry in STK vector
                  IF(IKL789.GT.0)THEN
                     IROW=IK7(IKL789)  !row number
                  D3(I)=D3(I)+2.0d0*D1(IROW+NNT3)*SL789((I-1)*19+J)*
     &                 STK8(IKL789)
                  ENDIF
               ENDDO
            ENDDO
         END IF
       ENDIF
       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
         CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
       ENDIF
       IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN !
           ↪ check flag matches pbc in z, 3
         IF(RANK.NE.NPROCS)CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
       ENDIF
C      STK9
C      assign values to SU789
       IF(RANK.NE.NPROCS) THEN !if not top processor
```

```
354         DO I=1,NN
355            DO J=1,19
356               SU789((I-1)*19+J)=0.0d0
357               IK789=KU789(J,I) !entry in STK vector
358               IF(IK789.GT.0)SU789((I-1)*19+J)=STK9(IK789)
359            ENDDO
360         ENDDO
361
362 C       send values of STK9 to processor above
363         CALL MPI_ISEND(SU789,NN*19,MPI_DOUBLE_PRECISION,
364      .          RANK,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
365       END IF
366       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
          ↪ check flag matches pbc in z, 3
367         IF(RANK.EQ.NPROCS) THEN !if top processor
368            DO I=1,NN
369               DO J=1,19
370                  SU789((I-1)*19+J)=0.0d0
371                  IK789=KU789(J,I) !entry in STK vector
372                  IF(IK789.GT.0)SU789((I-1)*19+J)=STK9(IK789)
373               ENDDO
374            ENDDO
375
376 C          send values of STK9 to bottom processor
377         CALL MPI_ISEND(SU789,NN*19,MPI_DOUBLE_PRECISION,
378      .          0,0,MPI_COMM_WORLD,RQUEST3,ERCODE)
379         END IF
380       ENDIF
381       IF (RANK.NE.1) THEN !if not bottom processor
382 C       receive values of RES from processor below
383         CALL MPI_RECV(SL789,NN*19,MPI_DOUBLE_PRECISION,
384      .          RANK-2,0,MPI_COMM_WORLD,STAT1,ERCODE)
385
386 C       multiply received values from processor below with
387 C       appropriate entries in STK9 on this processor
388         DO I=1,NN
389            DO J=1,19
390               IKL789=KL789(J,I) !entry in STK vector,processor below
391               IF(IKL789.GT.0)THEN
392                  IROW=IK7(IKL789)  !row number
393                  D3(I)=D3(I)+2.0d0*D1(IROW+2*NNT3)*SL789((I-1)*19+J)*
394      &                 STK9(IKL789)
395               ENDIF
396            ENDDO
397         ENDDO
398       END IF
399       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
          ↪ check flag matches pbc in z, 3
```

```
400         IF (RANK.EQ.1) THEN !if bottom processor
401 C          receive values of RES from top processor
402         CALL MPI_RECV(SL789,NN*19,MPI_DOUBLE_PRECISION,
403      .          NPROCS-1,0,MPI_COMM_WORLD,STAT1,ERCODE)
404
405 C          multiply received values from top processor with
406 C          appropriate entries in STK9 on this processor
407            DO I=1,NN
408               DO J=1,19
409                  IKL789=KL789(J,I) !entry in STK vector,processor below
410                  IF(IKL789.GT.0)THEN
411                     IROW=IK7(IKL789)  !row number
412                  D3(I)=D3(I)+2.0d0*D1(IROW+2*NNT3)*SL789((I-1)*19+J)*
413      &                 STK9(IKL789)
414                  ENDIF
415               ENDDO
416            ENDDO
417         END IF
418       ENDIF
419
420       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
421          CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
422       ENDIF
423       IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN !
              ↪ check flag matches pbc in z, 3
424          IF(RANK.NE.NPROCS)CALL MPI_WAIT(RQUEST3,STAT3,ERCODE)
425       ENDIF
426 C-----------------------------------------------------------------------
427       IF (RANK.NE.1)THEN !if not bottom processor
428 C        send first NN values of D3 to processor below
429          DO I=1,NN
430             D3L(I)=D3(I)
431          ENDDO
432
433          CALL MPI_ISEND(D3L,NN,MPI_DOUBLE_PRECISION,
434      .         RANK-2,1,MPI_COMM_WORLD,RQUEST2,ERCODE)
435       END IF
436       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
437          IF (RANK.EQ.1)THEN !if bottom processor
438 C          send first NN values of D3 to top processor
439             DO I=1,NN
440                D3L(I)=D3(I)
441             ENDDO
442
443          CALL MPI_ISEND(D3L,NN,MPI_DOUBLE_PRECISION,
444      .         NPROCS-1,1,MPI_COMM_WORLD,RQUEST2,ERCODE)
445          END IF
```

```
446       ENDIF
447       IF (RANK.NE.NPROCS)THEN !if not top processor
448  C       receive value of D3ADD from processor above
449          CALL MPI_RECV(D3U,NN,MPI_DOUBLE_PRECISION,
450       .       RANK,1,MPI_COMM_WORLD,STAT4,ERCODE)
451
452  C       set D3 equal to D3U at upper boundary entries
453          DO I=1,NN
454             D3(I+NN3-NN)=D3U(I)
455          ENDDO
456       END IF
457       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
458          IF (RANK.EQ.NPROCS)THEN !if top processor
459  C          receive value of D3ADD from bottom processor
460          CALL MPI_RECV(D3U,NN,MPI_DOUBLE_PRECISION,
461       .       0,1,MPI_COMM_WORLD,STAT4,ERCODE)
462
463  C          set D3 equal to D3U at upper boundary entries
464             DO I=1,NN
465                D3(I+NN3-NN)=D3U(I)
466             ENDDO
467          END IF
468       ENDIF
469
470       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN
471          CALL MPI_WAIT(RQUEST2,STAT2,ERCODE)
472       ENDIF
473       IF(PBC.NE.1 .AND. PBC.NE.11 .AND. PBC.NE.101 .AND. PBC.NE.111)THEN !
              ↪ check flag matches pbc in z, 3
474          IF (RANK.NE.1)CALL MPI_WAIT(RQUEST2,STAT2,ERCODE)
475       ENDIF
476       RETURN
477       END
478  C=====================================================================
479       SUBROUTINE MAKEK789(KL789,KU789,IK7,NK7,KBW,NN3,NN,NNT3,NNT)
480  C     KU789 contains the index to STK7,8 & 9
481  C     entries at the upper boundary at the local processor
482       implicit double precision (a-h,o-z)
483       DIMENSION IK7(NK7)
484       DIMENSION KL789(19,NN),KU789(19,NN)
485
486  C     initially zero KL789 and KU789
487       DO I=1,19
488          DO J=1,NN
489             KU789(I,J)=0
490             KL789(I,J)=0
491          ENDDO
```

```
492       ENDDO
493
494 C     find index for upper boundary (KU789)
495       DO I=1,NN  !loop over pressure nodes on upper 2D surface
496          INDEXI=I+NN3-NN !column number
497          ISK0=(INDEXI-1)*KBW
498          N=0
499          DO J=1,KBW  !loop over matrix entries in this column
500             ISK=IK7(ISK0+J) !row number
501             IF(ISK.GT.0)THEN !if still non-zero entries in STK7,8 and 9
502                DO K=1,NNT    !loop over velocity nodes on upper 2D surface
503                   INDEXK=K+NNT3-NNT !velocity node number
504                   IF(INDEXK.EQ.ISK)THEN !if the ISK entry is on this surface
505                      N=N+1   !one more entry in K789 found
506                      KU789(N,I)=ISK0+J !K789 contains index into STK7,8 and 9
507                      GO TO 102
508                   ENDIF
509                ENDDO
510             ENDIF
511  102        CONTINUE
512          ENDDO
513       ENDDO
514
515 C     find the corresponding index values for the lower boundary
516 C     (on the processor above)
517 C     this is used to multiply processor interface STK values in ADDPROD
518       DO I=1,NN
519          DO J=1,19
520             IU=KU789(J,I) !entry in upper boundary STK7,8 & 9 matrices
521             IF(IU.GT.0)THEN
522                IROWU=IK7(IU) !row number, upper boundary
523                IROWL=IROWU-NNT3+NNT !row number, lower boundary
524 C              find entry in STK that has column nr I and row number IROWL
525                ISK0=(I-1)*KBW
526                DO K=1,KBW    !loop over entries in column I
527                   ISK=IK7(ISK0+K)  !row number
528                   IF(ISK.LE.0)THEN
529                      GO TO 101  !no more entries in this column
530                   ENDIF
531                   IF(ISK.EQ.IROWL)THEN
532                      KL789(J,I)=ISK0+K
533                      GO TO 101
534                   ENDIF
535                ENDDO
536             ENDIF
537  101        CONTINUE
538          ENDDO
539       ENDDO
```

```
540
541       RETURN
542       END
```

A.2 conjgr3.f

```
1  C*--------------------------------------------------------------------
2  C*    Basil / Sybil:  conjgr.f 1.1  1 October 1998
3  C*
4  C*    Copyright (c) 1997 by G.A. Houseman, T.D. Barr, & L.A. Evans
5  C*    See README file for copying and redistribution conditions.
6  C*--------------------------------------------------------------------
7  C     Modified from conjgr.f
8  C     to solve 3D matrix equations
9  C     started Feb 5, 2004
10 C     Modified to solve problem in parallel
11 C     Started May 12, 2004
12 c
13       SUBROUTINE CONJGR3(X,UVP,KFIN,AC,STK1,STK2,STK3,STK4,STK5,STK6,
14      :            STK7,STK8,STK9,STK10,IK1,IK4,IK7,IK8,
15      &            NK1,NK4,NK7,NK8,KBW,D1,D3,
16      :            R,RTIL,P,WORK,NNT3,NN3,NNT,NN,NROWS,NITER,IERR,
17      :            ISU,ISL,SU,SL,D1U,D1L,D3U,D3L,ID1U,ID1L,ID3U,ID3L,
18      :            KL789,KU789,SL789,SU789,RANK,NPROCS,NSLSU,
19      .            IDISPMODEL,LUW,PBC)
20 C    conjugate gradient method used to solve matrix equation
21 C    Ax=b. X initially holds b, but is replaced by successive
22 C    versions of the solution X. UVP is the initial guess for x.
23 C    STK is stored in compressed format, row-wise, with column
24 C    indices stored in IK.
25 C    NROWS is the number of rows in complete matrix.
26 C    AC is the accuracy criterion of the convergence test.
27 C
28 C    The matrix equation Ax=b has the form:
29 C
30 C      [ K    K4   K5   K7 ] [ u ]  [a]
31 C      [ K4T  K2   K6   K8 ] [ v ]  [b]
32 C      [ K5T  K6T  K3   K9 ] [ w ] = [c] = X
33 C      [ K7T  K8T  K9T  0  ] [ p ]  [0]
34 C
35 C
36 C      [ K    K4   K5   K7 ] [ u ]  [a]
37 C      [ K4T  K2   K6   K8 ] [ v ]  [b]
38 C      [ K5T  K6T  K3   K9 ] [ w ] = [c] = X
39 C      [ K7T  K8T  K9T  K10] [ p ]  [0]
40 C
```

```
41 C     (K, K2, K3, K4, K5, K6, K7, K8, K9 represent
42 C      STK1,STK2,STK3,STK4,STK5,STK6,STK7,STK8,STK9,
43 C      STK10)
44 C     where u,v,w = the velocity components,
45 C     p = pressure
46 C
47 C    The second version of this routine uses a primitive
48 C    form of pre-conditioning, in which the aim is to rescale
49 C    the diagonal elements of the matrix to unity. The diagonal
50 C    elements are found in the first column of STK.
51       implicit double precision (a-h,o-z)
52       INCLUDE 'mpif.h'
53       DOUBLE PRECISION R,RTIL,WORK,RESID,PAPDN,ALPHA,BETA,RRP
54       DOUBLE PRECISION RESIDGLOBAL,PAPDNGLOBAL
55       DOUBLE PRECISION D1(NNT3*3),D3(NN3)
56       DOUBLE PRECISION X,P,AC2
57       DOUBLE PRECISION SU,SL,D1U,D1L,D3U,D3L,SL789,SU789
58 c     DOUBLE PRECISION RSAVE(3*NNT+NN),RTILSAVE(3*NNT+NN)
59 c     DOUBLE PRECISION PSAVE(3*NNT+NN),WORKSAVE(3*NNT+NN)
60       DOUBLE PRECISION RSAVE(NSLSU),RTILSAVE(NSLSU)
61       DOUBLE PRECISION PSAVE(NSLSU),WORKSAVE(NSLSU)
62       DIMENSION X(NROWS),UVP(NROWS)
63       DIMENSION R(NROWS),RTIL(NROWS),P(NROWS),WORK(NROWS)
64       DIMENSION STK1(NK1),STK2(NK1),STK3(NK1)
65       DIMENSION STK4(NK4),STK5(NK4),STK6(NK4)
66       DIMENSION STK7(NK7),STK8(NK7),STK9(NK7),STK10(NK8)
67       DIMENSION IK1(NK1),IK4(NK4),IK7(NK7),IK8(NK8)
68 c     DIMENSION ISU(3*NNT+NN) !index array top node degrees of freedom
69 c     DIMENSION ISL(3*NNT+NN)
70 c     DIMENSION SU(3*NNT+NN)
71 c     DIMENSION SL(3*NNT+NN)
72       DIMENSION ISU(NSLSU) !index array top node degrees of freedom
73       DIMENSION ISL(NSLSU)
74       DIMENSION SU(NSLSU)
75       DIMENSION SL(NSLSU)
76       DIMENSION D1U(3*NNT),D1L(3*NNT),D3U(NN),D3L(NN)
77       DIMENSION ID1U(3*NNT),ID1L(3*NNT),ID3U(NN),ID3L(NN)
78       DIMENSION KL789(19,NN),KU789(19,NN),SL789(NN*19),SU789(NN*19)
79       INTEGER ERCODE,RANK,PBC
80
81       IPRECON=1   !include preconditioner
82
83 C
84 C    set up initial vectors
85 C
86       AC2=10.0d0*AC*AC
87
88       DO 10 I=1,NROWS
```

```
89          WORK(I)=0.0d0
90    10    P(I)=UVP(I)
91  C     use P as temp double precision storage for UVP
92
93  C     compute STK*P=R - residual
94        CALL MATVEC3(P,R,STK1,STK2,STK3,STK4,STK5,STK6,
95       :           STK7,STK8,STK9,STK10,
96       :           IK1,IK4,IK7,IK8,
97       .           NK1,NK4,NK7,NK8,KBW,
98       :           NNT3,NN3,NROWS,IDISPMODEL)
99
100       DO I=1,NROWS
101          R(I)=X(I)-R(I)
102       ENDDO
103 c
104 c      write(27,*) 'R VECTOR '
105 c      do i=1,nnt3
106 c      write(27,127) i,r(i),r(i+nnt3),r(i+2*nnt3)
107 c127   format(i6,3(e12.5,' '))
108 c      call flush(27)
109 c      enddo
110 c      write(27,*) 'END R VECTOR '
111 c
112 c
113 C     Add contributions to matrix-vector product from
114 C     processors above and below
115
116       IF (IDISPMODEL.EQ.0) THEN
117       NSEND=3*NNT+NN !include pressure part
118       ENDIF
119       IF (IDISPMODEL.EQ.1) THEN
120       NSEND=3*NNT !do not include pressure part
121       ENDIF
122       CALL ADDMATVEC3(R,SU,SL,ISU,ISL,NPROCS,RANK,NROWS,NSEND,PBC)
123
124 c      call flushx(93)
125 c
126 C     normalise residual by condition matrix M (D1 and D3?)
127 c
128       IF(IPRECON.EQ.1)THEN
129          CALL PRECON3(RTIL,R,NROWS,1,STK1,STK2,STK3,STK7,STK8,STK9,
130       :               D1,D3,IK1,IK7,NK1,NK7,KBW,NNT3,NN3,NNT,NN,
131       :               ID1U,ID1L,ID3U,ID3L,D1U,D1L,D3U,D3L,
132       :               KL789,KU789,SL789,SU789,RANK,NPROCS,IDISPMODEL,PBC)
133
134 C        write preconditioner to files
135 C         CALL WRITEPRECON(D1,D3,NNT3,NN3,RANK)
136
```

```
137       ELSE
138          DO I=1,NROWS
139            RTIL(I)=R(I) !no preconditioner=>set RTIL TO R
140  C         RTIL is the preconditioned load residual
141          ENDDO
142       ENDIF
143
144       DO I=1,NROWS
145          P(I)=RTIL(I)  !p is now the load residual
146       ENDDO
147
148  C    zero entries corresponding to SU
149  C=======================================JAMES edit for periodic condition
         ↪ in residual
150       IF(RANK.NE.NPROCS)THEN !if not top processor
151          DO I=1,NSEND
152             RSAVE(I)=R(ISU(I))
153             RTILSAVE(I)=RTIL(ISU(I))
154             R(ISU(I))=0.0d0
155             RTIL(ISU(I))=0.0d0
156          ENDDO
157       ENDIF
158       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
            ↪ check flag matches pbc in z, 3
159          IF(RANK.EQ.NPROCS)THEN !if top processor
160             DO I=1,NSEND
161                RSAVE(I)=R(ISU(I))
162                RTILSAVE(I)=RTIL(ISU(I))
163                R(ISU(I))=0.0d0
164                RTIL(ISU(I))=0.0d0
165             ENDDO
166          ENDIF
167       ENDIF
168  C
169  C    calculate the norm of the residual, and set initial x
170       RESID=0.0d0
171       DO 40 I=1,NROWS
172          RESID=RESID+RTIL(I)*R(I) !RTIL=normalized residual
173     40   X(I)=UVP(I)  !set X equal to initial guess of solution
174
175  C    Subtract contributions to RESID associated with an entry in SU
176  C     IF(RANK.NE.NPROCS)THEN !if not top processor
177  C        DO I=1,NSEND
178  C           RESID=RESID-RTIL(ISU(I))*R(ISU(I))
179  C        ENDDO
180  C     ENDIF
181
182  C    reset R and RTIL values
```

```
183       IF(RANK.NE.NPROCS)THEN !if not top processor
184          DO I=1,NSEND
185             R(ISU(I))=RSAVE(I)
186             RTIL(ISU(I))=RTILSAVE(I)
187          ENDDO
188       ENDIF
189       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
190          IF(RANK.EQ.NPROCS)THEN !if top processor
191             DO I=1,NSEND
192                R(ISU(I))=RSAVE(I)
193                RTIL(ISU(I))=RTILSAVE(I)
194             ENDDO
195          ENDIF
196       ENDIF
197 C=========================================end of JAMES edit for periodic
      ↪ condition in residual
198 
199 C     find global values of RESID. Sum RESID from all processors
200       CALL MPI_ALLREDUCE(RESID,RESIDGLOBAL,1,
201      &                   MPI_DOUBLE_PRECISION,MPI_SUM,
202      &                   MPI_COMM_WORLD,ERCODE)
203 
204       RESID=RESIDGLOBAL !global residual
205 
206       IF(ABS(RESID).LT.AC2)THEN !if already converged
207          WRITE(*,*)'converged at it 0'
208          RETURN
209       END IF
210 
211 C     commence the iteration cycle
212       K=0
213    50 K=K+1
214 
215 C      CALL WRITEITEP(P,NROWS,RANK,K)
216 C
217 C     calculate alpha = (r.r)/(p.Ap)
218 C     calculate Ap=work (P is the preconditioned load residual. Is close to
      ↪ 1)
219       CALL MATVEC3(P,WORK,STK1,STK2,STK3,STK4,STK5,STK6,
220      :             STK7,STK8,STK9,STK10,
221      :             IK1,IK4,IK7,IK8,NK1,NK4,NK7,NK8,KBW,
222      :             NNT3,NN3,NROWS,IDISPMODEL)
223 
224 C     Add contributions to matrix-vector product from
225 C     processors above and below
226       CALL ADDMATVEC3(WORK,SU,SL,ISU,ISL,NPROCS,RANK,NROWS,NSEND,PBC)
227 
```

