Preface

The 2022 GFD Program theme was *Data-Driven GFD* with Professors Peter Schmid of King Abdullah University of Science and Technology (KAUST) and Laure Zanna of New York University serving as principal lecturers. Together they introduced the masked audience in the re-opened cottage and on the better-ventilated porch to a fascinating mixture of data-driven methods and their potential application to fluid mechanics in general and GFD in particular. The first ten chapters of this volume document these lectures, each prepared by teams of the summer's GFD fellows. Due to Covid-related restrictions, there was space for only eight fellows this summer, who were:

- Ludovico Giorgini, Stockholm University
- Sam Lewin, University of Cambridge
- Ruth Moorman, California Institute of Technology
- Kasturi Shah, Massachusetts Institute of Technology
- Iury Simoes-Sousa, University of Massachusetts Dartmouth
- Claire Valva, New York University
- Tilly Woods, University of Oxford
- Rui Yang, University of Twente

Their reports are included in this volume.

In 2022, the Sears Public Lecture was delivered by Professor Heidi Nepf of the Massachusetts Institute of Technology on the topic of "Coastal Vegetation and Coastal Flows: Restoration, Climate Mitigation and Adaptation". The topic was extremely well presented by Heidi to a large and appreciative audience who learnt a lot about the fascinating interactions between vegetation and fluid motions in the near-shore. Much animated discussion followed in the evening afterwards outside Redfield Auditorium as refreshments were consumed.

Stefan Llewellyn Smith and Colm-cille Caulfield acted as directors, and in spite of the Covid-related challenges a large number of long-term staff members ensured that the fellows never lacked for guidance. The seminar series was filled by a steady stream of visitors talking about topics as diverse as flow structures affecting search and rescue to bubbles in weightless water. Anders Jensen worked his usual magic in the Lab, dealing inventively with squishy balls and recalcitrant currents. As ever, Janet Fields and Julie Hildebrandt kept the program running smoothly behind the scenes, with their assistance (and limitless patience) hugely appreciated by the directors.

Table of Contents

Principal Lectures

Fellows Reports:

2022 Participants

FELLOWS

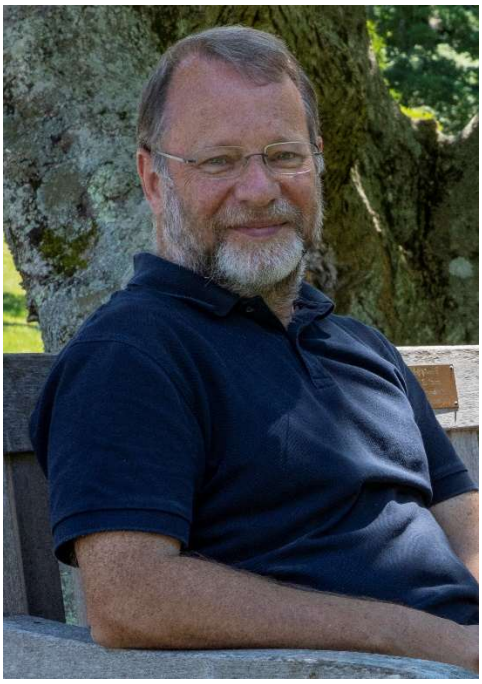
Ludovico Giorgini	Stockholm University
Samuel Lewin	University of Cambridge
Ruth Moorman	California Institute of Technology
Kasturi Shah	Massachusetts Institute of Technology
Iury Simoes-Sousa	University of Massachusetts Dartmouth
Claire Valva	New York University
Tilly Woods	University of Oxford
Rui Yang	University of Twente

STAFF AND VISITORS

Ashley Arroyo	Yale University
Peter Baddoo	Massachusetts Institute of Technology
Elizabeth Bailey	Yale University
Lois Baker	University of Edinburgh
Neil Balmforth	University of British Columbia
Aurora Basinski-Ferris	New York University
Biswajit Basu	Trinity College Dublin
Anthony Bonfils	Nordic Institute of Theoretical Physics
Freddy Bouchet	ENS de Lyon
Samuel Boury	New York University
Michael Brenner	Harvard University
Andrew Brettin	New York University
Marko Budišić	Clarkson University
Keaton Burns	Massachusetts Institute of Technology
Elizabeth Carlson	California Institute of Technology
Colm-cille Caulfield	University of Cambridge
Claudia Cenedese	Woods Hole Oceanographic Institution
Gregory Chini	University of New Hampshire
Laura Cope	University of Leeds
Rodrigo Duran	Theiss Research
Thomas Eaves	University of Dundee
Kelsey Everard	New York University
Raffaele Ferrari	Massachusetts Institute of Technology
Alessia Ferraro	École Polytechnique Fédérale de Lausanne
Glenn Flierl	Massachusetts Institute of Technology
Adrian Fraser	University of Colorado
Basile Gallet de Saint-Aurin	CEA, Saclay
Pascale Garaud	University of California, Santa Cruz
Renske Gelderloos	Delft University of Technology
Pedram Hassanzadeh	University of Chicago