```
228
229 C     write WORK vector to file
230 C      CALL WRITEITEVEC(WORK,NROWS,RANK,K)
231 C=======================================JAMES edit for periodic condition
    ↪ in PAPDN
232       IF(RANK.NE.NPROCS)THEN !if not top processor
233          DO I=1,NSEND
234             PSAVE(I)=P(ISU(I))
235             WORKSAVE(I)=WORK(ISU(I))
236             P(ISU(I))=0.0d0
237             WORK(ISU(I))=0.0d0
238          ENDDO
239       ENDIF
240       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
          ↪ check flag matches pbc in z, 3
241          IF(RANK.EQ.NPROCS)THEN !if top processor
242             DO I=1,NSEND
243                PSAVE(I)=P(ISU(I))
244                WORKSAVE(I)=WORK(ISU(I))
245                P(ISU(I))=0.0d0
246                WORK(ISU(I))=0.0d0
247             ENDDO
248          ENDIF
249       ENDIF
250
251 C     calculate p*Ap
252       PAPDN=0.0d0
253       DO I=1,NROWS
254          PAPDN=PAPDN+P(I)*WORK(I) !P=UVP (double precision)
255       ENDDO
256
257 C     Subtract contributions to PAPDN associated with an entry in SU
258 C      IF(RANK.NE.NPROCS)THEN  !if not top processor
259 C         DO I=1,NSEND
260 C            PAPDN=PAPDN-P(ISU(I))*WORK(ISU(I))
261 C         ENDDO
262 C      ENDIF
263
264 C     find global values of PAPDN. Sum PAPDN from all processors
265       CALL MPI_ALLREDUCE(PAPDN,PAPDNGLOBAL,1,
266      &            MPI_DOUBLE_PRECISION,
267      &            MPI_SUM,MPI_COMM_WORLD,ERCODE)
268
269       PAPDN=PAPDNGLOBAL
270
271 C     reset P and WORK
272       IF(RANK.NE.NPROCS)THEN
273          DO I=1,NSEND
```

```
274            P(ISU(I))=PSAVE(I)
275            WORK(ISU(I))=WORKSAVE(I)
276         ENDDO
277       ENDIF
278       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
279         IF(RANK.EQ.NPROCS)THEN
280            DO I=1,NSEND
281               P(ISU(I))=PSAVE(I)
282               WORK(ISU(I))=WORKSAVE(I)
283            ENDDO
284         ENDIF
285       ENDIF
286 C=======================================end of JAMES edit for periodic
      ↪ condition in PAPDN
287       ALPHA=RESID/PAPDN
288
289 C     update solution vector x and residual vector r
290       DO I=1,NROWS
291          X(I)=X(I)+ALPHA*P(I)
292          R(I)=R(I)-ALPHA*WORK(I)
293       ENDDO
294
295       IF(IPRECON.EQ.1)THEN
296 C        normalise residual by condition matrix M
297          CALL PRECON3(RTIL,R,NROWS,0,STK1,STK2,STK3,STK7,STK8,STK9,
298      :                D1,D3,IK1,IK7,NK1,NK7,KBW,NNT3,NN3,NNT,NN,
299      :                ID1U,ID1L,ID3U,ID3L,D1U,D1L,D3U,D3L,
300      :                KL789,KU789,SL789,SU789,RANK,NPROCS,IDISPMODEL,PBC)
301       ELSE
302 C        if PRECON3 is not called, set RTIL TO R
303          DO I=1,NROWS
304             RTIL(I)=R(I)
305          ENDDO
306       ENDIF
307
308 C     zero SU entries of RTIL and R
309       IF(RANK.NE.NPROCS)THEN
310          DO I=1,NSEND
311             RSAVE(I)=R(ISU(I))
312             RTILSAVE(I)=RTIL(ISU(I))
313             R(ISU(I))=0.0d0
314             RTIL(ISU(I))=0.0d0
315          ENDDO
316       ENDIF
317       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
318          IF(RANK.EQ.NPROCS)THEN
```

```
319            DO I=1,NSEND
320               RSAVE(I)=R(ISU(I))
321               RTILSAVE(I)=RTIL(ISU(I))
322               R(ISU(I))=0.0d0
323               RTIL(ISU(I))=0.0d0
324            ENDDO
325         ENDIF
326       ENDIF
327
328 C     calculate beta = (r(k+1).r(k+1))/(r(k).r(k))
329       RRP=RESID
330       RESID=0.0d0
331       DO 90 I=1,NROWS
332    90 RESID=RESID+RTIL(I)*R(I)
333
334 C     Subtract contributions to RESID associated with an entry in SU
335 C      IF(RANK.NE.NPROCS)THEN !if not top processor
336 C         DO I=1,NSEND
337 C            RESID=RESID-RTIL(ISU(I))*R(ISU(I))
338 C         ENDDO
339 C      ENDIF
340
341 C     find global values of RESID. Sum RESID from all processors
342       CALL MPI_ALLREDUCE(RESID,RESIDGLOBAL,1,
343      &                   MPI_DOUBLE_PRECISION,
344      &                   MPI_SUM,MPI_COMM_WORLD,ERCODE)
345
346       RESID=RESIDGLOBAL
347
348 C     reset R and RTIL
349       IF(RANK.NE.NPROCS)THEN
350          DO I=1,NSEND
351             R(ISU(I))=RSAVE(I)
352             RTIL(ISU(I))=RTILSAVE(I)
353          ENDDO
354       ENDIF
355       IF (PBC.EQ.1 .OR. PBC.EQ.11 .OR. PBC.EQ.101 .OR. PBC.EQ.111) THEN !
              ↪ check flag matches pbc in z, 3
356          IF(RANK.EQ.NPROCS)THEN
357             DO I=1,NSEND
358                R(ISU(I))=RSAVE(I)
359                RTIL(ISU(I))=RTILSAVE(I)
360             ENDDO
361          ENDIF
362       ENDIF
363
364       BETA=RESID/RRP !new residual/old residual
365
```

```
C     if satisfactory convergence return solution
C
C     CALL DATIME(IDAY,ISEC,1)
      IF(RANK.EQ.1)THEN
         IF(MOD(K,100).EQ.0)THEN
C         IF(MOD(K,1).EQ.0)THEN
            WRITE(LUW,10002)K,RESID,RRP,BETA
            WRITE(*,10002)K,RESID,RRP,BETA
         ENDIF
      ENDIF

       IF(DABS(RESID).LT.AC2)THEN
          IF(RANK.EQ.1)THEN
             WRITE(LUW,10002)K,RESID,RRP,BETA
             WRITE(*,10002)K,RESID,RRP,BETA
          ENDIF
          KFIN=K
          RETURN
       END IF !end if converged
10002 FORMAT(' in CONJGR, k = ',I8,' r.r(k+1) = ',G12.5,
     1' r.r(k) =',G12.5,' beta = ',G12.5)
C
C    update the direction vector p
C
      DO I=1,NROWS
         P(I)=RTIL(I)+BETA*P(I)
      ENDDO
C
C     return for the next iteration unless too many
C
      IF(K.LE.NITER)GO TO 50
      WRITE(*,10001)K,BETA
      WRITE(*,10001)K,BETA
10001 FORMAT(' Convergence failed in CONJGR. K =',I8,' BETA =',
     1 G12.5)
      IERR=1
      RETURN
      END
C-----------------------------------------------------------------
C     SUBROUTINE MATVEC3 (modified version of basil's MATVEC)
C-----------------------------------------------------------------
      SUBROUTINE MATVEC3(VEC,RES,STK1,STK2,STK3,STK4,STK5,STK6,
     :                   STK7,STK8,STK9,STK10,
     :                   IK1,IK4,IK7,IK8,NK1,NK4,NK7,NK8,KBW,
     :                   NNT3,NN3,NROWS,IDISPMODEL)
C
C    this routine called by CONJGR, to post-multiply
C    velocity part of stiffness matrix
```

```
C    by vector VEC. Matrix is stored in compressed mode, row-wise
C    with column indices stored in IK.
C    NROWS=3*NNT3+NN3 is total no. of rows
C    in matrix. KBW defines the max no of non-zero elements
C    in any row of matrix.
C
      implicit double precision (a-h,o-z)
      DOUBLE PRECISION RES,VEC
      DIMENSION VEC(NROWS),RES(NROWS)
      DIMENSION STK1(NK1),STK2(NK1),STK3(NK1)
      DIMENSION STK4(NK4),STK5(NK4),STK6(NK4)
      DIMENSION STK7(NK7),STK8(NK7),STK9(NK7),STK10(NK8)
      DIMENSION IK1(NK1),IK4(NK4),IK7(NK7),IK8(NK8)
C
      DO 100 I=1,NROWS !loop over velocity degrees of freedom
  100 RES(I)=0.0d0

C     Multiply STK1 part of stiffness with appropriate part of VEC
      DO IROW1=1,NNT3 !loop over rows in STK1 matrix
         IROWR=IROW1  !row in RES
         INDEX=(IROW1-1)*KBW
         DO I=1,KBW   !loop over column entries in this matrix
            INDEX=INDEX+1
            ICOL1=IK1(INDEX)
            IROWV=ICOL1
            IF(ICOL1.LE.0)GO TO 201 !no more entries in this row

C           dot product terms derived from upper half of STK1
            RES(IROWR)=RES(IROWR)+STK1(INDEX)*VEC(IROWV)

C           dot product terms derived from lower half of STK1
            IF(ICOL1.NE.IROW1)THEN
               RES(IROWV)=RES(IROWV)+STK1(INDEX)*VEC(IROWR)
            ENDIF
         ENDDO
  201 CONTINUE
      ENDDO

C     Multiply STK2 part of stiffness with appropriate part of VEC
      DO IROW1=1,NNT3 !loop over rows in STK2 matrix
         IROWR=IROW1+NNT3 !row in RES
         INDEX=(IROW1-1)*KBW
         DO I=1,KBW   !loop over column entries in this matrix
            INDEX=INDEX+1
C            ICOL1=IK2(INDEX)
            ICOL1=IK1(INDEX)
            IROWV=ICOL1+NNT3
            IF(ICOL1.LE.0)GO TO 202 !no more entries in this row
```

```
462
463 C          dot product terms derived from upper half of STK2
464            RES(IROWR)=RES(IROWR)+STK2(INDEX)*VEC(IROWV)
465
466 C          dot product terms derived from lower half of STK2
467            IF(ICOL1.NE.IROW1)THEN
468               RES(IROWV)=RES(IROWV)+STK2(INDEX)*VEC(IROWR)
469            ENDIF
470         ENDDO
471   202 CONTINUE
472      ENDDO
473
474 C    Multiply STK3 part of stiffness with appropriate part of VEC
475      DO IROW1=1,NNT3 !loop over rows in STK3 matrix
476         IROWR=IROW1+2*NNT3 !row in RES
477         INDEX=(IROW1-1)*KBW
478         DO I=1,KBW   !loop over column entries in this matrix
479            INDEX=INDEX+1
480 C           ICOL1=IK3(INDEX)
481            ICOL1=IK1(INDEX)
482            IROWV=ICOL1+2*NNT3
483            IF(ICOL1.LE.0)GO TO 203 !no more entries in this row
484
485 C          dot product terms derived from upper half of STK3
486            RES(IROWR)=RES(IROWR)+STK3(INDEX)*VEC(IROWV)
487
488 C          dot product terms derived from lower half of STK3
489            IF(ICOL1.NE.IROW1)THEN
490               RES(IROWV)=RES(IROWV)+STK3(INDEX)*VEC(IROWR)
491            ENDIF
492         ENDDO
493   203 CONTINUE
494      ENDDO
495
496 C    Multiply STK4 part of stiffness with appropriate part of VEC
497      DO IROW1=1,NNT3 !loop over rows in STK4 matrix
498         IROWR=IROW1  !row in RES
499         INDEX=(IROW1-1)*KBW
500         DO I=1,KBW   !loop over column entries in this matrix
501            INDEX=INDEX+1
502            ICOL1=IK4(INDEX)
503            IROWV=ICOL1+NNT3
504            IF(ICOL1.LE.0)GO TO 204 !no more entries in this row
505
506 C          dot product terms derived from STK4
507            RES(IROWR)=RES(IROWR)+STK4(INDEX)*VEC(IROWV)
508
509 C          dot product terms derived from transpose of STK4
```

```
510            RES(IROWV)=RES(IROWV)+STK4(INDEX)*VEC(IROWR)
511          ENDDO
512   204 CONTINUE
513        ENDDO
514
515 C     Multiply STK5 part of stiffness with appropriate part of VEC
516        DO IROW1=1,NNT3 !loop over rows in STK5 matrix
517          IROWR=IROW1  !row in RES
518          INDEX=(IROW1-1)*KBW
519          DO I=1,KBW   !loop over column entries in this matrix
520            INDEX=INDEX+1
521            ICOL1=IK4(INDEX)
522            IROWV=ICOL1+2*NNT3
523            IF(ICOL1.LE.0)GO TO 205 !no more entries in this row
524
525 C          dot product terms derived from STK5
526            RES(IROWR)=RES(IROWR)+STK5(INDEX)*VEC(IROWV)
527
528 C          dot product terms derived from transpose of STK5
529            RES(IROWV)=RES(IROWV)+STK5(INDEX)*VEC(IROWR)
530          ENDDO
531   205 CONTINUE
532        ENDDO
533
534 C     Multiply STK6 part of stiffness with appropriate part of VEC
535        DO IROW1=1,NNT3 !loop over rows in STK6 matrix
536          IROWR=IROW1+NNT3 !row in RES
537          INDEX=(IROW1-1)*KBW
538          DO I=1,KBW   !loop over column entries in this matrix
539            INDEX=INDEX+1
540            ICOL1=IK4(INDEX)
541            IROWV=ICOL1+2*NNT3
542            IF(ICOL1.LE.0)GO TO 206 !no more entries in this row
543
544 C          dot product terms derived from STK6
545            RES(IROWR)=RES(IROWR)+STK6(INDEX)*VEC(IROWV)
546
547 C          dot product terms derived from transpose of STK6
548            RES(IROWV)=RES(IROWV)+STK6(INDEX)*VEC(IROWR)
549          ENDDO
550   206 CONTINUE
551        ENDDO
552 c
553 C     Multiply STK10 part of stiffness with appropriate part of VEC
554 c
555        IF (IDISPMODEL.EQ.0) THEN
556        DO IROW1=1,NN3 !loop over rows in STK10 matrix
557          IROWR=IROW1+3*NNT3 !row in RES
```

```
558         INDEX=(IROW1-1)*KBW
559         DO I=1,KBW   !loop over column entries in this matrix
560            INDEX=INDEX+1
561C            ICOL1=IK3(INDEX)
562            ICOL1=IK8(INDEX)
563c           IROWV=ICOL1+2*NNT3
564            IROWV=ICOL1+3*NNT3
565            IF(ICOL1.LE.0)GO TO 214 !no more entries in this row
566c
567C           dot product terms derived from upper half of STK10
568c
569            RES(IROWR)=RES(IROWR)+STK10(INDEX)*VEC(IROWV)
570c
571C           dot product terms derived from lower half of STK10
572c
573            IF(ICOL1.NE.IROW1)THEN
574               RES(IROWv)=RES(IROWv)+STK10(INDEX)*VEC(IROWR)
575            ENDIF
576         ENDDO
577  214 CONTINUE
578      ENDDO
579c
580C ---------------------------------------
581C FOR THE INCOMPRESSIBLE CASE
582C ---------------------------------------
583C    now add in the cross-component contributions from
584C     the STK7 STK8 and STK9 matrices and their transpose.
585C     for incompressible flow only. Note that
586C     STK7, STK8 and STK9 are stored columnwise,
587C     whereas other matrices
588C     are stored rowwise.
589C
590      DO 600 ICOL7=1,NN3 !loop over columns in STK7
591         INDEX=(ICOL7-1)*KBW
592C        for all entries in this column
593         DO I=1,KBW     !loop over rows
594            INDEX=INDEX+1
595            IROW7=IK7(INDEX)    !local row number
596            IF(IROW7.LE.0)GO TO 600 !if no more non-zero entries in this
                  ↪ column
597            ITRAN7=ICOL7+3*NNT3 !entry in VEC
598            IROWR=IROW7         !entry in RES
599C
600C           dot product terms derived from K7.p
601            RES(IROWR)=RES(IROWR)+STK7(INDEX)*VEC(ITRAN7)
602
603C           dot product terms derived from K7(t).u
604            RES(ITRAN7)=RES(ITRAN7)+STK7(INDEX)*VEC(IROWR)
```

```
605         ENDDO
606   600 CONTINUE
607
608       DO 608 ICOL8=1,NN3 !loop over columns in STK8
609          INDEX=(ICOL8-1)*KBW
610 C        for all entries in this column
611          DO 558 I=1,KBW
612             INDEX=INDEX+1
613             IROW8=IK7(INDEX)
614             IF(IROW8.LE.0)GO TO 608
615             ITRAN8=ICOL8+3*NNT3 !entry in VEC
616             IROWR=IROW8+NNT3    !entry in RES
617 C
618 C           dot product terms derived from K8.p
619             RES(IROWR)=RES(IROWR)+STK8(INDEX)*VEC(ITRAN8)
620 C
621 C           dot product terms derived from K8(t).u
622             RES(ITRAN8)=RES(ITRAN8)+STK8(INDEX)*VEC(IROWR)
623   558    CONTINUE
624   608 CONTINUE
625
626       DO 609 ICOL9=1,NN3 !loop over columns in STK9
627          INDEX=(ICOL9-1)*KBW
628 C        for all entries in this column
629          DO 559 I=1,KBW
630             INDEX=INDEX+1
631             IROW9=IK7(INDEX)
632             IF(IROW9.LE.0)GO TO 609
633             ITRAN9=ICOL9+3*NNT3 !entry in VEC
634             IROWR=IROW9+2*NNT3  !entry in RES
635 C
636 C           dot product terms derived from K9.p
637             RES(IROWR)=RES(IROWR)+STK9(INDEX)*VEC(ITRAN9)
638 C
639 C           dot product terms derived from K9(t).u
640             RES(ITRAN9)=RES(ITRAN9)+STK9(INDEX)*VEC(IROWR)
641   559    CONTINUE
642   609 CONTINUE
643       ENDIF
644
645       RETURN
646       END
647 C-------------------------------------------------------------
648 C     SUBROUTINE PRECON3
649 C-------------------------------------------------------------
650       SUBROUTINE PRECON3(RTIL,R,NROWS,JTIME,STK1,STK2,STK3,
651      :            STK7,STK8,STK9,D1,D3,IK1,IK7,NK1,NK7,
652      :            KBW,NNT3,NN3,NNT,NN,ID1U,ID1L,ID3U,ID3L,
```