Karl Helfrich	Woods Hole Oceanographic Institution
Edward Johnson	University College London
Alexis Kaminski	University of California, Berkeley
Wanying Kang	Massachusetts Institute of Technology
Richard Kerswell	University of Cambridge
Zhiming Kuang	Harvard University
Anuj Kumar	University of California, Santa Cruz
Daniel Lecoanet	Northwestern University
Stefan Llewellyn Smith	University of California, San Diego
Annika Margevich	Yale University
James McElwaine	Durham University
Gianluca Meneghello	Massachusetts Institute of Technology
Philip Morrison	University of Texas, Austin
Sankalp Nambiar	NORDITA
Heidi Nepf	Massachusetts Institute of Technology
Jason Olsthoorn	Queen's University
Jeremy Parker	ECPS, EPFL
Joseph Pedlosky	Woods Hole Oceanographic Institution
Channing Prend	University of California, San Diego
Jason Prochaska	University of California, Santa Cruz
Anubhab Roy	Indian Institute of Technology, Madras
Christopher Rycroft	University of Wisconsin, Madison
Matthew Scase	University of Nottingham
Tobias Schneider	EPFL, STI IGM, ECPS
Mattia Serra	University of California, San Diego
Andre Souza	Massachusetts Institute of Technology
Adam Subel	New York University
Daisuke Takagi	University of Hawaii at Manoa
Lynne Talley	Scripps Institution of Oceanography
Shuwen Tan	Columbia University
Mary-Louise Timmermans	Yale University
Barbara Turnbull	University of Nottingham
John Whitehead	Woods Hole Oceanographic Institution
Madeleine Youngs	New York University
Laure Zanna	New York University

2022 Principal Lecturers



Peter Schmid
King Abdullah University of Science and Technology

Laure Zanna
New York University

2022 Geophysical Fluid Dynamics Summer School Participants

First Row (L-R): Iury Simoes-Sousa, Ruth Moorman, Tilly Woods, Samuel Lewin, Ludovico Giorgini, Rui Yang, Claire Valva, Kasturi Shah

Second Row (L-R): Jack Whitehead (standing), Andrew Brettin, Shuwen Tan, Jeremy Parker, Laure Zanna, Peter Schmid, Phil Morrison, Renske Gelderloos, Wanying Kang, Pascale Garaud, DaisukeTakagi, Neil Balmforth

Third Row: Aurora Basinski-Ferris, Karl Helfrich, Adam Subel, Rich Kerswell, Colm Caulfield, Gianluca Meneghello, Stefan Llewelyn Smith, Andre Souza, Glenn Flierl, Keaton Burns, Ted Johnson, Peter Baddoo, Pedram Hassanzadeh, Anubhab Roy, Elizabeth Carlson

Not in photo: Ashley Arroyo, Elizabeth Bailey, Lois Baker, Biswajit Basu, Freddy Bouchet, Anthony Bonfils, Samuel Boury, Marko Budišić, Claudia Cenedese, Laura Cope, Miles Couchman, Tom Eaves, Kelsey Everard, Adrian Fraser, Basile Gallet, Alexis Kaminski, Zhiming Kuang, Anuj Kumar, Daniel Lecoanet, Andrea Lehn, Annika Margevich, Jim McElwaine, Sankalp Nambiar, Jason Olsthoorn, Joe Pedlosky, Channing Prend, Xavier Prochaska, Christopher Rycroft, Matthew Scase, Mattia Serra, Lynne Talley, Shuwen Tan, Mary-Louise Timmermans, Barbara Turnbull, Madeleine Youngs, Hadi Zolfaghari