```
653      :            D1U,D1L,D3U,D3L,KL789,KU789,SL789,SU789,
654      :            RANK,NPROCS,IDISPMODEL,PBC)
655  C     normalise residual by condition matrix M
656       implicit double precision (a-h,o-z)
657       include 'mpif.h'
658       DOUBLE PRECISION R,RTIL
659       DOUBLE PRECISION D1,D3
660       DOUBLE PRECISION D1U,D1L,D3U,D3L,SL789,SU789
661       DIMENSION D1(3*NNT3),D3(NN3)
662       DIMENSION R(NROWS),RTIL(NROWS)
663       DIMENSION STK1(NK1),STK2(NK1),STK3(NK1)
664       DIMENSION STK7(NK7),STK8(NK7),STK9(NK7)
665       DIMENSION IK1(NK1),IK7(NK7)
666       DIMENSION D1U(3*NNT),D1L(3*NNT)
667       DIMENSION D3U(NN),D3L(NN)
668       DIMENSION ID1U(3*NNT),ID1L(3*NNT)
669       DIMENSION ID3U(NN),ID3L(NN)
670       DIMENSION KL789(19,NN),KU789(19,NN)
671       DIMENSION SL789(NN*19),SU789(NN*19)
672
673       INTEGER RANK,PBC
674
675  C    if JTIME=1 the diagonal matrices are defined
676  C
677       IF(JTIME.EQ.1)THEN !before starting cg iteration loop
678         DO I1=1,NNT3    !loop over all node points
679            KC=KBW*(I1-1)+1 !first entry in stiffness array of this row
680            D1(I1)=STK1(KC)
681            D1(I1+NNT3)=STK2(KC)
682            D1(I1+2*NNT3)=STK3(KC)
683         ENDDO
684
685  C      create index arrays for D1U an D1L
686         ISHIFT=NNT3-NNT
687         DO J=0,2       !loop over directions
688            DO I=1,NNT  !loop over nodes on 2D surface
689               ID1L(I+J*NNT)=I+J*NNT3
690               ID1U(I+J*NNT)=I+J*NNT3+ISHIFT
691            ENDDO
692         ENDDO
693
694  C      add contributions from STK on adjoining processors
695  C      to appropriate entries in D1
696         NSEND=3*NNT
697         CALL ADDMATVEC3(D1,D1U,D1L,ID1U,ID1L,NPROCS,RANK,(3*NNT3),
698      :              NSEND,PBC)
699         DO I1=1,(3*NNT3)
700            IF (D1(I1).NE.0.0d0)THEN
```

```
701             D1(I1)=1.0d0/D1(I1)
702           ELSE
703             D1(I1)=1.0d0
704           ENDIF
705         ENDDO
706 C ---------------------------
707 C FOR THE INCOMPRESSIBLE CASE
708 C ---------------------------
709         IF (IDISPMODEL.EQ.0) THEN
710         DO I2=1,NN3 !loop over columns in STK7,8 & 9 (on this processor)
711 C         compute value of D3 at this pressure node (for this processor)
712           D3(I2)=0.0d0
713           INDEX=(I2-1)*KBW
714           DO I3=1,KBW
715             INDEX=INDEX+1
716             K7=IK7(INDEX)  !row number in STK matrix
717             IF(K7.EQ.0)GO TO 208
718             D3(I2)=D3(I2)+STK7(INDEX)*STK7(INDEX)*D1(K7)
719      &             +STK8(INDEX)*STK8(INDEX)*D1(K7+NNT3)
720      &             +STK9(INDEX)*STK9(INDEX)*D1(K7+2*NNT3)
721           ENDDO
722   208     CONTINUE
723         ENDDO
724
725 C       create index arrays for D3U and D3L
726 C       pressure part
727         ISHIFT=NN3-NN
728         DO I=1,NN
729           ID3L(I)=I
730           ID3U(I)=I+ISHIFT
731         ENDDO
732
733 C       add contributions to D3 from adjoining processors
734 C       to appropriate entries in D3 (STKu*STKl and 2*STKu*STKl)
735         CALL ADDPROD(D3,NPROCS,RANK,NN3,NN,NNT3,
736      &             STK7,STK8,STK9,IK7,NK7,D1,
737      &             SL789,SU789,KL789,KU789,D3L,D3U,PBC)
738
739         DO I2=1,NN3
740           IF(D3(I2).NE.0.0d0) THEN
741             D3(I2)=1.0d0/D3(I2)
742           ELSE
743             D3(I2)=1.0d0
744           END IF
745         ENDDO !end loop over pressure columns
746
747 C       Set D3 to 1 - just temporary until we find
748 C       a good preconditioner for the pressure part
```

```
749 C        DO I2=1,NN3
750 C           D3(I2)=1.0
751 C        ENDDO
752 C---------------------------
753         ENDIF  ! end if idispmodel
754       END IF !endif JTIME.EQ.1
755
756 c     normalise residual
757       DO I1=1,(3*NNT3)
758          RTIL(I1)=R(I1)*D1(I1)
759       ENDDO
760 c
761       IF (IDISPMODEL.EQ.0) THEN
762       DO I1=1,NN3
763          I2=3*NNT3+I1 !entry in stiffness
764          RTIL(I2)=R(I2)*D3(I1)
765       ENDDO
766       ENDIF
767 C
768       RETURN
769       END
```

Appendix B

Mathematica codes for viscous shear zone calculations

```
Needs["NDSolve`FEM`"];
Needs["DifferentialEquations`InterpolatingFunctionAnatomy`"];
qs = {129, 99, 89, 59, 54.4} * 4.185 * 1000;
bs = {5.2 * 10^5, 4.4 * 10^5, 1.8 * 10^2, 2.6 * 10^-3, 18};
Q = qs[[2]];
B = bs[[bob]];
mesh = ToElementMesh[FullRegion[2], {{0, 4.}, {0, 1.}},
    "MeshElementType" → TriangleElement, MeshRefinementFunction →
     Function[{vertices, area}, v2 = vertices; v2[[All, 2]] = v2[[All, 2]] - 1;
      area > 0.0001 (0.05 + 2 Norm[Mean[v2]])]];
(*mesh["Wireframe"]*)
Ts = 274.2 + 10;
βd = 20 * 30;
R = 8.314;
T = Ts + βd (y + 1 / 2);
Q = qs[[1]];
uif = NDSolveValue[{Laplacian[u[x, y], {x, y}] ==
      (Q βd / (R (Ts + βd (1 / 2 + y))^2) - βd / (Ts + βd (1 / 2 + y))) D[u[x, y], y], u[x, 0] == 1,
     Derivative[0, 1][u][x, 1] == 0, u[0, y] == 0, u[4, y] == 1}, u, {x, y} ∈ mesh];

pt = DensityPlot[uif[x, -y], {x, 0, 1}, {y, 0, -1}, PerformanceGoal → "Quality",
    PlotPoints → 200, ColorFunction → "Rainbow", PlotRange → {0, 1},
    AspectRatio → Automatic, ClippingStyle → ColorData["Rainbow"][1],
    PlotRangePadding → None, ImageSize → 800, BaseStyle → FontSize -> 24,
    PlotLegends → BarLegend[Automatic, LabelStyle → FontSize -> 24]];
Show[pt, ContourPlot[uif[x, -y], {x, 0, 1}, {y, 0, -1}, ContourShading → None,
  ContourStyle → Thick, Contours → Table[0.1 i, {i, 1, 9}],
  PlotRange → Full, PlotPoints → 200, PerformanceGoal → "Quality"]]
```

```
{539 865., 414 315., 372 465., 246 915., 227 664.}
```

```
InterpolatingFunction[ Domain : {{0., 4.}, {0., 1.}}
                       Output : scalar ]
```

```
years = 3.15569 * 10^7;
d = 30 000;
χ = 1120;
ρ = 2800;
κ = 3.36;
s1 = D[uif[x, y], x]^2 + D[uif[x, y], y]^2;
s2 = Piecewise[{{s1 /. x → Abs[x], y ≥ 0}}];
s4 = Interpolation[Table[{x, y, s2}, {x, -4, 4, 0.01}, {y, -0.5, 1, 0.01}] //
     Flatten[#, 1] &, InterpolationOrder → 1];
(*ContourPlot[Log[10,s4[x,y]],{x,-4,4},{y,-0.5,1},PlotRange→Full,
  AspectRatio→Automatic,ImageSize→1000]
 Plot[s4[0,y],{y,-0.5,1},PlotRange→Full]*)
v2 = (35 / 2) * 0.001 / years;
γ1 = κ 250 000 years / (ρ χ d^2);
γ2 = 250 000 years / (ρ χ) (v2 / d)^2;
(*ρ χ  D[w[t,x,y],t]-κ/d^2 Laplacian[w[t,x,y],{x,y}]-s4==NeumannValue[βd,y==1]*)
testloc = 0.54;
test2 = 20 + T /. y → (testloc - 0.5);
output = {};
state = First[NDSolve`ProcessEquations[
    {w^(0,0,1)[x, y, t] == γ1 (w^(0,2,0)[x, y, t] + w^(2,0,0)[x, y, t]) +
       γ2 w[x, y, t] Exp[Q / (R w[x, y, t])] s4[Abs[x], y],
     w[x, y, 0] == T, w[-2, y, t] == T, w[2, y, t] == T, w[x, -0.5, t] == Ts,
     Derivative[0, 1, 0][w][x, 1.0, t] == βd}, w, {t, 0, 2}, {x, -2, 2},
    {y, -0.5, 1}, Method → {"MethodOfLines", "SpatialDiscretization" →
       {"TensorProductGrid", (*"Coordinates"→{xgrid,ygrid}*)
        "MinPoints" → 30, "DifferenceOrder" → "Pseudospectral"}}
      (*Method→{"EventLocator","Event":→w[0,testloc,t]-test2,
       "Direction"→1,"EventLocationMethod"→"StepBegin"}}*)]]
NDSolve`Iterate[state, 2] // AbsoluteTiming
wout = NDSolve`ProcessSolutions[state]
AppendTo[output, wout]
InterpolatingFunctionCoordinates[w /. output[[-1]]];
coords = Flatten[Outer[List, %[[1]], %[[2]]], 1];
ListPlot[coords, ImageSize → 1000]
rms = 100;
i = 1;
While[rms ≥ 1.0, i++; Print[i];
  wnew = w[x, y, 2] /. wout;
  state = First[NDSolve`Reinitialize[state, w[x, y, 0] == wnew]];
  tim = AbsoluteTiming[NDSolve`Iterate[state, 2]];
  Print[tim];
  w0 = w /. wout;
  w1 = w0[#1, #2, 2] & @@@ coords;
  wout = NDSolve`ProcessSolutions[state];
  w2 = w /. wout;
  w3 = w2[#1, #2, 2] & @@@ coords;
  rms = RootMeanSquare[w3 - w1];
  Print[rms];
  AppendTo[output, wout]] // AbsoluteTiming
```

```
NDSolve`StateData[<0.>]
```

```
{66.816010, Null}
```

```
{9997.852320, Null}
```

```
(out2[[-1]] /. t → 2)
Show[DensityPlot[%, {x, 0, 1.5}, {y, 0.5, -1},
  ImageSize → 1000, AspectRatio → Automatic, PlotRange → {0, 1100},
  ColorFunction → "SolarColors", PlotRangePadding → None,
  BaseStyle → FontSize → 36, PlotLegends → Placed[BarLegend[Automatic,
     LegendLayout → "Row", LabelStyle → FontSize → 36], Bottom]],
 ContourPlot[%, {x, 0, 1.5}, {y, 0.5, -1}, ContourShading → None,
  ContourStyle → Thick, Contours → Range[50, 7000, 50]]]
```

$$-284.2 - 600\left(\frac{1}{2} - y\right) +$$

InterpolatingFunction[Domain : {{−2., 2.}, {−0.5, 1.}, {0., 2.}} Output : scalar][x, −y, 2]

Appendix C

oregano_ve job files

```
#This input file solves the visco-elastic response to the displacement of fault

OUTPUT BIN=n30_long BINDIR=FD.sols OUTDIR=FD.out
MESH NX=10, NY=20, NZ=4, TYPE=1
GEOMETRY XLEN=10.0, YLEN=20.0, ZLEN=0.5
DEFORM_GRID 1 1 0.0 10.0 2.0 4.0 8.0 6.0

#Timestep commands
STEP TEXIT=100.0, TIMWSTEPS = 10.0,20.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0 ;
#     KSAVE=100, TIMFACT = 1.0

#Gravity switched off for this test
RHOG 0.0 0.00  0.0

#Boundary conditions entries
#Other boundaries assumed to be traction free
#Side boundaries in Z-direction
ON X = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=1, UX=0.0 TO 0.0: &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON X = 10.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=2, UX=0.0 TO 0.0: &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON Y = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=3, UY=0.0 TO 0.0: &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON Y = 20.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=4, UY=0.0 TO 0.0: &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
PERIODIC Z

```

```
#Entry of split nodes on fault surface; on each line there is an entry only
    ↪ for one
#component of fault displacement, i.e., X, Y, or Z direction.
#There is also an entry for time-dependent fault movement.

FAULT X = 5.0 FOR 18.99 < Y < 20.01 FOR -20 < Z < 20 : LB=5, SZ=1.0 TO 1.0: &
    ↪
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0

#Two types of material: different constant viscosity in each, constant
    ↪ density in each
REG A : MR = 1 ;

REG P { 0.0 65.0 0.0 19.0 0.0 2.0 } : MR = 2 ;

MAT 1 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 1.0 &
        VR = 0 VC = 1.0e20 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25

MAT 2 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 3.0 &
        VR = 0 VC = 1.0 0.0 0.0 0.0 ; YOUNGS = 3.0 POISSON = 0.25

#Time-series data: to monitor displacement of midplane surface
SERIES TIME, UMIN, UMAX, VMIN, VMAX, WMIN, WMAX
SERIES UZ = 0.0, 20.0, 0.25 ; UZ = 0.5, 20.0, 0.25 ; UZ = 1.0, 20.0, 0.25 ;
SERIES UZ = 1.5, 20.0, 0.25 ; UZ = 2.0, 20.0, 0.25 ; UZ = 2.5, 20.0, 0.25 ;
SERIES UZ = 3.0, 20.0, 0.25 ; UZ = 3.5, 20.0, 0.25 ; UZ = 4.0, 20.0, 0.25 ;
SERIES UZ = 4.5, 20.0, 0.25 ; UZ = 5.0, 20.0, 0.25 ; UZ = 5.5, 20.0, 0.25 ;
SERIES UZ = 6.0, 20.0, 0.25 ; UZ = 6.5, 20.0, 0.25 ; UZ = 7.0, 20.0, 0.25 ;
SERIES UZ = 7.5, 20.0, 0.25 ; UZ = 8.0, 20.0, 0.25 ; UZ = 8.5, 20.0, 0.25 ;
SERIES UZ = 9.0, 20.0, 0.25 ; UZ = 9.5, 20.0, 0.25 ; UZ = 10.0, 20.0, 0.25 ;
```

A1_Code/n30_long

```
#This input file solves the visco-elastic response to the displacement of
    ↪ fault

OUTPUT BIN=tdbc_n30_long BINDIR=FD.sols OUTDIR=FD.out
MESH NX=10, NY=20, NZ=4, TYPE=1
GEOMETRY XLEN=10.0, YLEN=20.0, ZLEN=0.5
DEFORM_GRID 1 1 0.0 10.0 2.0 4.0 8.0 6.0

#Timestep commands
STEP TEXIT=100.0, TIMWSTEPS =
    ↪ 10.0,20.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0 ;
#     KSAVE=100, TIMFACT = 10.0

#Gravity switched off for this test
RHOG 0.0 0.00  0.0
```

```
14
15 #Boundary conditions entries
16 #Other boundaries assumed to be traction free
17 #Side boundaries in Z-direction
18 ON X = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=1, UX=0.0 TO 0.0:
       ↪ &
19   TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
20 ON X = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=1, UZ=-9000.0 TO
       ↪ -9000.0: &
21   TP=1,TDBC=2,TDRV=9000.0,0.0,0.0;
22 ON X = 10.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=2, UX=0.0 TO 0.0:
       ↪  &
23   TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
24 ON X = 10.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=2, UZ=9000.0 TO
       ↪ 9000.0: &
25   TP=1,TDBC=2,TDRV=9000.0,0.0,0.0;
26 ON Y = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=3, UY=0.0 TO 0.0:
       ↪ &
27   TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
28 ON Y = 20.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=4, UY=0.0 TO 0.0:
       ↪  &
29   TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
30 PERIODIC Z
31
32 #Entry of split nodes on fault surface; on each line there is an entry only
       ↪ for one
33 #component of fault displacement, i.e., X, Y, or Z direction.
34 #There is also an entry for time-dependent fault movement.
35
36 FAULT X = 5.0 FOR 18.99 < Y < 20.01 FOR -20 < Z < 20 : LB=5, SZ=1.0 TO 1.0:
       ↪ &
37   TP=1,TDBC=0,TDRV=0.0,0.0,0.0
38
39 #Two types of material: different constant viscosity in each, constant
       ↪ density in each
40 REG A : MR = 1 ;
41
42 REG P { 0.0 65.0 0.0 19.0 0.0 2.0 } : MR = 2 ;
43
44 MAT 1 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 1.0 &
45        VR = 0 VC = 1.0e20 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25
46
47 MAT 2 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 3.0 &
48        VR = 0 VC = 1.0 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25
49
50 #Time-series data: to monitor displacement of midplane surface
51 SERIES TIME, UMIN, UMAX, VMIN, VMAX, WMIN, WMAX
52 SERIES UZ = 0.0, 20.0, 0.25 ; UZ = 0.5, 20.0, 0.25 ; UZ = 1.0, 20.0, 0.25 ;
```

```
SERIES UZ = 1.5, 20.0, 0.25 ; UZ = 2.0, 20.0, 0.25 ; UZ = 2.5, 20.0, 0.25 ;
SERIES UZ = 3.0, 20.0, 0.25 ; UZ = 3.5, 20.0, 0.25 ; UZ = 4.0, 20.0, 0.25 ;
SERIES UZ = 4.5, 20.0, 0.25 ; UZ = 5.0, 20.0, 0.25 ; UZ = 5.5, 20.0, 0.25 ;
SERIES UZ = 6.0, 20.0, 0.25 ; UZ = 6.5, 20.0, 0.25 ; UZ = 7.0, 20.0, 0.25 ;
SERIES UZ = 7.5, 20.0, 0.25 ; UZ = 8.0, 20.0, 0.25 ; UZ = 8.5, 20.0, 0.25 ;
SERIES UZ = 9.0, 20.0, 0.25 ; UZ = 9.5, 20.0, 0.25 ; UZ = 10.0, 20.0, 0.25 ;
```

A1_Code/tdbc_n30_long

```
#This input file solves the visco-elastic response to the displacement of
    ↪ fault

OUTPUT BIN=cycle_n30_long BINDIR=FD.sols OUTDIR=FD.out
MESH NX=10, NY=20, NZ=4, TYPE=1
GEOMETRY XLEN=10.0, YLEN=20.0, ZLEN=0.5
DEFORM_GRID 1 1 0.0 10.0 2.0 4.0 8.0 6.0

#Timestep commands
STEP TEXIT=100.0, TIMWSTEPS =
    ↪ 10.0,20.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0 ;
#     KSAVE=100, TIMFACT = 10.0

#Gravity switched off for this test
RHOG 0.0 0.00  0.0

#Boundary conditions entries
#Other boundaries assumed to be traction free
#Side boundaries in Z-direction
ON X = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=1, UX=0.0 TO 0.0:
    ↪ &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON X = 10.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=2, UX=0.0 TO 0.0:
    ↪  &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON Y = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=3, UY=0.0 TO 0.0:
    ↪ &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON Y = 20.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=4, UY=0.0 TO 0.0:
    ↪  &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
PERIODIC Z

#Entry of split nodes on fault surface; on each line there is an entry only
    ↪ for one
#component of fault displacement, i.e., X, Y, or Z direction.
#There is also an entry for time-dependent fault movement.