Lecture Schedule

PRINCIPAL LECTURES

Tuesday, June 21

Review of Data-decomposition Based on Linear Algebra
Peter Schmid

Wednesday, June 22

Spatio-temporal Decomposition of Time Series
Laure Zanna

Thursday, June 23

Transfer Operator for Data Analysis (part 1)
Peter Schmid

Friday, June 24

Transfer Operator for Data Analysis (part 2)
Peter Schmid

Monday, June 27

Forced Response from Climate Statistics
Laure Zanna

Uncertainty, Outliers, Predictability
Peter Schmid

Tuesday, June 28

Bayesian and Markovian Approaches to Data Analysis
Laure Zanna

Wednesday, June 29

Discovering Equations and Operators from Data
Laure Zanna

Thursday, June 30

Advanced Approaches in Signal Processing
Peter Schmid

Friday, July 1

Advanced Approaches in ML for Physics
Laure Zanna

SEMINARS

Monday, July 4

HOLIDAY

Tuesday, July 5

Accurately and Efficiently Modeling Turbulent Flows: Mathematical Analysis and Computations on the Effectiveness and Robustness of the AOT Algorithm
Elizabeth Carlson, University of Victoria

Wednesday, July 6

Uncovering the Rules of Crumpling with a Data-driven Approach
Christopher Rycroft, Harvard University

Thursday, July 7

Sedimenting Anisotropic Particles: Dynamics of Ice Crystals in Clouds
Anubhab Roy, Indian Institute of Technology

Friday, July 8

Search and Rescue at Sea Aided by Hidden Flow Structures
Mattia Serra, University of California, San Diego

Monday, July 11

Optimizing Scalar Transport Using Three-dimensional Branching Pipe Flows
Anuj Kumar, University of California, Santa Cruz

The Impact of Realistic Topographic Representation on the Parameterization of Lee Wave Energy Flux
Lois Baker, Imperial College London

Tuesday, July 12

Meridional Buoyancy Transport by Baroclinic Turbulence
Basile Gallet, CEA Saclay

Wednesday, July 13

Improving Efficiency of Data Assimilation by Particle Filters Using Data-driven Model Decompositions
Marko Budišić, Clarkson University

Thursday, July 14

Baroclinic Annular Mode in the Southern Hemisphere
Madeleine Youngs, New York University

Asymptotic Interpretation of the Miles Mechanism of Wind-Wave Instability
Anthony Bonfils, NORDITA

Friday, July 15

Large-scale and Mesoscale Dynamics of the Arctic Ocean
Gianluca Meneghello, Massachusetts Institute of Technology

Hydrodynamics of Slender Swimmers Near Deformable Interfaces
Sankalp Nambiar, NORDITA

Monday, July 18

A Double Diffusion Calculation That Produces Continents and Ocean Basins
Jack Whitehead, Woods Hole Oceanographic Institution

Tuesday, July 19

Learning Data-driven Subgrid-scale Models: Stability, Extrapolation, and Interpretation
Pedram Hassanzadeh, Rice University

Wednesday, July 20

Invariant Tori in Turbulence and Chaos
Jeremy Parker, École Polytechnique Fédérale de Lausanne

Thursday, July 21

Pondering How Ocean Waters Rise from the Abyss: From Theory to Observations
Raffaele Ferrari, Massachusetts Institute of Technology

Friday, July 22

Towards Hygienic Modelling of Complex Phenomena: From Asymptotic Expansions (Implemented in Code) to Machine-learned Closures
Michael Brenner, Harvard University

Monday, July 25

Predicting Extreme Heat Waves Using Rare Even Simulations and Deep Neural Networks
Freddy Bouchet, École Normale Supérieure de Lyon

Tuesday, July 26

On the 3D Modelling of Equatorial Undercurrent and Some Insight into Particle Paths in Stratified Rotational Flows
Biswajit Basu, Trinity College Dublin

Wednesday, July 27

Transition and Equilibria in Stratified Flows
Tom Eaves, University of Dundee

Thursday, July 28

Physics-informed Dynamic Mode Decomposition
Peter Baddoo, Massachusetts Institute of Technology

Friday, July 29

Non-deal Instabilities in Sinusoidal Shear Flows with a Streamwise Magnetic Field
Adrian Fraser, University of California, Santa Cruz

Monday, August 1

Modern Spectral Methods for PDEs
Keaton Burns, Massachusetts Institute of Technology

Tuesday, August 2

Topographic Solitary Waves and Groups
Glenn Flierl, Massachusetts Institute of Technology

Wednesday, August 3

Snapshots and Transition Problems
Andre Souza, Massachusetts Institute of Technology

Thursday, August 4

Experiments on Air Bubbles in Weightless Water and the Instability of a Rotating Jet
Matthew Scase, University of Nottingham

SEARS PUBLIC LECTURE

Coastal Vegetation and Coastal Flows: Restoration, Climate Mitigation and Adaptation
Heidi Nepf, Massachusetts Institute of Technology

Friday, August 5

Vegetation Hydrodynamics for Restoration, Climate Mitigation and Adaptation
Heidi Nepf, Massachusetts Institute of Technology

Monday, August 8

Southeast Greenland Fjord-Shelf Interaction at Subinertial Frequencies
Renske Gelderloos, Johns Hopkins University

Tuesday, August 9

Distilling the Physics of Primitive-equation Model Currents
Rodrigo Duran, Theiss Research

Wednesday, August 10

Exploiting Marginal Stability in Slow-Fast Quasilinear Dynamical Systems
Alessia Ferraro, EPFL

Thursday, August 11

Combining Physical and Data-driven Methods to Improve Historical Earth Surface Temperature Estimates
Duo Chan, Woods Hole Oceanographic Institution

Friday, August 12

A Stochastic Dynamical Perspective on Optimizing Learning for Climate Applications
Sai Ravela, Massachusetts Institute of Technology

FELLOWS' PRESENTATIONS

Monday, August 22

Continental Shelf Waves around a Pseudo-Iceland
Ruth Moorman, California Institute of Technology

Fun with Squishy Balls: Theory and Experiments on Deformable Porous Media
Tilly Woods, University of Oxford

Tuesday, August 23

Understanding Invariant Solutions of the Korteweg-de Vries Equation
Claire Valva, New York University

Equatorial Ocean Dynamics on Enceladus Driven by Ice Topography
Rui Yang, University of Twente

Wednesday, August 24

Stochasticity of Turbulence
Iury Simoes-Sousa, University of Dartmouth

Statistical Analysis of Multidimensional Dynamical Systems
Ludovico Giorgini, Stockholm University

Thursday, August 25

Experiments on the Instability of Buoyancy-driven Coastal Currents
Samuel Levin, University of Cambridge

Scaling with the Stars: Emergence of Self-organized Criticality in low Péclet Flows
Kasturi Shah, Massachusetts Institute of Technology

GFD 2022 Lecture 1: Review of Data Decomposition with Linear Algebra

Peter Schmid; notes by Ruth Moorman and Ludovico Giorgini

1 Introduction

Not every evolution process can be modelled easily and successfully by deriving systems of equations from first principles. While we can model many observed phenomena within the field of geophysical fluid dynamics (GFD) using quintessential processes with known governing equations (e.g., advection, diffusion, dispersion, wave propagation, growth and decay of instabilities, cascades of spatio-temporal scales, and so on), these are often insufficient or impractical. In such cases we have to deduce either model equations or emerging coherent patterns, or a combination of both, by processing observable data. Figure 1 provides a conceptual illustration of this process, highlighting some of the types of information we may extract from data to guide our interpretation of the processes and mechanisms governing a given system. The algorithms used to extract physically useful information from data, as well as the application of these algorithms to GFD problems, are the focus of this year's lecture series.

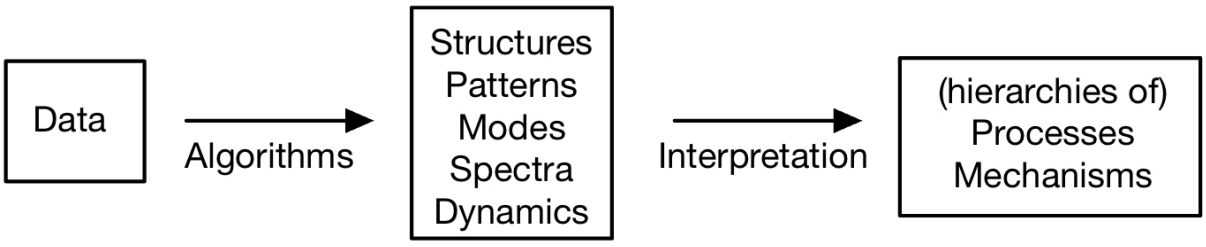


Figure 1: A schema of the course rationale.

Lectures 1 to 3 will focus on **dimensionality reduction**, algorithms used to identify the "essence" of our data by decomposing it into dominant structures that comprise the bulk of the variability in the studied system. In Lectures 3 to 5 we move on to the practice of approximating **dynamical operators** from data, using algorithms to help determine from data how systems propagate through time. Later in the course we reframe these procedures as **optimization** problems in the presence of uncertainty or in situations where external constraints on our solutions are desirable.