FAULT X = 5.0 FOR 18.99 < Y < 20.01 FOR -20 < Z < 20 : LB=5, SZ=1.0 TO 1.0:
```

```
    ↪ &
  TP=1,TDBC=5,TDRV=0.0,1.0,0.0

#Two types of material: different constant viscosity in each, constant
    ↪ density in each
REG A : MR = 1 ;

REG P { 0.0 65.0 0.0 19.0 0.0 2.0 } : MR = 2 ;

MAT 1 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 1.0 &
        VR = 0 VC = 1.0e20 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25

MAT 2 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 3.0 &
        VR = 0 VC = 1.0 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25

#Time-series data: to monitor displacement of midplane surface
SERIES TIME, UMIN, UMAX, VMIN, VMAX, WMIN, WMAX
SERIES UZ = 0.0, 20.0, 0.25 ; UZ = 0.5, 20.0, 0.25 ; UZ = 1.0, 20.0, 0.25 ;
SERIES UZ = 1.5, 20.0, 0.25 ; UZ = 2.0, 20.0, 0.25 ; UZ = 2.5, 20.0, 0.25 ;
SERIES UZ = 3.0, 20.0, 0.25 ; UZ = 3.5, 20.0, 0.25 ; UZ = 4.0, 20.0, 0.25 ;
SERIES UZ = 4.5, 20.0, 0.25 ; UZ = 4.7, 20.0, 0.25 ; UZ = 4.9, 20.0, 0.25 ;
SERIES UZ = 5.1, 20.0, 0.25 ; UZ = 5.3, 20.0, 0.25 ; UZ = 5.5, 20.0, 0.25 ;
SERIES UZ = 6.0, 20.0, 0.25 ; UZ = 6.5, 20.0, 0.25 ; UZ = 7.0, 20.0, 0.25 ;
SERIES UZ = 7.5, 20.0, 0.25 ; UZ = 8.0, 20.0, 0.25 ; UZ = 8.5, 20.0, 0.25 ;
SERIES UZ = 9.0, 20.0, 0.25 ; UZ = 9.5, 20.0, 0.25 ; UZ = 10.0, 20.0, 0.25 ;
```

A1_Code/cycle_n30_long

```
#This input file solves the visco-elastic response to the displacement of
    ↪ fault

OUTPUT BIN=tdbc_cycle_n30_long BINDIR=FD.sols OUTDIR=FD.out
MESH NX=10, NY=20, NZ=4, TYPE=1
GEOMETRY XLEN=10.0, YLEN=20.0, ZLEN=0.5
DEFORM_GRID 1 1 0.0 10.0 2.0 4.0 8.0 6.0

#Timestep commands
STEP TEXIT=100.0, TIMWSTEPS =
    ↪ 10.0,20.0,30.0,40.0,50.0,60.0,70.0,80.0,90.0,100.0 ;
#     KSAVE=100, TIMFACT = 10.0

#Gravity switched off for this test
RHOG 0.0 0.00  0.0

#Boundary conditions entries
#Other boundaries assumed to be traction free
#Side boundaries in Z-direction
ON X = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=1, UX=0.0 TO 0.0:
```

```
    ↪ &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON X = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=1, UZ=-9000.0 TO
    ↪ -9000.0: &
  TP=1,TDBC=2,TDRV=9000.0,0.0,0.0;
ON X = 10.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=2, UX=0.0 TO 0.0:
    ↪  &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON X = 10.0 FOR -20.0 < Z < 20.0 FOR -20.0 < Y < 30.0 : LB=2, UZ=9000.0 TO
    ↪ 9000.0: &
  TP=1,TDBC=2,TDRV=9000.0,0.0,0.0;
ON Y = 0.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=3, UY=0.0 TO 0.0:
    ↪ &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
ON Y = 20.0 FOR -20.0 < Z < 20.0 FOR -20.0 < X < 30.0 : LB=4, UY=0.0 TO 0.0:
    ↪  &
  TP=1,TDBC=0,TDRV=0.0,0.0,0.0;
PERIODIC Z

#Entry of split nodes on fault surface; on each line there is an entry only
    ↪ for one
#component of fault displacement, i.e., X, Y, or Z direction.
#There is also an entry for time-dependent fault movement.

FAULT X = 5.0 FOR 18.99 < Y < 20.01 FOR -20 < Z < 20 : LB=5, SZ=1.0 TO 1.0:
    ↪ &
  TP=1,TDBC=5,TDRV=0.0,1.0,0.0

#Two types of material: different constant viscosity in each, constant
    ↪ density in each
REG A : MR = 1 ;

REG P { 0.0 65.0 0.0 19.0 0.0 2.0 } : MR = 2 ;

MAT 1 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 1.0 &
        VR = 0 VC = 1.0e20 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25

MAT 2 : DR = 0 RHO = 1.0 0.0 0.0 0.0 ; SE = 3.0 &
        VR = 0 VC = 1.0 0.0 0.0 0.0 ; YOUNGS = 2.5 POISSON = 0.25

#Time-series data: to monitor displacement of midplane surface
SERIES TIME, UMIN, UMAX, VMIN, VMAX, WMIN, WMAX
SERIES UZ = 0.0, 20.0, 0.25 ; UZ = 0.5, 20.0, 0.25 ; UZ = 1.0, 20.0, 0.25 ;
SERIES UZ = 1.5, 20.0, 0.25 ; UZ = 2.0, 20.0, 0.25 ; UZ = 2.5, 20.0, 0.25 ;
SERIES UZ = 3.0, 20.0, 0.25 ; UZ = 3.5, 20.0, 0.25 ; UZ = 4.0, 20.0, 0.25 ;
SERIES UZ = 4.5, 20.0, 0.25 ; UZ = 4.7, 20.0, 0.25 ; UZ = 4.9, 20.0, 0.25 ;
SERIES UZ = 5.1, 20.0, 0.25 ; UZ = 5.3, 20.0, 0.25 ; UZ = 5.5, 20.0, 0.25 ;
SERIES UZ = 6.0, 20.0, 0.25 ; UZ = 6.5, 20.0, 0.25 ; UZ = 7.0, 20.0, 0.25 ;
```

```
SERIES UZ = 7.5, 20.0, 0.25 ; UZ = 8.0, 20.0, 0.25 ; UZ = 8.5, 20.0, 0.25 ;
SERIES UZ = 9.0, 20.0, 0.25 ; UZ = 9.5, 20.0, 0.25 ; UZ = 10.0, 20.0, 0.25 ;
```

A1_Code/tdbc_cycle_n30_long

Appendix D

Mathematica codes for plotting post-seismic models

Plotting Definitions and Initialisation

```
<< EarthUtil`
SetDirectory["~/Documents/Postseismic/Solutions"]
cf1 = ColorData["TemperatureMap"][Mod[Log10[#], 1]] &
g2 = Graphics[{GrayLevel[0.65], Rectangle[{0, 0}, {5, 1}]}];
plogs =
  {Text["Elastic layer", {4, 0.5}], Text["Viscoelastic layer", {4, -0.5}]};
plotting1[ini_, nin_, tmaxin_] :=
 Module[{i = ini, in1, in2, gf, pwr = nin, tmax = tmaxin},
  in1 = grdImport[i];
  gf = GaussianFilter[in1[[3]], 2];
  in1[[3]] = gf;
  in2 = toMatrix[in1];
  in2 = in2[[All, {2, 1, 3}]];
  in2 = Cases[in2, {x_, y_, z_} /; 5 ≤ x && 16.5 ≤ y → {x - 5, y - 19, z}];
  in2[[All, 3]] = in2[[All, 3]]^(1/pwr - 1);
  Show[ListDensityPlot[in2, PlotRange → Full, FrameLabel → {"x/d", "y/d"},
    PlotLabel → "n = " <> ToString[pwr] <> " and t = " <> ToString[tmax],
    LabelStyle → Black, ColorFunction → cf1, ColorFunctionScaling → False,
    FrameStyle → Black, AspectRatio → Automatic, ClippingStyle → Gray,
    ImageSize → 800, BaseStyle → FontSize → 28, PlotRangePadding → None,
    InterpolationOrder → 0, Epilog :→ plogs], ListContourPlot[in2,
    PlotRange → {0, 10.1}, ContourShading → None, Contours → {1.05},
    ContourStyle → {{Thickness[0.005], Dashed, Green, Opacity[0.99]}}], g2]]
plottingv[ini_, elasin_, nin_, tmaxin_] :=
 Module[{i = ini, in1, in2, in3, gf, pwr = nin, tmax = tmaxin, elas = elasin},
  in1 = grdImport[i];
  in3 = grdImport[elas];
  (*Print[{Norm[in1[[3]]],Norm[in3[[3]]]}];*)
  gf = in1[[3]] - in3[[3]];
  (*in1[[3]]=in1[[3]]-in2[[3]];*)
  (*gf=GaussianFilter[in1[[3]],0];*)
  in1[[3]] = Abs[2 gf];
  (*Print[{i,Max@in1[[3]]}];*)
  in2 = toMatrix[in1];
  in2 = in2[[All, {2, 1, 3}]];
  in2 = Cases[in2, {x_, y_, z_} /; 5 ≤ x && 16.5 ≤ y → {x - 5, y - 19, z}];
  (*in2[[All,3]]=in2[[All,3]]^(1/pwr-1);*)
  (*Show[*)ListDensityPlot[in2, PlotRange → Full, FrameLabel → {"x/d", "y/d"},
   PlotLabel → "n = " <> ToString[pwr] <> " and t = " <> ToString[tmax],
   LabelStyle → Black, ColorFunction → (ColorData["Rainbow"][#] &),
   ColorFunctionScaling → False(*,ClippingStyle→ColorData["Rainbow"][0]*),
   FrameStyle → Black, AspectRatio → Automatic,
   ClippingStyle → Gray, ImageSize → 800, BaseStyle → FontSize → 28,
   PlotRangePadding → None, InterpolationOrder → 0](*,
  ListContourPlot[in2,PlotRange→{0,10.1},ContourShading→None,
   Contours→0.1Range[10],ContourStyle→{{Thick,Black}}]*)]
plottingtdbcv[ini_, elasin_, nin_, tmaxin_] :=
 Module[{i = ini, in1, in2, in3, gf, pwr = nin, tmax = tmaxin, elas = elasin},
```

```
  in1 = grdImport[i];
  in3 = grdImport[elas];
  mat = tmax/5 (in1[[1]] - 5);
  mat = Table[mat, {Length[in1[[2]]]}];
  (*Print[{Dimensions[in1[[3]]],Dimensions@mat}];*)
  in3[[3]] = in3[[3]] - mat / tmax;
  in1[[3]] = in1[[3]] - mat;
  (*Print[{Norm[in1[[3]]],Norm[in3[[3]]]}];*)
  gf = in1[[3]] - in3[[3]];
  (*in1[[3]]=in1[[3]]-in2[[3]];*)
  (*gf=GaussianFilter[in1[[3]],0];*)
  in1[[3]] = 2 gf;
  (*Print[{i,Max@in1[[3]]}];*)
  in2 = toMatrix[in1];
  in2 = in2[[All, {2, 1, 3}]];
  in2 = Cases[in2, {x_, y_, z_} /; 5 ≤ x && 16.5 ≤ y → {x - 5, y - 19, z}];
  (*in2[[All,3]]=in2[[All,3]]^(1/pwr-1);*)
  ListDensityPlot[in2, PlotRange → Full, FrameLabel → {"x/d", "y/d"},
   PlotLabel → "n = " <> ToString[pwr] <> " and t = " <> ToString[tmax],
   LabelStyle → Black, ColorFunction → (ColorData["Rainbow"][#] &),
   ColorFunctionScaling → False, ClippingStyle → ColorData["Rainbow"][0],
   FrameStyle → Black, AspectRatio → Automatic,
   ClippingStyle → Gray, ImageSize → 800, BaseStyle → FontSize → 28,
   PlotRangePadding → None, InterpolationOrder → 0]
  (*Show[*)(*,ListContourPlot[in2,PlotRange→{0,10.1},ContourShading→None,
   Contours→0.1Range[10],ContourStyle→{{Thick,Black}}]*)]
plottingtdbcvwide[ini_, elasin_, nin_, tmaxin_] :=
 Module[{i = ini, in1, in2, in3, gf, pwr = nin, tmax = tmaxin, elas = elasin},
  in1 = grdImport[i];
  in3 = grdImport[elas];
  mat = tmax/10 (in1[[1]] - 10);
  mat = Table[mat, {Length[in1[[2]]]}];
  (*Print[{Dimensions[in1[[3]]],Dimensions@mat}];*)
  in3[[3]] = in3[[3]] - mat / tmax;
  in1[[3]] = in1[[3]] - mat;
  (*Print[{Norm[in1[[3]]],Norm[in3[[3]]]}];*)
  gf = in1[[3]] - in3[[3]];
  (*in1[[3]]=in1[[3]]-in2[[3]];*)
  (*gf=GaussianFilter[in1[[3]],0];*)
  in1[[3]] = 2 gf;
  (*Print[{i,Max@in1[[3]]}];*)
  in2 = toMatrix[in1];
  in2 = in2[[All, {2, 1, 3}]];
  in2 = Cases[in2, {x_, y_, z_} /; 10 ≤ x && 16.5 ≤ y → {x - 10, y - 19, z}];
  (*in2[[All,3]]=in2[[All,3]]^(1/pwr-1);*)
  ListDensityPlot[in2, PlotRange → Full, FrameLabel → {"x/d", "y/d"},
   PlotLabel → "n = " <> ToString[pwr] <> " and t = " <> ToString[tmax],
   LabelStyle → Black, ColorFunction → (ColorData["Rainbow"][#] &),
   ColorFunctionScaling → False, ClippingStyle → ColorData["Rainbow"][0],
   FrameStyle → Black, AspectRatio → Automatic,
   ClippingStyle → Gray, ImageSize → 800, BaseStyle → FontSize → 28,
   PlotRangePadding → None, InterpolationOrder → 0]
  (*Show[*)(*,ListContourPlot[in2,PlotRange→{0,10.1},ContourShading→None,
```

```
    Contours→0.1Range[10],ContourStyle→{{Thick,Black}}]*)]

plottingvc[ini_, nin_, tmaxin_] := Module[
  {i = ini, in1, in2, in3, gf, pwr = nin, tmax = tmaxin},
  in1 = grdImport[i];
  in1[[3]] = Abs[in1[[3]]];
  in2 = toMatrix[in1];
  in2 = in2[[All, {2, 1, 3}]];
  in2 = Cases[in2, {x_, y_, z_} /; 5 ≤ x && 16.5 ≤ y → {x - 5, y - 19, z}];
  (*Show[*)ListDensityPlot[in2, PlotRange → Full,
   FrameLabel → {"x/d", "y/d"}(*,PlotLegends→Automatic*),
   PlotLabel → "n = " <> ToString[pwr] <> " and t = " <> ToString[tmax],
   LabelStyle → Black, ColorFunction → "Rainbow",
   (*(ColorData["Rainbow"][#]&),ColorFunctionScaling→False*)(*,
   ClippingStyle→ColorData["Rainbow"][0]*)FrameStyle → Black,
   AspectRatio → Automatic, ClippingStyle → Gray, ImageSize → 800,
   BaseStyle → FontSize → 28, PlotRangePadding → None, InterpolationOrder → 0]
   (*,ListContourPlot[in2,PlotRange→{0,10.1},ContourShading→None,
   Contours→0.1Range[10],ContourStyle→{{Thick,Black}}]*)]

/Users/James/Documents/Postseismic/Solutions

ColorData[TemperatureMap][Mod[Log10[#1], 1]] &
```

Viscosity Plots

```
(*Viscosity plot for postseismic shear free boundary*)
bl1 = BarLegend[{cf1, {0.1, 10}},
   LabelStyle → {Black, FontSize -> 28}, LegendLayout → "Row",
   LegendMarkerSize → 1600, Ticks → Flatten[{0.1, Range[10]}],
   LegendLabel → "Log10(η/η0), cycling each order of magnitude",
   ColorFunctionScaling → False];
(*Grid[{{plotting1["n20.1.ed2i.grd",2,1],
    plotting1["n20_long.10.ed2i.grd",2,100]},
   {plotting1["n30.1.ed2i.grd",3,1],plotting1["n30_long.10.ed2i.grd",3,100]},
   {plotting1["n40.1.ed2i.grd",4,1],plotting1["n40_long.10.ed2i.grd",4,100]},
   {plotting1["n50.1.ed2i.grd",5,1],plotting1["n50_long.10.ed2i.grd",5,100]},
   {bl1,SpanFromLeft}}]
 Export["~/Documents/Thesis/C4_Images/nvisc1.png",%]*)

(*Viscosity plot for postseismic driving boundary conditions*)
(*Grid[
  {{plotting1["tdbc_n20.1.ed2i.grd",2,1],plotting1["tdbc_n20.5.ed2i.grd",2,5],
    plotting1["tdbc_n20_long.10.ed2i.grd",2,100]},
   {plotting1["tdbc_n30.1.ed2i.grd",3,1],plotting1["tdbc_n30.5.ed2i.grd",3,5],
    plotting1["tdbc_n30_long.10.ed2i.grd",3,100]},
   {plotting1["tdbc_n40.1.ed2i.grd",4,1],plotting1["tdbc_n40.5.ed2i.grd",4,5],
    plotting1["tdbc_n40_long.10.ed2i.grd",4,100]},
   {plotting1["tdbc_n50.1.ed2i.grd",5,1],plotting1["tdbc_n50.5.ed2i.grd",5,5],
    plotting1["tdbc_n50_long.10.ed2i.grd",5,100]},
   {bl1,SpanFromLeft,SpanFromLeft}}]
 Export["~/Documents/Thesis/C4_Images/tdbcvisc2.png",%]*)

(*Viscosity plot for earthquake cycle with shear free boundary*)
(*Grid[{{plotting1["cycle_n20.1.ed2i.grd",2,1],plotting1[
     "cycle_n20.4.ed2i.grd",2,4],plotting1["cycle_n20.9.ed2i.grd",2,100]},
   {plotting1["cycle_n30.1.ed2i.grd",3,1],plotting1["cycle_n30.4.ed2i.grd",
     3,4],plotting1["cycle_n30.9.ed2i.grd",3,100]},
   {plotting1["cycle_n40.1.ed2i.grd",4,1],plotting1["cycle_n40.4.ed2i.grd",
     4,4],plotting1["cycle_n40.9.ed2i.grd",4,100]},
   {plotting1["cycle_n50.1.ed2i.grd",5,1],
    plotting1["cycle_n50.4.ed2i.grd",2,4],
    plotting1["cycle_n50.9.ed2i.grd",5,100]},{bl1,SpanFromLeft,SpanFromLeft}}]
 Export["~/Documents/Thesis/C4_Images/cyclevisc2.png",%]*)


(*Viscosity plot for earthquake cycle with driving boundary conditions*)
Grid[{{plotting1["tdbc_cycle_n20.1.ed2i.grd", 2, 1],
   plotting1["tdbc_cycle_n20.5.ed2i.grd", 2, 5],
   plotting1["tdbc_cycle_n20.9.ed2i.grd", 2, 100]},
  {plotting1["tdbc_cycle_n30.1.ed2i.grd", 3, 1],
   plotting1["tdbc_cycle_n30.5.ed2i.grd", 3, 5],
   plotting1["tdbc_cycle_n30.9.ed2i.grd", 3, 100]},
  {plotting1["tdbc_cycle_n40.1.ed2i.grd", 4, 1],
   plotting1["tdbc_cycle_n40.5.ed2i.grd", 4, 5],
   plotting1["tdbc_cycle_n40.9.ed2i.grd", 4, 100]}
   (*,{plotting1["tdbc_cycle_n50.1.ed2i.grd",5,1],
   plotting1["tdbc_cycle_n50.5.ed2i.grd",2,5],
   plotting1["tdbc_cycle_n50.9.ed2i.grd",5,100]}*),
  {bl1, SpanFromLeft, SpanFromLeft}}]