2 Data Categorization

Many of the algorithms used in data decompositions assume certain characteristics of the input data, and we need to be cognizant of these assumptions to ensure our processing and interpretations are appropriate.

The following data characteristics will be referenced throughout this course:

Stationarity and Homogeneity: Data is 'stationary' if its statistical moments are constant in time. 'Non-stationary' data has, by contrast, time-varying statistical moments. If the evolution direction relevant for our analysis (see §3.1) is space, rather than time, we would equivalently describe our data as 'homogeneous' or 'inhomogeneous' depending on the variability of its statistical moments in space.

Ergodicity: Ergodocity refers to the equivalence of averaging operations in space and time. Temporally averaging a time series of an 'ergodic' system should return the same result as spatially averaging a snapshot of the system. In other words, the range of variability existing in time is captured within a single snapshot. For example, a *global* Sea-Surface Temperature (SST) time series may be ergodic, but a *regional* SST time series from the mid-latitudes would not, unless a climatology is removed in a preprocessing step. Another way of conceptualizing ergodicity is to say that an ergodic system visits all of its possible states, covering its entire phase space.

Multi-Scale: A 'multiscale' system contains processes at vastly different spatial or temporal scales. One example would be systems with a slow hydrodynamics coupled to fast chemical reactions.

Structure: Structured data is a reference to the spatial configuration of the measurements comprising our data. 'Structured' data is defined on a regular lattice, with a uniform number of nearest neighbors and clear directivity between the neighboring data points. For 'unstructured' data, neighborhoods of points and relational distances need to be defined explicitly, and are heterogeneous throughout the dataset.

Eulerian vs. Lagrangian Data: In fluid dynamics, we distinguish between Eulerian and Lagrangian frames of reference, where the Eulerian frame adopts the viewpoint of a fixed-in-space control volume, while the Lagrangian frame follows particles along a trajectory. Algorithms for analyzing data observed according to an Eulerian or Lagrangian perspective differ greatly.

3 Data Preprocessing

3.1 Constructing the data matrix

Throughout this lecture series we will consider data in the following form,

$$D = \begin{bmatrix} | & | & & | \\ \mathbf{d}_1 & \mathbf{d}_2 & \dots & \mathbf{d}_m \\ | & | & & | \end{bmatrix} \in \mathbb{C}^{n \times m} \tag{1}$$

where

n = number of state vector components, and
m = number of data instances along the evolution direction.

The $\mathbf{d}_i$ ($i \in [1, m]$) represent individual instances of data along its **evolution direction**. This evolution direction may be time, such that the $\mathbf{d}_i$ represent spatial snapshots of a temporally evolving system, or space, such that the $\mathbf{d}_i$ represent the temporal or a mixed spatio-temporal snapshot of the system at a fixed location in space along an evolution direction. **Throughout this lecture series we take the evolution direction to be time unless otherwise stated**, however, the algorithms presented here are general and may be used to analyse data with other evolution directions, in which case the language and interpretation may differ somewhat but the algebra remains the same.

The $\mathbf{d}_i$ may be high dimensional, for example they may contain numerous observed quantities (or 'composite data') in two or three spatial dimensions, however they must be stacked or vectorized in a sorted manner in D such that each $\mathbf{d}_i$ is a state vector with n components. It is important to respect the ordering in this flattening process; while there is flexibility in how to shape the measurements into a one-dimensional array, the ordering has to be consistent from state vector to state vector. If our data does not easily conform to this constraint (e.g., if our data is Lagrangian and thus the position associated with each measurement evolves in time) we must attach a location information to each state vector.

3.2 Centering and scaling the data matrix

It is often necessary to calibrate our data in various ways prior to decomposing it. Such calibrations may be achieved via either post-multiplication or pre-multiplication of our data matrix D with some preprocessing matrix P,

$$D' = DP \qquad\qquad D' = PD.$$

Post-multiplication scales the data along the evolution direction. An important example of post-multiplication is data centering or the removal of the mean value of each state vector component along its evolution direction. This is implemented as,

$$D' = DP, \qquad P = I - \frac{1}{m} 1 1^T \tag{2}$$

where I is the identity matrix and 1 is a column vector of ones. Removal of the mean before data decomposition can be important for reasons discussed in §5.