Export["~/Documents/Thesis/C4_Images/tdbccyclevisc2.png", %]

~/Documents/Thesis/C4_Images/tdbccyclevisc2.png
```

Displacement Plots

```
bl2 = BarLegend[{"Rainbow", {0, 1}},
    LabelStyle → {Black, FontSize -> 28}, LegendLayout → "Row",
    LegendMarkerSize → 1000, LegendLabel → "Displacement (u/s)"];
(*Plotting pure postseismic*)
(*Grid[{{plottingv["n10.10.velz.grd","n10.1.velz.grd",1,10],
     plottingv["n10_long.10.velz.grd","n10.1.velz.grd",1,100]},
    {plottingv["n20.10.velz.grd","n20.1.velz.grd",2,10],
     plottingv["n20_long.10.velz.grd","n20.1.velz.grd",2,100]},
    {plottingv["n30.10.velz.grd","n30.1.velz.grd",3,10],
     plottingv["n30_long.10.velz.grd","n30.1.velz.grd",3,100]},
    {plottingv["n40.10.velz.grd","n40.1.velz.grd",4,10],plottingv[
      "n40_long.10.velz.grd","n40.1.velz.grd",4,100]},{bl2,SpanFromLeft}}]
 Export["~/Documents/Thesis/C4_Images/nvelz2.png",%]*)

(*Plotting far field displacement postseismic TDBC*)
(*Grid[{{plottingtdbcv["tdbc_n10.2.velz.grd","tdbc_n10.1.velz.grd",1,2],
     plottingtdbcv["tdbc_n10_long.10.velz.grd","tdbc_n10.1.velz.grd",1,100]},
    {plottingtdbcv["tdbc_n20.2.velz.grd","tdbc_n20.1.velz.grd",2,2],
     plottingtdbcv["tdbc_n20_long.10.velz.grd","tdbc_n20.1.velz.grd",2,100]},
    {plottingtdbcv["tdbc_n30.2.velz.grd","tdbc_n30.1.velz.grd",3,2],
     plottingtdbcv["tdbc_n30_long.10.velz.grd","tdbc_n30.1.velz.grd",3,100]},
    {plottingtdbcv["tdbc_n40.2.velz.grd","tdbc_n40.1.velz.grd",4,2],
     plottingtdbcv["tdbc_n40_long.10.velz.grd",
      "tdbc_n40.1.velz.grd",4,100]},{bl2,SpanFromLeft}}]
 Export["~/Documents/Thesis/C4_Images/tdbcvelz2.png",%]*)


(*Plotting Earthquake cycle*)
(*bl3a=BarLegend[{"Rainbow",{0,1}},LabelStyle→{Black,FontSize->28},
  LegendLayout→"Row",LegendMarkerSize→800,LegendLabel→"Displacement (u/s)"];
bl3b=BarLegend[{"Rainbow",{0,100}},LabelStyle→{Black,FontSize->28},
  LegendLayout→"Row",LegendMarkerSize→800,LegendLabel→"Displacement (u/s)"];
Grid[{{plottingvc["cycle_n10.1.velz.grd",1,1],
     plottingvc["cycle_n10_long.10.velz.grd",1,100]},
    {plottingvc["cycle_n20.1.velz.grd",2,1],
     plottingvc["cycle_n20_long.10.velz.grd",2,100]},
    {plottingvc["cycle_n30.1.velz.grd",3,1],
     plottingvc["cycle_n30_long.10.velz.grd",3,100]},
    {plottingvc["cycle_n40.1.velz.grd",4,1],
     plottingvc["cycle_n40_long.10.velz.grd",4,100]},{bl3a,bl3b}}]
 Export["~/Documents/Thesis/C4_Images/cyclevelz1.png",%]*)

(*Plotting Time Dependent Earthquake cycle*)

FileNames["tdbc_cycle*velz*"]
bl3a = BarLegend[{"Rainbow", {0, 1}},
    LabelStyle → {Black, FontSize -> 28}, LegendLayout → "Row",
    LegendMarkerSize → 800, LegendLabel → "Displacement (u/s)"];
bl3b = BarLegend[{"Rainbow", {0, 100}}, LabelStyle → {Black, FontSize -> 28},
    LegendLayout → "Row", LegendMarkerSize → 800,
    LegendLabel → "Displacement (u/s)"];
Grid[{{plottingvc["tdbc_cycle_n20.1.velz.grd", 1, 1],
    plottingvc["tdbc_cycle_n10_long.10.velz.grd", 1, 100]},
   {plottingvc["tdbc_cycle_n20.1.velz.grd", 2, 1],
    plottingvc["tdbc_cycle_n20_long.10.velz.grd", 2, 100]},
   {plottingvc["tdbc_cycle_n30.1.velz.grd", 3, 1],
```

```
    plottingvc["tdbc_cycle_n30_long.10.velz.grd", 3, 100]},
   {plottingvc["tdbc_cycle_n40.1.velz.grd", 4, 1],
    plottingvc["tdbc_cycle_n40_long.10.velz.grd", 4, 100]}, {bl3a, bl3b}}]
Export["~/Documents/Thesis/C4_Images/tdbccyclevelz1.png", %]

(*
(*Plotting Wide TDBC*)
FileNames["tdbc*wide*velz*"]
 Grid[
  {{plottingtdbcvwide["tdbc_n10_wide.2.velz.grd","tdbc_n10_wide.1.velz.grd",
     1,2],plottingtdbcvwide["tdbc_n10_wide_long.10.velz.grd",
     "tdbc_n10_wide.1.velz.grd",1,100]},
   {plottingtdbcvwide["tdbc_n20_wide.2.velz.grd",
     "tdbc_n20_wide.1.velz.grd",2,2],plottingtdbcvwide[
     "tdbc_n20_wide_long.10.velz.grd","tdbc_n20_wide.1.velz.grd",2,100]},
   {plottingtdbcvwide["tdbc_n30_wide.2.velz.grd",
     "tdbc_n30_wide.1.velz.grd",3,2],plottingtdbcvwide[
     "tdbc_n30_wide_long.10.velz.grd","tdbc_n30_wide.1.velz.grd",3,100]},
   {plottingtdbcvwide["tdbc_n40_wide.2.velz.grd",
     "tdbc_n40_wide.1.velz.grd",4,2],plottingtdbcvwide[
     "tdbc_n40_wide_long.10.velz.grd","tdbc_n40_wide.1.velz.grd",4,100]},
   {plottingtdbcvwide["tdbc_n50_wide.2.velz.grd","tdbc_n50_wide.1.velz.grd",
     5,2],plottingtdbcvwide["tdbc_n50_wide_long.10.velz.grd",
     "tdbc_n50_wide.1.velz.grd",5,100]},{bl2,SpanFromLeft}}]
 Export["~/Documents/Thesis/C4_Images/tdbcvelzwide1.png",%]
```

```
{tdbc_cycle_n10.10.velz.grd, tdbc_cycle_n10.1.velz.grd,
 tdbc_cycle_n10.2.velz.grd, tdbc_cycle_n10.3.velz.grd,
 tdbc_cycle_n10.4.velz.grd, tdbc_cycle_n10.5.velz.grd,
 tdbc_cycle_n10.6.velz.grd, tdbc_cycle_n10.7.velz.grd,
 tdbc_cycle_n10.8.velz.grd, tdbc_cycle_n10.9.velz.grd,
 tdbc_cycle_n10_long.10.velz.grd, tdbc_cycle_n10_long.1.velz.grd,
 tdbc_cycle_n10_long.2.velz.grd, tdbc_cycle_n10_long.3.velz.grd,
 tdbc_cycle_n10_long.4.velz.grd, tdbc_cycle_n10_long.5.velz.grd,
 tdbc_cycle_n10_long.6.velz.grd, tdbc_cycle_n10_long.7.velz.grd,
 tdbc_cycle_n10_long.8.velz.grd, tdbc_cycle_n10_long.9.velz.grd,
 tdbc_cycle_n20.10.velz.grd, tdbc_cycle_n20.1.velz.grd,
 tdbc_cycle_n20.2.velz.grd, tdbc_cycle_n20.3.velz.grd,
 tdbc_cycle_n20.4.velz.grd, tdbc_cycle_n20.5.velz.grd,
 tdbc_cycle_n20.6.velz.grd, tdbc_cycle_n20.7.velz.grd,
 tdbc_cycle_n20.8.velz.grd, tdbc_cycle_n20.9.velz.grd,
 tdbc_cycle_n20_long.10.velz.grd, tdbc_cycle_n20_long.1.velz.grd,
 tdbc_cycle_n20_long.2.velz.grd, tdbc_cycle_n20_long.3.velz.grd,
 tdbc_cycle_n20_long.4.velz.grd, tdbc_cycle_n20_long.5.velz.grd,
 tdbc_cycle_n20_long.6.velz.grd, tdbc_cycle_n20_long.7.velz.grd,
 tdbc_cycle_n20_long.8.velz.grd, tdbc_cycle_n20_long.9.velz.grd,
 tdbc_cycle_n30.10.velz.grd, tdbc_cycle_n30.1.velz.grd,
 tdbc_cycle_n30.2.velz.grd, tdbc_cycle_n30.3.velz.grd,
 tdbc_cycle_n30.4.velz.grd, tdbc_cycle_n30.5.velz.grd,
 tdbc_cycle_n30.6.velz.grd, tdbc_cycle_n30.7.velz.grd,
 tdbc_cycle_n30.8.velz.grd, tdbc_cycle_n30.9.velz.grd,
 tdbc_cycle_n30_long.10.velz.grd, tdbc_cycle_n30_long.1.velz.grd,
 tdbc_cycle_n30_long.2.velz.grd, tdbc_cycle_n30_long.3.velz.grd,
 tdbc_cycle_n30_long.4.velz.grd, tdbc_cycle_n30_long.5.velz.grd,
 tdbc_cycle_n30_long.6.velz.grd, tdbc_cycle_n30_long.7.velz.grd,
 tdbc_cycle_n30_long.8.velz.grd, tdbc_cycle_n30_long.9.velz.grd,
 tdbc_cycle_n40.10.velz.grd, tdbc_cycle_n40.1.velz.grd,
 tdbc_cycle_n40.2.velz.grd, tdbc_cycle_n40.3.velz.grd,
 tdbc_cycle_n40.4.velz.grd, tdbc_cycle_n40.5.velz.grd,
 tdbc_cycle_n40.6.velz.grd, tdbc_cycle_n40.7.velz.grd,
 tdbc_cycle_n40.8.velz.grd, tdbc_cycle_n40.9.velz.grd,
 tdbc_cycle_n40_long.10.velz.grd, tdbc_cycle_n40_long.1.velz.grd,
 tdbc_cycle_n40_long.2.velz.grd, tdbc_cycle_n40_long.3.velz.grd,
 tdbc_cycle_n40_long.4.velz.grd, tdbc_cycle_n40_long.5.velz.grd,
 tdbc_cycle_n40_long.6.velz.grd, tdbc_cycle_n40_long.7.velz.grd,
 tdbc_cycle_n40_long.8.velz.grd, tdbc_cycle_n40_long.9.velz.grd,
 tdbc_cycle_n50.10.velz.grd, tdbc_cycle_n50.1.velz.grd,
 tdbc_cycle_n50.2.velz.grd, tdbc_cycle_n50.3.velz.grd,
 tdbc_cycle_n50.4.velz.grd, tdbc_cycle_n50.5.velz.grd,
 tdbc_cycle_n50.6.velz.grd, tdbc_cycle_n50.7.velz.grd,
 tdbc_cycle_n50.8.velz.grd, tdbc_cycle_n50.9.velz.grd}

~/Documents/Thesis/C4_Images/tdbccyclevelz1.png
```

Appendix E

Mathematica code for Chapter 5

```
(*Run wavelength calculations for selected parameter sets*)
(*Issues with thermal cooling and magma chamber deflation*)
(*Thermal cooling, how much would it contract in total over 141 years*)
(*Compare to GPS and InSAR,
try and do 3 parameter fit for tmax as well as mu and te -
 look at tony poster for how he likes figures*)
years = 3.15569 * 10^7;
B1 = hc β1;
F1 = (u^4 + B1)/(u^4 + B2 u + B1);
I1 = BesselJ[1, A u/hc] (u^4 + B1)^-1 (1 - B2 u/(u^4 + B2 u + B1) Exp[(-F1/τmax) t]);

(*Guestimated parameters from submarine load, A=1500,h0=60m*)
disp[Ain_, h0in_, β1in_, B2in_, tin_, τin_, hcin_] :=
 Ain h0in β1in NIntegrate[I1 /. {β1 → β1in, B2 → B2in,
     A → Ain, t → tin years, τmax → τin years, hc → hcin}, {u, 0, ∞}]
rate[Ain_, h0in_, β1in_, B2in_, tin_, τin_, hcin_] :=
 disp[Ain, h0in, β1in, B2in, tin + 1, τin, hcin] -
  disp[Ain, h0in, β1in, B2in, tin, τin, hcin]
disp2[Ain_, h0in_, β1in_, B2in_, tin_, τin_, hcin_, rin_] :=
 Ain h0in β1in NIntegrate[I1 BesselJ[0, rin/hcin u] /. {β1 → β1in, B2 → B2in,
     A → Ain, t → tin years, τmax → τin years, hc → hcin}, {u, 0, ∞}]
rate2[Ain_, h0in_, β1in_, B2in_, tin_, τin_, hcin_, rin_] :=
 disp2[Ain, h0in, β1in, B2in, tin + 1, τin, hcin, rin] -
  disp2[Ain, h0in, β1in, B2in, tin, τin, hcin, rin]

-F1/τmax;
1 / %
% /.
```

$$-\frac{\left(\text{B2 u}+\text{u}^4+\text{hc}\,\beta 1\right)\,\tau\text{max}}{\text{u}^4+\text{hc}\,\beta 1}$$

```
dpth = 8;
area = 5200 * 10^6;
radius = r /. Solve[{π r^2 == area, r > 0}][[1]] // N
tab4 = ParallelTable[{η, -1000 rate2[radius, dpth, 10^-3.828,
       8.4, 10, 10^η / (10^9 years), 1000, 2 radius]}, {η, 18.5, 21, 0.05}];
tab5 = ParallelTable[{η, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
       10^η / (10^9 years), 5000, 2 radius]}, {η, 18.5, 21, 0.05}];
tab6 = ParallelTable[{η, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
       10^η / (10^9 years), 10000, 2 radius]}, {η, 18.5, 21, 0.05}];
tab7 = ParallelTable[{η, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
       10^η / (10^9 years), 15000, 2 radius]}, {η, 18.5, 21, 0.05}];
tab8 = ParallelTable[{η, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
       10^η / (10^9 years), 25000, 2 radius]}, {η, 18.5, 21, 0.05}];
ListPlot[{tab5, tab6, tab7, tab8, tab4}, Joined → True,
 Frame → True, BaseStyle → FontSize → 36,
 FrameLabel → {"Viscosity, η (Pa s)", "Displacement rate (mm/yr)"},
 PlotLabel → "Peak displacement rate at 140 years, μ=1GPa",
 PlotRangePadding → {{0, 0.02}, {0.2, 0.05}}, ImageSize → 1000,
 PlotStyle → {{Thick, Blue}, {Thick, Red}, {Thick, Darker[Green]},
   {Thick, Purple}, {Thick, Orange}}, PlotLegends → LineLegend[{"Te=5000m",
    "Te=10000m", "Te=15000m", "Te=25000m", "Te=1000m"}, LabelStyle → 36]]
```

```
40 684.3
```

```
rad1 = ParallelTable[{r / radius, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
      10^η / (10^9 years) /. η → 19, 10000, r]}, {r, 0, 5 radius, radius / 10}]
rad2 = ParallelTable[{r / radius, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
      10^η / (10^9 years) /. η → 19, 15000, r]}, {r, 0, 5 radius, radius / 10}]
rad3 = ParallelTable[{r / radius, -1000 rate2[radius, dpth, 10^-3.828, 8.4, 10,
      10^η / (10^9 years) /. η → 19, 25000, r]}, {r, 0, 5 radius, radius / 10}]
rad4 = ParallelTable[{r / radius, -1000 rate2[radius, dpth, 10^-3.828, 8.4,
      10, 10^η / (10^9 years) /. η → 19, 5000, r]}, {r, 0, 5 radius, radius / 10}]
ListPlot[{rad1, rad2, rad3, rad4}, Joined → True, Frame → True,
 BaseStyle → FontSize → 36,
 FrameLabel → {"Position (m)", "Displacement rate (mm/yr)"},
 PlotLabel → "Peak displacement rate at 140 years, μ=1GPa",
 PlotRangePadding → {{0, 0.02}, {0.2, 0.05}}, ImageSize → 1000,
 PlotStyle → {{Thick, Blue}, {Thick, Red}, {Thick, Darker[Green]},
   {Thick, Purple}, {Thick, Orange}}, PlotLegends → LineLegend[{"Te=10000m",
    "Te=15000m", "Te=25000m", "Te=5000m", "Te=1000m"}, LabelStyle → 36]]
```

```
(*Shale*)
dave = 0;
davemax = (21 - 16.5)/0.02 + 1) * 3;
SetSharedVariable[dave]
starttime = SessionTime[];
TableForm[{{"Progressbar", "Total calculations", "Calculations remaining",
   "Estimated time remaining"}, {ProgressIndicator[Dynamic[dave], {0, davemax}],
   davemax, Dynamic[davemax - dave],
   Dynamic[NumberForm[(SessionTime[] - starttime)/dave (davemax - dave), 2]]}}]
tab1 = ParallelTable[{dave++;
    η, -1000 rate[1000, 100, 10^-4.6, 9, 140, 10^η / (5 × 10^9 years), 3000]},
   {η, 16.5, 21, 0.02}];
tab2 = ParallelTable[{dave++;
    η, -1000 rate[1000, 100, 10^-4.6, 9, 140, 10^η / (5 × 10^9 years), 2500]},
   {η, 16.5, 21, 0.02}];
tab3 = ParallelTable[{dave++;
    η, -1000 rate[1000, 100, 10^-4.6, 9, 140, 10^η / (5 × 10^9 years), 2000]},
   {η, 16.5, 21, 0.02}];
(*(*Sandstone*)
tab1a=ParallelTable[
  {η,-1000rate[823,80,10^-3.4,9,140,10^η/(10^9years),3000]},{η,16.5,20,0.02}];
tab2a=ParallelTable[{η,-1000rate[823,80,10^-3.4,9,140,10^η/(10^9years),2500]},
  {η,16.5,20,0.02}];
tab3a=ParallelTable[{η,-1000rate[823,80,10^-3.4,9,140,10^η/(10^9years),2000]},
  {η,16.5,20,0.02}];
(*Limestone*)
tab1b=ParallelTable[
  {η,-1000rate[823,80,10^-4.8,9,140,10^η/(10^9years),3000]},{η,16.5,20,0.02}];
tab2b=ParallelTable[{η,-1000rate[823,80,10^-4.8,9,140,10^η/(10^9years),2500]},
  {η,16.5,20,0.02}];
tab3b=ParallelTable[{η,-1000rate[823,80,10^-4.8,9,140,10^η/(10^9years),2000]},
  {η,16.5,20,0.02}];*)
```

```
pt1 = ListPlot[{tab1, tab2a, tab3},
   Joined → True, Frame → True, BaseStyle → FontSize → 36,
   FrameLabel → {"Viscosity, η (Pa s)", "Displacement rate (mm/yr)"},
   PlotLabel → "Displacement rate at 140 years, μ=5GPa",
   PlotRangePadding → {{0, 0.02}, {0.2, 0.05}}, ImageSize → 1000,
   PlotStyle → {{Thick, Blue}, {Thick, Red}, {Thick, Darker[Green]}}, PlotLegends →
    LineLegend[{"Te=3000m", "Te=2500m", "Te=2000m"}, LabelStyle → 36]]
Export["~/Documents/Michelle/linedisprate5GPa_new.png", pt1]
Export["~/Documents/Michelle/linedisprate5GPa_new.eps", pt1]
(*
(*pt2=
 ListPlot[{tab1a,tab2a,tab3a},Joined→True,Frame→True,BaseStyle→FontSize→36,
    FrameLabel→{"Viscosity, η (Pa s)","Displacement rate (mm/yr)"},
    PlotLabel→"Displacement rate at 140 years, μ=0.5GPa",
    PlotRangePadding→{{0,0.02},{0.2,0.05}},ImageSize→1000,
    PlotStyle→{{Thick,Blue},{Thick,Red},{Thick,Darker[Green]}},
    PlotLegends→LineLegend[{"Te=3000m","Te=2500m","Te=2000m"},LabelStyle→36]]
   pt3=ListPlot[{tab1b,tab2b,tab3b},Joined→True,
   Frame→True,BaseStyle→FontSize→36,
   FrameLabel→{"Viscosity, η (Pa s)","Displacement rate (mm/yr)"},
   PlotLabel→"Displacement rate at 140 years, μ=10GPa",
   PlotRangePadding→{{0,0.02},{0.2,0.05}},ImageSize→1000,
   PlotStyle→{{Thick,Blue},{Thick,Red},{Thick,Darker[Green]}},PlotLegends→
    LineLegend[{"Te=3000m","Te=2500m","Te=2000m"},LabelStyle→36]]*)
Export["~/Documents/Michelle/linedisprate5GPa.eps",pt1]
 Export["~/Documents/Michelle/linedisprate5GPa.png",pt1]
 (*Export["~/Documents/Michelle/linedisprate05GPa.eps",pt2]
  Export["~/Documents/Michelle/linedisprate05GPa.png",pt2]
  Export["~/Documents/Michelle/linedisprate10GPa.eps",pt3]
  Export["~/Documents/Michelle/linedisprate10GPa.png",pt3]*)
```

~/Documents/Michelle/linedisprate5GPa_new.png

~/Documents/Michelle/linedisprate5GPa_new.eps

```
i1 = Import["~/Documents/Michelle/disprate_5GPa.dat"];

pt1 = Show[ListDensityPlot[i1, ImageSize → 1200, PlotRange → {0, -10},
    ClippingStyle → {ColorData["Rainbow"][0], ColorData["Rainbow"][1]},
    InterpolationOrder → 2, PlotRangePadding → None,
    BaseStyle → FontSize → 36, ColorFunction → "Rainbow",
    FrameLabel → {"Elastic Thickness, Te (m)", "Viscosity, η (Pa s)"},
    PlotLabel → "Displacement rate at 140 years, μ=5GPa",
    PlotLegends → BarLegend[Automatic, LabelStyle → 36, LegendLabel →
       Placed["Displacement rate (mm/yr)", Left, Rotate[#, 90 Degree] &],
      LegendMarkerSize → 900]], ListContourPlot[i1, Contours → {-4, -6, -8},
    ContourShading → None, ContourStyle → Thick, InterpolationOrder → 2]]
```

SetOptions[SelectedNotebook[], PrintingStyleEnvironment -> "Printout",
ShowSyntaxStyles -> True]

SetOp

```
(*Initialisation cell*)
(*Run wavelength calculations for selected parameter sets*)
(*Issues with thermal cooling and magma chamber deflation*)
(*Thermal cooling, how much would it contract in total over 141 years*)
(*Compare to GPS and InSAR,
try and do 3 parameter fit for tmax as well as mu and te -
 look at tony poster for how he likes figures*)

<< EarthUtil`
years = 3.15569 * 10^7;
b1 = 6 (1 - ν) ρc 9.81 / mu;
b2 = 12 (1 - ν);
B1 = hc β1;
F1 = (u^4 + B1) / (u^4 + B2 u + B1);
I1 = BesselJ[1, A u / hc] (u^4 + B1)^-1 (1 - B2 u / (u^4 + B2 u + B1) Exp[-F1 / τmax t]);
TeXForm[I1]
I2 = D[I1, t];
(*Guestimated parameters from submarine load, A=1500,h0=60m*)
disp = Compile[{{Ain, _Real}, {h0in, _Real}, {β1in, _Real},
    {B2in, _Real}, {tin, _Real}, {τin, _Real}, {hcin, _Real}},
  Module[{hc = hcin, β1 = β1in, B2 = B2in, B1, A = Ain,
    τmax = τin years, t = tin years, tmax},
   B1 = hc β1;
   F1 = (u^4 + B1) / (u^4 + B2 u + B1);
   I1 = BesselJ[1, A u / hc] (u^4 + B1)^-1 (1 - B2 u / (u^4 + B2 u + B1) Exp[-F1 / τmax t]);
   Ain h0in β1in NIntegrate[I1, {u, 0, ∞}]
  ]
 ]
disp2 = Compile[{{Ain, _Real}, {h0in, _Real}, {β1in, _Real},
    {B2in, _Real}, {tin, _Real}, {τin, _Real}, {hcin, _Real}, {rin, _Real}},
  Module[{hc = hcin, β1 = β1in, B2 = B2in, B1, A = Ain,
    τmax = τin years, t = tin years, tmax, r = rin},
   B1 = hc β1;
   F1 = (u^4 + B1) / (u^4 + B2 u + B1);
   I1 =
    BesselJ[1, A u / hc] BesselJ[0, r / hc u] (u^4 + B1)^-1 (1 - B2 u / (u^4 + B2 u + B1) Exp[-F1 / τmax t]);
   Ain h0in β1in NIntegrate[I1, {u, 0, ∞}]
  ]
 ]

SetDirectory["~/Documents/Tony Watts/newgrids"];
t1 = Import["fullload.mx"][[All, {2, 1, 3}]];
heights = t1[[All, 3]];
importantpts = t1[[Flatten@Position[heights, x_ /; x ≠ 0]]];
```

```
heightspos = importantpts[[All, 3]];
cf = Compile[{{x, _Real}, {y, _Real}},
  Module[{pos2, ans},
   pos2 = importantpts[[All, 1 ;; 2]];
   ans = Norm[{x - pos2[[#, 1]], y - pos2[[#, 2]]}] & /@ Range[1, Length@pos2]
  ]
 ]
scaled = heightspos.(f[#] & /@ cf[#[[1]], #[[2]]]) &;

(*

rate[Ain_,h0in_,β1in_,B2in_,tin_,τin_,hcin_]:=years Ain h0in β1in NIntegrate[
     I2/.{β1→β1in,B2→B2in,A→Ain,t→tin years,τmax→τin years,hc→hcin},{u,0,∞}]
    disp2[Ain_,h0in_,β1in_,B2in_,tin_,τin_,hcin_,rin_]:=
   Ain h0in β1in NIntegrate[I1 BesselJ[0, rin/hcin u ]/.
       {β1→β1in,B2→B2in,A→Ain,t→tin years,τmax→τin years,hc→hcin},{u,0,∞}]
     rate2[Ain_,h0in_,β1in_,B2in_,tin_,τin_,hcin_,rin_]:=
    years Ain h0in β1in NIntegrate[I2 BesselJ[0, rin/hcin u ]/.
       {β1→β1in,B2→B2in,A→Ain,t→tin years,τmax→τin years,hc→hcin},{u,0,∞}]
     SetDirectory["~/Documents/Tony Watts/Paper1_Figures/"];*)
eta = 10^η;(*.10^16.3;*)
mu = 10^μ;
ρc = 1800;
ν = 0.25;
b1 = 6 (1 - ν) ρc 9.81 / mu;
b2 = 12 (1 - ν);
τ = (eta / mu) / years;
(*
pt2=Import["data1.wdx"];
pt2b=Import["data2.wdx"];*)
```

CompiledFunction[WVM | Argument count : 7 | Argument types : {_Real, _Real, _Real, _Real, _Real, _Real, _Real}]

CompiledFunction[WVM | Argument count : 8 | Argument types : {_Real, _Real, _Real, _Real, _Real, _Real, _Real, _Real}]

CompiledFunction[WVM | Argument count : 2 | Argument types : {_Real, _Real}]

```
ν =.; mu =.; ρc =.;
b1 = 6 (1 - ν) ρc 9.81 / mu
b2 = 12 (1 - ν)
TeXForm[b1]
TeXForm[b2]
```

$$\frac{58.86\ (1-\nu)\ \rho c}{\text{mu}}$$

$$12\ (1-\nu)$$

```
12 (1-\nu )
```

```
Te = 500;
{b1a, b2a, τ} = {b1, b2, (10^η / 10^μ) / years} /. {μ → 7.0, η → 15.9}
disp[1150, 110, b1a, b2a, 140, τ, Te]
f1 = Interpolation@ParallelTable[
    {dist, -disp2[39.4721, 1, b1a, b2a, 140, τ, Te, dist]}, {dist, 0, 3000, 100}]
Te = 1000;
f2 = Interpolation@ParallelTable[
    {dist, -disp2[39.4721, 1, b1a, b2a, 140, τ, Te, dist]}, {dist, 0, 3000, 100}]
Te = 2500;
f3 = Interpolation@ParallelTable[
    {dist, -disp2[39.4721, 1, b1a, b2a, 140, τ, Te, dist]}, {dist, 0, 3000, 100}]
Plot[{f1[x], f2[x], f3[x]}, {x, 0, 3000}, PlotRange → Full, ImageSize → 1000,
 Frame → True, BaseStyle → FontSize → 22, FrameStyle → Black,
 FrameLabel -> {"Radial distance (m)", "Displacement for 1m high point load"},
 PlotStyle → {{Thick, Blue}, {Thick, Red}, {Thick, Darker[Green]}},
 GridLines → {{0}, {0}}, PlotRangePadding → {None, Automatic},
 GridLinesStyle → Directive[Gray, Dashed, Thick],
 PlotLegends → Placed[LineLegend[{"Te=500m", "Te=1000m", "Te=2500m"},
    LabelStyle → {Black, 22}, LegendMarkerSize → 50], {0.84, 0.25}]]
Export["~/Documents/Thesis/C5_Images/greensfcns.pdf", %]
(*scaled[{4000,2500}]*)
```

```
{0.0079461, 9., 25.1713}
```

```
113.038
```

```
InterpolatingFunction[ Domain : {{0., 3.00 ×10^3}}
                       Output : scalar ]
```

```
InterpolatingFunction[ Domain : {{0., 3.00 ×10^3}}
                       Output : scalar ]
```

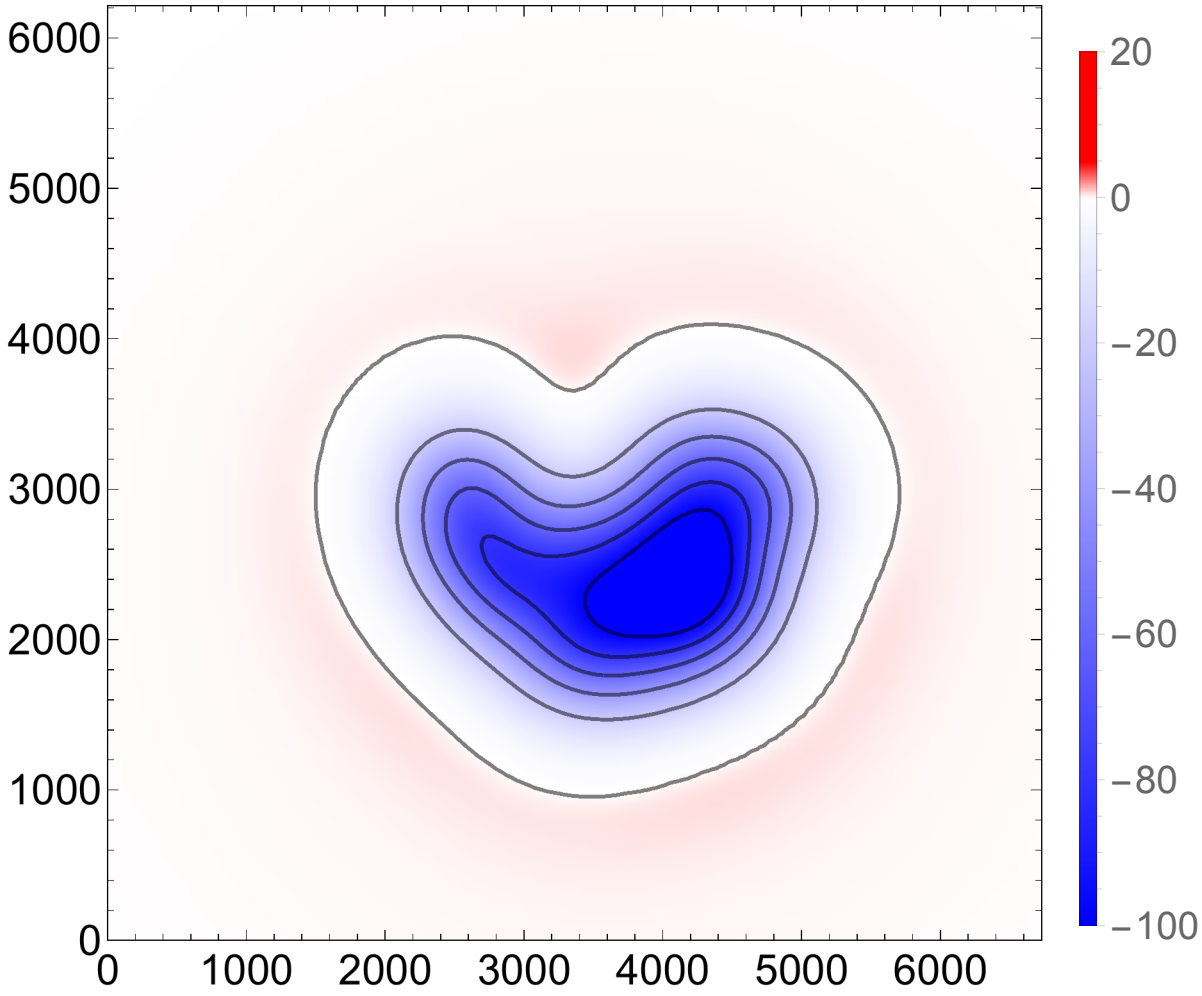
```
InterpolatingFunction[ Domain : {{0., 3.00 ×10^3}}
                       Output : scalar ]
```

```
~/Documents/Thesis/C5_Images/greensfcns.pdf
```

```
b1 = Blend[{{-100, Blue}, {0, White}, {5, Red}}, #] &
```

```
Blend[{{-100, Blue}, {0, White}, {5, Red}}, #1] &
```

```
Show[
 ListDensityPlot[t2, ColorFunction → b1, ColorFunctionScaling → False,
  ImageSize → 800, PlotRange → Full, PlotRangePadding → None,
  BaseStyle → FontSize → 22, FrameStyle → Black, PlotLegends → BarLegend[
    {b1, {-100, 20}}, LabelStyle → {FontSize → 20}, LegendMarkerSize → 780]],
 ListContourPlot[t2, ContourShading → None, ContourStyle → {{Thick, Black}},
  PlotRange → Full, Contours → Range[20, -100, -20]]
]
```



```
SetDirectory["~/Documents/Tony Watts/newgrids"];
files = FileNames["*.grd"]
in1a = grdImport[files[[2]]];
mat1 = toMatrix[in1a];
mat1[[All, 3]] = t2[[All, 3]];
v1 = DeleteDuplicates@mat1[[All, 1]]
v2 = DeleteDuplicates@mat1[[All, 2]]
Partition[mat1[[All, 3]], 90]
Export["displacements.dat", mat1]
(*
lcp2=Show[ListDensityPlot[mat1[[All,{2,1,3}]],
   ColorFunction→({(*ColorData["Rainbow"][#]*)b1[#],Opacity[0.5]}&),
   ColorFunctionScaling→None,InterpolationOrder→1,PlotRange→Full],
  ListContourPlot[mat1[[All,{2,1,3}]],ContourShading→None,
   Contours→{0,10,-20,-40,-60,-80,-100},
   ContourStyle→{{Thick,Black}},PlotRange→Full]];
bl2=BarLegend[{b1,{-100,20}},ColorFunctionScaling→False,
  LabelStyle→FontSize→20,LegendFunction→frame,
  LegendMarkerSize→400,LegendLabel→"Difference (m)"];
gg2=Globe[GeoGraphics[lcp2[[1]],GeoRange→{{36.377`,36.433`},{25.358`,25.433`}},
    GeoRangePadding→Quantity[1,"Kilometer"],
    GeoBackground→GeoStyling["StreetMapNoLabels"],
    Frame→True,GeoScaleBar→Placed[Automatic,{Left,Top}],
    ImageSize→1000,BaseStyle→FontSize→22],{36.38,25.36}]
  Row[{%,bl2},"    "]
  Export["1866-1928driving.png",%]

  (*
  in1a[[All,3]]=t2[[All,3]];
  overlay1=Show[
    ListDensityPlot[in1a,ColorFunction→b1,ColorFunctionScaling→False,
     ImageSize→800,PlotRange→Full,PlotRangePadding→None,Frame→None],
    ListContourPlot[in1a[[All,{2,3,1}]],ContourShading→None,
     ContourStyle→{{Thick,Black}},PlotRange→Full,Contours→Range[20,-100,-20]]
   ]


(*This first set of code if for the single loading case*)
α1 = ((20 - 15)/0.05 + 1) (30) ((10.5 - 6)/0.1 + 1);
SetSharedVariable[dave1]
dave1 = 0;
ProgressIndicator[Dynamic[dave1], {0, α1}]
pt1 = ParallelTable[{dave1++;
      η, μ, 100 Te, disp[1150, 110, b1, b2, 140, τ, 100 Te],
      1000 rate[1150, 110, b1, b2, 140, τ, 100 Te]}, {η, Range[15, 20, 0.05]},
     {Te, 1, 30}, {μ, 6, 10.5, 0.1}] // AbsoluteTiming;
pt1[[1]]
pt2 = Flatten[pt1[[2]], 2];
ListPlot[pt2[[All, {1, 4}]]]
Export["data1.wdx", pt2]
```

```
(*This second set of code if for the double loading case*)
α2 = (20 - 15)/0.05 + 1) (30) ((10.5 - 6)/0.1 + 1);
SetSharedVariable[dave2]
dave2 = 0;
ProgressIndicator[Dynamic[dave2], {0, α2}]
pt1b = ParallelTable[{dave2++;
      η, μ, 100 Te, disp[1150, 110, b1, b2, 140, τ, 100 Te] + disp[550, 50, b1,
        b2, 78, τ, 100 Te], 1000 rate[1150, 110, b1, b2, 140, τ, 100 Te],
      1000 rate[550, 50, b1, b2, 78, τ, 100 Te]}, {η, Range[15, 20, 0.05]},
     {Te, 1, 30}, {μ, 6, 10.5, 0.1}] // AbsoluteTiming;
pt1b[[1]]
pt2b = Flatten[pt1b[[2]], 2]
ListPlot[pt2b[[All, {1, 4}]]]
Export["data2.wdx", pt2b]

(*Plotting code, 2D slices through solution space, for the single load case*)
b3 = Blend[{{100, Blue}, {15, White}}, #] &;
b1 = Blend[{{-100, Blue}, {0, White}, {5, Red}}, #] &
plot1[μin_] := Module[{μ1 = μin},
  ptcases = Cases[pt2, {i_, j_, k_, l_, m_} /; j == μ1 → {i, k, l, m}];
  lcp1 = ListContourPlot[ptcases[[All, {1, 2, 4}]], PlotRange → Full,
    ContourShading → {None, Directive[Green, Opacity[0.4]]},
    Contours → {2, 10}, ContourStyle → {{Thickness[0.0025], Black}}];
  ldp2 = ListDensityPlot[ptcases[[All, {1, 2, 3}]], PlotRange → All,
    ClippingStyle → ColorData["TemperatureMap"][1], LabelStyle → Black,
    PlotLabel → "Displacements and rates (green shading 2-10mm/yr)\n
       at 140 years for μ=5×10^7Pa, r=1150m, h=110m",
    FrameLabel → {"Log10(Viscosity)", "Elastic Lid Thickness (m)"},
    ColorFunction → b3, ColorFunctionScaling → False,
    PlotRangePadding → None, ImageSize → 1000,
    BaseStyle → {Black, FontSize → 30}, FrameStyle → Black,
    PlotLegends → BarLegend[Automatic, LabelStyle → FontSize → 36, LegendLabel →
       Placed[Rotate["Displacement at 140 years (m)", 90 °], Left]]];
  lcp2 = ListContourPlot[ptcases[[All, {1, 2, 3}]], PlotRange → Full,
    ContourShading → None, PlotRangePadding → None, Contours → Range[10, 200, 10],
    ContourStyle → {{Thick, Dashed}}, ImageSize → 1000];
  Show[ldp2, lcp2, lcp1]]
mus = DeleteDuplicates@pt2[[All, 2]]
filenames = "mu" <> ToString[#] <> ".png" & /@ mus;
Export["S1.png", plot1[7.6]]
(*
plots1=ParallelMap[plot1[#]&,mus];
Export[#1,#2]&@@@Transpose[{filenames,plots1}]
```

```
Blend[{{-100, Blue}, {0, White}, {5, Red}}, #1] &
```

```
{6., 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 6.7, 6.8, 6.9, 7., 7.1, 7.2, 7.3, 7.4, 7.5,
 7.6, 7.7, 7.8, 7.9, 8., 8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 8.7, 8.8, 8.9, 9., 9.1,
 9.2, 9.3, 9.4, 9.5, 9.6, 9.7, 9.8, 9.9, 10., 10.1, 10.2, 10.3, 10.4, 10.5}
```

```
S1.png
```

```
(*Plotting code, 2D slices through solution space, for double load case*)
ptcases2 = Cases[pt2b,
    {i_, j_, k_, l_, m_, n_} /; j == 7 && m > 0 && n > 0 && m > n → {i, k, l, m, n}];
lcp1 = ListContourPlot[ptcases2[[All, {1, 2, 4}]], PlotRange → {1, 15},
    ContourShading → {None, Directive[Green, Opacity[0.4]]},
    Contours → {2, 10}, ContourStyle → {{Thickness[0.0025], Black}}];
ldp2 = ListDensityPlot[ptcases2[[All, {1, 2, 3}]],
    PlotRange → Full, LabelStyle → Black,
    PlotLabel → "Displacements and rates (green shading 2-10mm/yr)\n at 140
       years for μ=10^7Pa, r=1150m, h=110m",
    FrameLabel → {"Log10(Viscosity)", "Elastic Lid Thickness (m)"},
    ColorFunction → "TemperatureMap", PlotRangePadding → None,
    ImageSize → 1000, BaseStyle → {Black, FontSize → 30}, FrameStyle → Black,
    PlotLegends → BarLegend[Automatic, LabelStyle → FontSize → 36, LegendLabel →
       Placed[Rotate["Displacement at 140 years (m)", 90 °], Left]]];
lcp2 = ListContourPlot[ptcases2[[All, {1, 2, 3}]], PlotRange → Full,
    ContourShading → None, PlotRangePadding → None, Contours → Range[10, 200, 10],
    ContourStyle → {{Thick, Dashed}}, ImageSize → 1000];
Show[ldp2, lcp2, lcp1]

(*Code to generate time series, for single load case*)
(*15.95,8.1,600*)
(*15.75`,7.1`,1300*)
pt4[[1]]
singleload[params_] :=
 Block[{η = params[[1]], μ = params[[2]], Te = params[[3]]},
  (*τ=(η/μ)/years;*)
  pt3 = ParallelTable[
     {{tim + 1866, 1, 1}, -disp[1150, 110, b1, b2, tim, τ, Te]}, {tim, 0, 150, 1}];
  rt3 = 1000 rate[1150, 110, b1, b2, 140, τ, Te];
  DateListPlot[Re[pt3], ImageSize → 1200,
   BaseStyle → FontSize → 24, Joined → True, PlotStyle → {{Thick, Blue}},
   PlotRange → Full, FrameStyle → Black, LabelStyle → Black,
   FrameLabel → {"Year", "Vertical Displacement (m)"},
   PlotLabel → "Peak subsidence, single loading event\n (Log10η=" <>
     ToString[η] <> ", Log10μ=" <> ToString[μ] <> ", Te=" <> ToString[Te] <> ")",
   PlotRangePadding → {{0, 10}, {Scaled[0.05], 0}},
   GridLines → {None, {pt3[[-1, 2]]}},
   Epilog → {Text[ToString[NumberForm[rt3, 2]] <> "mm/yr → ",
      {{2005, 1, 1}, 0.98 pt3[[-1, 2]]}, Background → White]}]]
plots = singleload[pt5[[-1]]]; // AbsoluteTiming
plots
Export["bestfitsingle.eps", %]
```

```
{15., 6., 100, 109.876, 4.21984}
```

```
{12.077848, Null}
```

```
bestfitsingle.eps
```

```
(*Code to generate time series, for double load case*)

doubleload[params_] := Block[{η = params[[1]], μ = params[[2]], Te = params[[3]]},
  (*τ=(η/μ)/years;*)
  pt3 = ParallelTable[{{tim + 1866, 1, 1},
      (-disp[1021, 120, b1, b2, tim, τ, Te] - HeavisideTheta[tim - 62.1]
         disp[660, 110, b1, b2, tim - 62, τ, Te])}, {tim, 0, 155, 1}];
  rt3 = 1000 rate[1150, 110, b1, b2, 140, τ, Te];
  rt4 = 1000 rate[650, 50, b1, b2, 78, τ, Te];
  DateListPlot[Re[pt3], ImageSize → 1200,
   BaseStyle → FontSize → 24, Joined → True, PlotStyle → {{Thick, Blue}},
   PlotRange → Full, FrameStyle → Black, LabelStyle → Black,
   FrameLabel → {"Year", "Vertical Displacement (m)"},
   PlotLabel → "Peak subsidence, two loading events\n (Log10η=" <>
     ToString[η] <> ", Log10μ=" <> ToString[μ] <> ", Te=" <> ToString[Te] <> ")",
   PlotRangePadding → {{0, 10}, {Scaled[0.05], 0}},
   GridLines → {None, {pt3[[-1, 2]]}},
   Epilog → {Text[ToString[NumberForm[rt3, 2]] <> "mm/yr → ",
      {{2005, 1, 1}, 0.97 pt3[[-1, 2]]}, Background → White]
      (*Text[Style[ToString[NumberForm[rt4,2]]<>"mm/yr → ",Red],
      {{2005,1,1},0.95pt3[[-1,2]]},Background→White]*)}]]
plots2 = doubleload[{15.9, 7.6, 1000}](*pt5b[[;;;;100]]*); // AbsoluteTiming
plots2
Export["bestfitdouble.eps", %]
(*

Block[{η=16,μ=6.4,Te=400},
  pt3=ParallelTable[{{tim+1866,1,1},-disp[1150,110,b1,b2,tim,τ,Te]-
      HeavisideTheta[tim-62.1]disp[650,50,b1,b2,tim-62,τ,Te]},{tim,0,155,1}];
  rt3=1000rate[1150,110,b1,b2,140,τ,Te];
  rt4=1000rate[650,50,b1,b2,78,τ,Te]]//AbsoluteTiming
  (*pl4=Table[{{tim+1866,1,1},0},{tim,1,63}]~Join~ParallelTable[
       {{tim+1866,1,1},-2.5disp[550,70,b1,b2,tim,τ,400]},{tim,64,155}]
     pt3[[All,2]]=pt3[[All,2]]+pl4[[All,2]]*)
  DateListPlot[Re[pt3],ImageSize→1200,BaseStyle→FontSize→24,
   Joined→True,PlotStyle→{{Thick,Blue}},PlotRange→Automatic
   (*{Ceiling[pt3[[1,2]],5],Floor[pt3[[-1,2]],5]}*),FrameStyle→Black,
   (*AxesOrigin→{pt3[[1,1]],Floor[pt3[[-1,2]],5]},*)LabelStyle→Black,
   FrameLabel→{"Year","Vertical Displacement (m)"},PlotLabel→
    "Peak subsidence, for double load case\n (η=10^15.7Pa s, μ=10^7Pa, Te=1400m)",
   PlotRangePadding→{{0,10},{Scaled[0.05],0}},GridLines→{None,{pt3[[-1,2]]}},
   Epilog→{Text[ToString[NumberForm[rt3,2]]<>"mm/yr → ",
      {{2005,1,1},0.98pt3[[-1,2]]},Background→White],
     Text[Style[ToString[NumberForm[rt4,2]]<>"mm/yr → ",Red],
      {{2005,1,1},0.95pt3[[-1,2]]},Background→White]}]
  (*Export["tvary.eps",%]*)
{27.549922, Null}

bestfitdouble.eps

Length@pt5b

81
```

```
start2 = ParallelMap[disp[1150, 110, b1 /. μ → #[[2]], b2, 0,
       τ /. {μ → #[[2]], η → #[[1]]}, #[[3]]] &, pt4b]; // AbsoluteTiming
start2
```

```
{208.639247, Null}
```

```
mid2 = ParallelMap[disp[1150, 110, b1 /. μ → #[[2]], b2, 62,
       τ /. {μ → #[[2]], η → #[[1]]}, #[[3]]] &, pt4b]; // AbsoluteTiming
```

```
{252.830302, Null}
```

```
pt4 = Cases[pt2, {i_, j_, k_, l_, m_} /; 60 < l < 110 && 1 < m < 8 → {i, j, k, l, m}]
grp = GatherBy[DeleteDuplicates[Sort[%[[All, {2, 3}]]]], First];
(*{2,3} for shear modulus against elastic
 thickness or {2,1} for shear modlus against viscosity*)
low = {#[[1, 1]], Min[#[[All, 2]]]} & /@ grp;
high = {#[[1, 1]], Max[#[[All, 2]]]} & /@ grp;
ListPlot[{low, high}, Frame → True, ImageSize → 1200,
 BaseStyle → FontSize → 36, FrameStyle → Black, PlotLabel →
  "Region of acceptable parameters \n (60m < Δz < 110m, 2mm/yr < v < 8mm/yr)",
 LabelStyle → Black, PlotStyle → Blue,
 FrameLabel → {"Shear Modulus (Pa)", "Log10(Viscosity)"},
 Joined → True, Filling → {1 → {2}}, PlotRangePadding → None]
(*Export["region2.png",%]*)
```

```
lpp3 = ListPointPlot3D[pt4[[All, {2, 1, 3}]],
    ImageSize → 1500, ColorFunction → (ColorData["Rainbow"][#3] &),
    PlotStyle → {PointSize[Large], Opacity[0.8]}, PlotLabel →
     "Acceptable parameters \n (60m < Δz < 110m, 2mm/yr < v < 8mm/yr)",
    PlotRangePadding → None, LabelStyle → Black,
    BaseStyle → FontSize → 22, AxesLabel →
     {"Log10(Shear Modulus)", "Log10(Viscosity)", "Elastic Thickness (m)"},
    AxesStyle → Black, PlotLegends → BarLegend[{"Rainbow", {100, 3000}},
      LegendMarkerSize → 400, LabelStyle → Directive[Black, FontSize → 22],
      LegendLabel → Placed[Rotate["Elastic Thickness (m)", 90 °], Left]]];
dp1 = DensityHistogram[pt4[[All, {2, 1}]], 30, ColorFunction → "SolarColors",
    ImageSize → 1000, PlotRange → {{6, 10.5}, {15, 20}},
    PlotRangePadding → None, Frame → None];
gr = Graphics3D[{Texture[dp1], EdgeForm[], Opacity[0.5],
     Polygon[{{6, 15, 0}, {10.5, 15, 0}, {10.5, 20, 0}, {6, 20, 0}},
      VertexTextureCoordinates → {{0, 0}, {1, 0}, {1, 1}, {0, 1}}]}];
Show[lpp3, gr, PlotRange → All, BoxRatios → {1, 1, .6},
 FaceGrids → {Back, Left}, Lighting → "Neutral"]
(*
Export["3dplot2.png",%]
```

```
pt4b = Cases[pt2b, {i_, j_, k_, l_, m_, n_} /; 80 < l < 200 && 2 < m < 12 && n < m && n < 4]
%[[All, {-2, -1}]];
ListPlot[%, AspectRatio → Automatic, AxesLabel → {"1866 Rate", "1928 Rate"},
 PlotLabel → "InSAR Rates from two loads, all parameters",
 BaseStyle → FontSize → 22, ImageSize → 1000,
 PlotStyle → {PointSize[Small], Blue}]
ListPointPlot3D[pt4b[[All, 1 ;; 3]]]
(*,Plot[{x,2},{x,1,10},PlotStyle→{{Thick,Red},{Dashed,Gray}}]]*)
```

```
lpp3b = ListPointPlot3D[pt4b[[All, {2, 1, 3}]],
    ImageSize → 1500, ColorFunction → (ColorData["Rainbow"][#3] &),
    PlotStyle → {PointSize[Large], Opacity[0.8]},
    PlotLabel → "Acceptable parameters for two loads \n (60m < Δz <
       110m, 2mm/yr < v < 8mm/yr, v > u)", PlotRangePadding → None,
    LabelStyle → Black, BaseStyle → FontSize → 22, AxesLabel →
     {"Log10(Shear Modulus)", "Log10(Viscosity)", "Elastic Thickness (m)"},
    AxesStyle → Black, PlotLegends → BarLegend[{"Rainbow", {100, 3000}},
      LegendMarkerSize → 400, LabelStyle → Directive[Black, FontSize → 22],
      LegendLabel → Placed[Rotate["Elastic Thickness (m)", 90 °], Left]]];
dp1b = DensityHistogram[pt4b[[All, {2, 1}]], 30, ColorFunction → "SolarColors",
    ImageSize → 1000, PlotRange → {{6, 10.5}, {15, 20}},
    PlotRangePadding → None, Frame → None];
grb = Graphics3D[{Texture[dp1b], EdgeForm[], Opacity[0.5],
     Polygon[{{6, 15, 0}, {10.5, 15, 0}, {10.5, 20, 0}, {6, 20, 0}},
      VertexTextureCoordinates → {{0, 0}, {1, 0}, {1, 1}, {0, 1}}]}];
Show[lpp3b, grb, PlotRange → All, BoxRatios → {1, 1, .6},
 FaceGrids → {Back, Left}, Lighting → "Neutral"]
(*Export["3dplot3.png",%]*)
```

```
waveLength[ηin_, μin_, Tein_] := Module[
  {η = ηin, μ = μin, Te = Tein, tim = 140, pt3w, sp, r1, x, eta, mu, ρc, ν, b1, b2, τ},
  eta = 10^η; (*.10^16.3;*)
  mu = 10^μ;
  ρc = 1800;
  ν = 0.25;
  b1 = 6 (1 - ν) ρc 9.81 / mu;
  b2 = 12 (1 - ν);
  τ = (eta / mu) / years;
  pt3w = Table[{r, -disp2[856, 134, b1, b2, tim, τ, Te, r]}, {r, 0, 7000, 500}];
  sp = Cases[pt3w, {x_, y_} /; y > 0];
  If[Length[sp] > 0,
   r1 = FindRoot[Interpolation[pt3w, InterpolationOrder → 1][x],
     {x, 0, sp[[1, 1]]}], r1 = {x -> pt3w[[-1, 1]] + 1}];
  {η, μ, Te, x /. r1, pt3w[[1, 2]], Max[pt3w[[All, 2]]]}]
waveLengthData[ηin_, μin_, Tein_] := Module[
  {η = ηin, μ = μin, Te = Tein, tim = 140, pt3w, sp, r1, x, eta, mu, ρc, ν, b1, b2, τ},
  eta = 10^η; (*.10^16.3;*)
  mu = 10^μ;
  ρc = 1800;
  ν = 0.25;
  b1 = 6 (1 - ν) ρc 9.81 / mu;
  b2 = 12 (1 - ν);
  τ = (eta / mu) / years;
  pt3w = Table[{r, -disp2[856, 134, b1, b2, tim, τ, Te, r]}, {r, 0, 7000, 100}]]
(*
ListPlot[pt3w,ImageSize→1000,PlotRange→Full]
 r1
```

```
Interpolation[pt3w, InterpolationOrder → 1][x]
Plot[%, {x, 1000, 6000}]
```

```
InterpolatingFunction[ Domain : {{0., 6.00 ×10^3}}
                       Output : scalar ][x]
```

```
(*Initialisation Cells*)

(*Equation simplification*)
days = 86 400;
newmodel = 1/(R1^3 t1 μ^2 (1 + ν)) p0 π R2^3
  (-3 (-1 + e^(R1^3 t μ (1+ν)/(3 R2^3 η (-1+ν))) R2^3 η (-1 + ν) + R1^3 (3 (-1 + e^(R1^3 t μ (1+ν)/(3 R2^3 η (-1+ν))) η (-1 + ν) + t μ (1 + ν)) +
    (3 (-1 + e^(R1^3 (t-t1) μ (1+ν)/(3 R2^3 η (-1+ν))) R2^3 η (-1 + ν) + R1^3 (-3 (-1 + e^(R1^3 (t-t1) μ (1+ν)/(3 R2^3 η (-1+ν))) η (-1 + ν) -
        (t - t1) μ (1 + ν))) HeavisideTheta[t - t1]);
FullSimplify[newmodel /. p0 → α μ t1/(π R1^3 tr)];
FullSimplify[% /. μ → 3 η (1 - ν) R2^3/((1 + ν) R1^3 tr)];
bob = FullSimplify[% /. {R2 → R1 R}];
(*bob2=bob/.{t→t-(t1+dt)}*)
(*model2=bob/.{tr→#1,R→#2,α→#3,t1→#4}&
   model3=bob2/.{tr→#1,R→#2,α→#3,t1→#4,dt→20}&*)
(*bob;*)
Assuming[t > t1, Simplify[bob]];
(*Limit[%,t→∞,Assumptions→{tr>0,R>0,α>0}]*)
bob /. α → tr β/R^3 t1
steve = FullSimplify[% /. β → β1] HeavisideTheta[t];
steve2 = steve /. {t1 → t3, β1 → β2}
steve2 = steve2 /. {t → t - t2, t3 → t3 - t2};
model = steve + steve2 // FullSimplify
(*Data Import*)
ays = 86 400;
SetDirectory["~/Documents/Michelle/"];
Needs["ErrorBarPlots`"]
data = Drop[Import["joint_new2.dat"], 2];
Max[data];
data[[All, 2]] = (data[[All, 2]] - Min[data]);(*/(Max[data]-Min[data])*)
data1 = Cases[data, {x_, y_, z_} → {x, 1000^-3 y(*-0.0001dx*)(*-0.0006*)}];
errors = Cases[data, {x_, y_, z_} → 1000^-3 z];
data1[[All, 2]] = data1[[All, 2]] * 10^3;
errors = errors * 10^3;
datan = Partition[Riffle[data1, ErrorBar[#] & /@ errors], 2];
```

$$\frac{1}{R^3\,t1}e^{-\frac{t}{tr}}\beta\left(\left(-1+R^3\right)tr+e^{t/tr}\left(R^3\,(t-tr)+tr\right)+\left(-e^{t1/tr}\left(-1+R^3\right)tr+e^{t/tr}\left(-tr+R^3\,(-t+t1+tr)\right)\right)\text{HeavisideTheta}[t-t1]\right)$$

$$\frac{1}{R^3\,t3}e^{-\frac{t}{tr}}\beta 2\,\text{HeavisideTheta}[t]\left(\left(-1+R^3\right)tr+e^{t/tr}\left(R^3\,(t-tr)+tr\right)+\left(-e^{t3/tr}\left(-1+R^3\right)tr+e^{t/tr}\left(-tr+R^3\,(-t+t3+tr)\right)\right)\text{HeavisideTheta}[t-t3]\right)$$

$$\frac{1}{R^3} e^{-\frac{t}{tr}} \left(\frac{1}{t1} \beta 1 \, \text{HeavisideTheta}[t] \left(\left(-1+R^3\right) tr + e^{t/tr} \left(R^3 \, (t-tr) + tr\right) + \left(-e^{t1/tr} \left(-1+R^3\right) tr + e^{t/tr} \left(-tr + R^3 \, (-t+t1+tr)\right)\right) \text{HeavisideTheta}[t-t1]\right) + \frac{1}{t2-t3} \beta 2 \, \text{HeavisideTheta}[t-t2] \left(-e^{t2/tr} \left(-1+R^3\right) tr + e^{t/tr} \left(-tr + R^3 \, (-t+t2+tr)\right) + \left(e^{t3/tr} \left(-1+R^3\right) tr + e^{t/tr} \left(R^3 \, (t-t3-tr) + tr\right)\right) \text{HeavisideTheta}[t-t3]\right)\right)$$

```
model /. Flatten[
  {{tr → 87.7}, {β1 -> Max[data1[[All, 2]]] / 2}, {β2 -> Max[data1[[All, 2]]] / 2},
   {R -> 1.5}, {t1 -> 178.1}, {t2 -> 291.5}, {t3 -> 398.2}}]
Show[Plot[%, {t, 0, 800}, PlotStyle → {Thick, Red}, ImageSize → 1200,
  Frame → True, BaseStyle → FontSize → 24], ListPlot[data1]]
```

$$0.296296\, e^{-0.0114025\, t} \left(-0.100354 \left(-5783.42 + e^{0.0114025\, t} \left(-87.7 + 3.375 \left(379.2 - t\right)\right) + \left(19\,524. + e^{0.0114025\, t} \left(87.7 + 3.375 \left(-485.9 + t\right)\right)\right) \text{HeavisideTheta}[-398.2 + t]\right) \text{HeavisideTheta}[-291.5 + t] + 0.0601222 \left(208.288 + e^{0.0114025\, t} \left(87.7 + 3.375 \left(-87.7 + t\right)\right) + \left(-1587.17 + e^{0.0114025\, t} \left(-87.7 + 3.375 \left(265.8 - t\right)\right)\right) \text{HeavisideTheta}[-178.1 + t]\right) \text{HeavisideTheta}[t]\right)$$

```
presmodel = Piecewise[{{β1 t / t1, t < t1}, {β1, t1 ≤ t ≤ t2},
    {β1 + β2 (t - t2) / (t3 - t2), t2 < t < t3}, {β1 + β2, t ≥ t3}}];
param = {tr, β1, β2, R, t1, t2, t3, x};
model2 = model + 0.0001 x;
(*works well with t2>280*)
nlm4 = NonlinearModelFit[data1, {model, 170 < t1 < 184, t2 > 285, t3 > 390},
   {{tr, 77.7}, {β1, Max[data1[[All, 2]]] / 2}, {β2, Max[data1[[All, 2]]] / 2},
    {R, 1.5}, {t1, 178.1}, {t2, 291.5}, {t3, 398.2}, x}, t,
   Weights → 1 / errors^2, VarianceEstimatorFunction → (1 &)];
nlm4["BestFitParameters"]
Print["95%"]
nlm4["ParameterConfidenceIntervals", ConfidenceLevel → .95]
Table[{param[[i]], NumberForm[param[[i]] /. nlm4["BestFitParameters"], 4],
  NumberForm[Mean[Abs[(param[[i]] /. nlm4["BestFitParameters"]) - # & /@ %[[i]]]],
   2]}, {i, 1, 8}]
Print["99.7%"]
nlm4["ParameterConfidenceIntervals", ConfidenceLevel → .997]
Table[{param[[i]], NumberForm[param[[i]] /. nlm4["BestFitParameters"], 4],
  NumberForm[Mean[Abs[(param[[i]] /. nlm4["BestFitParameters"]) - # & /@ %[[i]]]],
   2]}, {i, 1, 8}]
clevels = %[[All, 3]];
nlm4["ParameterTable"]
Grid[Transpose[{#, nlm4[#]} &[{"AdjustedRSquared", "AIC",
    "BIC", "RSquared", "EstimatedVariance"}]], Alignment → Left]
resp = ListPlot[nlm4["FitResiduals"], Filling → Axis, ImageSize → 200]
(*
{fr,sr1,sr2}=
 nlm4[{"FitResiduals","StandardizedResiduals","StudentizedResiduals"}];
Map[ListPlot[#,Filling→0,ImageSize→750]&,{fr,sr1,sr2}]//Row
  {Min[#],Max[#]}&[nlm4["SingleDeletionVariances"]]
  ListPlot[nlm4["SingleDeletionVariances"],PlotRange→{0,All},Filling→0]
  ListPlot[nlm4["HatDiagonal"],PlotRange→{0,All},Filling→0]
  ListPlot[Transpose[nlm4[{"Response","PredictedResponse"}]],
   FrameLabel→{"observed","predicted"},Frame→True,Axes→False]
```

```
{tr → 98.0211, β1 → 11.4434, β2 → 9.77136,
 R → 1.78895, t1 → 184., t2 → 285., t3 → 390., x → 0.001}

{{95.3538, 100.688}, {11.3481, 11.5386}, {9.70379, 9.83894}, {1.66958, 1.90831},
 {178.285, 189.715}, {282.376, 287.624}, {385.446, 394.554}, {0.001, 0.001}}

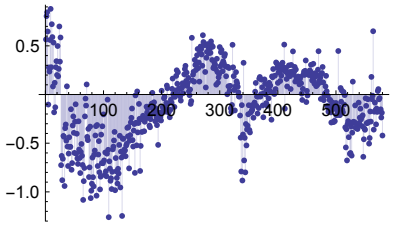
{{tr, 98.02, 2.7}, {β1, 11.44, 0.095}, {β2, 9.771, 0.068}, {R, 1.789, 0.12},
 {t1, 184., 5.7}, {t2, 285., 2.6}, {t3, 390., 4.6}, {x, 0.001, 0.}}

{{93.9736, 102.069}, {11.2988, 11.5879}, {9.66882, 9.87391}, {1.60781, 1.97008},
 {175.328, 192.672}, {281.019, 288.981}, {383.089, 396.911}, {0.001, 0.001}}

{{tr, 98.02, 4.}, {β1, 11.44, 0.14}, {β2, 9.771, 0.1}, {R, 1.789, 0.18},
 {t1, 184., 8.7}, {t2, 285., 4.}, {t3, 390., 6.9}, {x, 0.001, 0.}}
```

	Estimate	Standard Error	t- Statistic	P- Value
tr	98.0211	1.35798	72.1815	1.59552×10^{-289}
$\beta 1$	11.4434	0.0485061	235.916	$4.499903722210 \times 10^{-572}$
$\beta 2$	9.77136	0.0344051	284.009	$9.04257203319 \times 10^{-618}$
R	1.78895	0.0607744	29.4358	1.22569×10^{-116}
t1	184.	2.90959	63.2391	2.50536×10^{-260}
t2	285.	1.33579	213.357	$2.193750646772 \times 10^{-547}$
t3	390.	2.31868	168.199	$3.053307840124 \times 10^{-489}$
x	0.001	0.	∞	$0. \times 10^{-308}$

```
AdjustedRSquared   0.999651
AIC                2697.28
BIC                2736.55
RSquared           0.999655
EstimatedVariance  1
```




```
Clear[TwoAxisDateListPlot];
TwoAxisDateListPlot[f_List, g_List, opts : OptionsPattern[]] :=
 Module[{p1, p2, fm, fM, gm, gM, old, new, newg},
  p1 = DateListPlot[f, Axes → True, Frame → False, PlotRange → Automatic];
p2 = DateListPlot[g, Axes → True, Frame → False, PlotRange → Automatic];
  {fm, fM} = AbsoluteOptions[p1, PlotRange][[1, 2, 2]];
{gm, gM} = AbsoluteOptions[p2, PlotRange][[1, 2, 2]];
  old = AbsoluteOptions[p2, Ticks][[1, 2, 2]];
  new =
   Flatten[{Rescale[First[#1], {gm, gM}, {fm, (fM + 1)}], Rest[#1]}, 1] & /@ old;
  newg = {#[[1]], Rescale[#[[2]], {gm, gM}, {fm, fM}]} & /@ g;
  new[[All, 2]] = If[# ≠ "", IntegerPart[#]] & /@ new[[All, 2]];
  DateListPlot[{f, newg}, Axes → False, Frame → True,
   FrameTicks → {Automatic, All, None, new}, PlotRange → {(fm - 1), (fM + 1)}, opts]]
seisdates = Flatten[Import["dates.dat"]];
sdat = "seismicity_July13_matlab.dat";
gpsdates = Flatten[Import["dates_joint.dat"]];
timegps = AbsoluteTime[#] & /@ gpsdates;
data2 = Cases[data1, {x_, y_} → {x, If[x < 80, y - 0.3, y]}];
errors2 = Drop[errors, Dimensions[errors][[1]] - Dimensions[data2][[1]]];
datag = datan;
datag[[All, 1, 2]] = data2[[All, 2]];
Dimensions[timegps][[1]] - Dimensions[data2][[1]];
```

```
timegps = Drop[timegps, %];
seisold = Partition[Riffle[seisdates, Import["j_seis.dat"][[All, 2]]], 2];
seis = Drop[Import[sdat], 1][[All, {1, 3}]];
test1 = Partition[Riffle[timegps, data2[[All, 2]]], 2];
test1[[-1, 2]] = test1[[-1, 2]] + 1;
seis = Drop[seis, 1];
seis[[All, 2]] = seis[[All, 2]] * 10^-21;
ta = Min[timegps];
(*st1=clevels[[5]];
st2=clevels[[6]];
st3=clevels[[7]];*)
st2 = 11;
st3 = 11;
dates2 =
 {(t2) days + ta - st2 days, (t2) days + ta + st2 days} /. nlm4["BestFitParameters"]
dates3 = {(t3) days + ta - st3 days, (t3) days + ta + st3 days} /.
  nlm4["BestFitParameters"]
t1i = t1 /. nlm4["BestFitParameters"];
p12 = Plot[10^3 0.025, {x, t1i days + ta - 11 days, t1i days + ta + 11 days},
   Filling → Bottom, PlotRange → 10^3 {-0.015, 0.025}];
p12b = Plot[10^3 0.025, {x, dates2[[1]], dates2[[2]]},
   Filling → Bottom, PlotRange → 10^3 {-0.015, 0.025}];
p12c = Plot[10^3 0.025, {x, dates3[[1]], dates3[[2]]},
   Filling → Bottom, PlotRange → 10^3 {-0.015, 0.025}];

dc = Directive[Thick, Gray];
de = Directive[Blue];
presmodel /. {t -> t - Min[timegps], dt → dt days,
   t1 → t1 days, t2 → t2 days, t3 → t3 days, t0 → Min[timegps]};
Piecewise[{{0, t ≤ Min[timegps]},
   {% /. nlm4["BestFitParameters"], t > Min[timegps]}}];
p7 = Plot[%, {t, ta(*0.99Min[timegps]*), Max[timegps]},
   PlotStyle → {Thick, Dashed, Darker[Green]}];

m1 = model /.
   {tr → tr days, t1 → t1 days, t2 → t2 days, t3 → t3 days, t → t - Min[timegps]};
m2 = m1 /. nlm4["BestFitParameters"];

p9 = Plot[m2, {t, ta(*0.999Min[timegps]*), Max[timegps]},
   PlotRange → {0, 10^3 0.0225}, PlotStyle → {{Thick, Red}, {Thick, Dashed, Gray}},
   BaseStyle → FontSize → 24, ImageSize → 1200, Frame → True,
   FrameLabel → {"Days", "Volume Change (km^3)"}];
pleg = PointLegend[{Darker[Blue], {Darker[Orange], 36}},
   {"Input data", "CO2 output"},
   LegendMarkers → {{"▪", 24}, {Graphics[Disk[]], 24}}, LabelStyle → 24];
lleg = LineLegend[{{Red, Thickness[0.2]}, {Darker[Green], Thickness[0.2], Dashed},
    {Gray, Thickness[0.2]}, Blue, {Thickness[0.2], Black}},
   {"Best fit model", "Pressure change", "Fitted dates",
    "Errors on dates", "Seismicity"}, LabelStyle → 24];

glines2 = {{{ta, dc}, {t1 days + ta - 11 days, de}, {t1 days + ta, dc},
    {t1 days + ta + 11 days, de}, {t2 days + ta - st2 days, de}, {t2 days + ta, dc},
    {t2 days + ta + st2 days, de}, {t3 days + ta - st3 days, de}, {t3 days + ta, dc},
    {t3 days + ta + st3 days, de}}, None} /. nlm4["BestFitParameters"]
```

```
plogs = {Text["t0", {ta, 10^3 0.0215}, Background → White],
    Text["t2", {t2 days + ta, 10^3 0.0215}, Background → White],
    Text["t1", {(t1) days + ta, 10^3 0.0215}, Background → White], Text["t3",
     {t3 days + ta, 10^3 0.0215}, Background → White]} /. nlm4["BestFitParameters"]

axplot =
  TwoAxisDateListPlot[test1, Prepend[seis, {DateString[ta], 0}], PlotStyle →
    {{Thick, PointSize[Large], Black, Opacity[0]}, {Black, Opacity[1]}},
    Joined → True, BaseStyle → {FontSize → 36}, ImageSize → 1600,
    FrameLabel → {{Text["Volume change (×10^6 m^3)"], "Cumulative seismic
moment (×10^14 N m)"}, {"Date", " "}}, DateTicksFormat ->
     {"MonthNameShort", " ", "YearShort"}, GridLines → glines2,
    PlotLegends → {Placed[pleg, {Right, Bottom}], Placed[lleg, {Right, Bottom}]},
    Prolog :→ plogs];
datah = datag;
datah[[All, 1, 1]] = timegps;
ep1 = ErrorListPlot[datah];
Show[axplot, p12, p12b, p12c, ep1, p7, p9, gasplot]
(*Export["twopulsesfinal5.png",%]
 Export["twopulsesfinal5.pdf",%%]*)
{t1, t2, t3} days /. nlm4["BestFitParameters"];
DateString[# + ta] & /@ %
```

$\{3.52555 \times 10^9, 3.52745 \times 10^9\}$

$\{3.53462 \times 10^9, 3.53652 \times 10^9\}$

$\{\{\{3\,501\,878\,400, \text{Directive}[\text{Thickness}[\text{Large}], \text{GrayLevel}[0.5]]\}, \{3.51683 \times 10^9, \text{Directive}[\text{RGBColor}[0, 0, 1]]\}, \{3.51778 \times 10^9, \text{Directive}[\text{Thickness}[\text{Large}], \text{GrayLevel}[0.5]]\}, \{3.51873 \times 10^9, \text{Directive}[\text{RGBColor}[0, 0, 1]]\}, \{3.52555 \times 10^9, \text{Directive}[\text{RGBColor}[0, 0, 1]]\}, \{3.5265 \times 10^9, \text{Directive}[\text{Thickness}[\text{Large}], \text{GrayLevel}[0.5]]\}, \{3.52745 \times 10^9, \text{Directive}[\text{RGBColor}[0, 0, 1]]\}, \{3.53462 \times 10^9, \text{Directive}[\text{RGBColor}[0, 0, 1]]\}, \{3.53557 \times 10^9, \text{Directive}[\text{Thickness}[\text{Large}], \text{GrayLevel}[0.5]]\}, \{3.53652 \times 10^9, \text{Directive}[\text{RGBColor}[0, 0, 1]]\}\}, \text{None}\}$

$\{\text{Text}[t_0, \{3\,501\,878\,400, 21.5\}, \text{Background} \to \text{GrayLevel}[1]], \text{Text}[t_2, \{3.5265 \times 10^9, 21.5\}, \text{Background} \to \text{GrayLevel}[1]], \text{Text}[t_1, \{3.51778 \times 10^9, 21.5\}, \text{Background} \to \text{GrayLevel}[1]], \text{Text}[t_3, \{3.53557 \times 10^9, 21.5\}, \text{Background} \to \text{GrayLevel}[1]]\}$

{Thu 23 Jun 2011 00:00:00, Sun 2 Oct 2011 00:00:00, Sun 15 Jan 2012 00:00:00}

```
AbsoluteTime[#] & /@ {"16-Jun-2011", "8-Oct-2011", "23-Jan-2012"}
(% - ta) / days
{# - 7, # + 7} & /@ %
```

```
{3 517 171 200, 3 527 020 800, 3 536 265 600}
```

```
{177, 291, 398}
```

```
{{170, 184}, {284, 298}, {391, 405}}
```

```
sc = 2;
{"04-21-2011", "07-21-2011", "09-15-2011", "01-29-2012"};
Transpose@{AbsoluteTime /@ %, {24, 21, 29, 38} / sc};
ErrorBar /@ ({5, 4, 5, 6} / sc);
Partition[Riffle[%%, %], 2];
gasplot = ErrorListPlot[%, PlotRange → Full,
    PlotStyle → {PointSize[0.01], Darker[Orange]}];
allplotted = Show[axplot, p12, p12b, p12c, ep1, p7, p9, gasplot,
  PlotRangePadding → {{5 000 000, 500 000}, {0, 1}}, ImageSize → 1600, Epilog →
   {Inset[Text[Rotate[Style["CO₂ output (tonnes day⁻¹)", Darker[Orange]], 90 °]],
     {ta - 20 days, 11}], Inset[Text[Style["20", Darker[Orange]]], {ta - 42 days,
      10}], Inset[Text[Style["10", Darker[Orange]]], {ta - 42 days, 5}],
    Inset[Text[Style["30", Darker[Orange]]], {ta - 42 days, 15}],
    Inset[Text[Style["40", Darker[Orange]]], {ta - 42 days, 20}],
    Inset[Text[Style["0", Darker[Orange]]], {ta - 46 days, 0}]}]
Export["twopulsesandgas.png", %]
Export["twopulsesandgas.pdf", %%]
```

```
twopulsesandgas.png
```

```
twopulsesandgas.pdf
```