Pre-multipication scales the data within each state vector or snapshot. This is frequently used to ensure that measured variables in composite data sets are not unduly prioritized by decomposition algorithms due to their absolute magnitude. For example, aeroacoustic systems generate both hydrodynamic and acoustic data, and the hydrodynamic variables are often substantially larger in magnitude than the pressure readings representing the acoustic field. In the absence of scaling during preprocessing, decompositions would typically overemphasize the hydrodynamic component of the system and discount the acoustic processes. Rescaling permits a more even consideration of these components.

Another instance of preprocessing is the expression of the data sequence in a reduced basis. This expression is accomplished by a projection of the entire data set onto a subspace spanned by the identified and user-supplied structures, and is implemented according to

$$D' = PD, \qquad P = VV^T \tag{3}$$

where V represents the column space onto which we want to project. There are various options for the structures included in V depending on the chosen method of scaling. For example, we may scale

each variable such that it has zero mean and unit variance, match the minimum and maximum values of each variable, set the median of the data to zero and constrain its interquartile range, and so on.

4 Data Decompositions

In their most general form, decompositions of matrices are the equivalent of factorizations of numbers into more elementary components. Just as there are a great many ways of factorizing numbers, there are as many ways of breaking apart a matrix into components, and each decomposition is identified by the type of factors it produces. Most of the algorithms discussed in this series are three-factor decompositions (§4.2), which are better suited to analysis, but for context we will start by describing two-factor decompositions (§4.1), which are usually used for data management.

4.1 Two-factor decomposition

Two-factor decompositions take the general form,

$$D = AB, \quad \text{where} \quad A \in \mathbb{C}^{n \times k} \quad \text{and} \quad B \in \mathbb{C}^{k \times m}. \tag{4}$$

Generally we have more information within each state vector than we have snapshots, i.e., $n > m$. Given this, two special cases of interest involve setting $k = m$, which yields a 'reduced' decomposition, and $k = n$ which produces a 'full' decomposition. Reduced decompositions reorganize the subspace spanned by the columns of D, whereas a full decomposition also describes the null-space of D, i.e., the dynamics orthogonal to those contained in D.

In its general form the two-factor decomposition above is not useful, we need to impose a constraint on the nature of A or B to extract a unique decomposition. One classic constraint is to require that A is orthogonal. This produces the QR-decomposition, which breaks down D into its orthonormal basis, Q, and an upper triangular matrix of coefficients in this basis, R. Another example is to require that A is positive definite. This produces the polar decomposition, which expresses D in terms analogous to the expression $re^{i\theta}$ by breaking down D into a positive definite matrix (equivalent to the radius r) and a rotation matrix (equivalent to the phase-part $e^{i\theta}$).

4.2 Three-factor decompositions

The general form of a three-factor decomposition is,

$$D = ABC, \quad \text{where} \quad A \in \mathbb{C}^{n \times k}, \quad B \in \mathbb{C}^{k \times k} \quad \text{and} \quad C \in \mathbb{C}^{k \times m}, \tag{5}$$

with $n > m$. Again, in this series we will focus on reduced decompositions ($k = m$), though full decompositions ($k = n$) may be important in other contexts. The above decomposition is often reformulated to stress that this three-factor decomposition is akin to an input-output analysis,

$$DC^{-1} = AB. \tag{6}$$

This formulation clarifies that we are applying the data matrix D on an input basis C^{-1}, resulting in an output basis A multiplied by a deformation matrix B. It is important to note that all three-factor decompositions assume stationary data (or homogeneous data, if the evolution direction is space).

We need to impose two constraints on (5) to obtain a unique decomposition. Many constraints, and many combinations of constraints, are possible, but enforcing any two of the following three constraints generates three common and illustrative decompositions:

1. Diagonality of B,
2. Equality of A and C, (sometimes framed as equality of the input and output basis) and
3. Orthogonality of A and/or C (note that sometimes orthogonality of one implies both).

Case 1: Eigenvalue Decomposition

If we enforce constraints 1 and 2 (i.e., B is diagonal and $A = C$) we arrive at the **eigenvalue decomposition**,

$$D = \Lambda V \Lambda^{-1}, \tag{7}$$

where Λ and V are square matrices. Note that whilst it is true that in some special cases we may simultaneously satisfy conditions 1, 2, and 3, generally, V is non-orthogonal.

Case 2: Singular Value Decomposition (SVD)

If we enforce constraints 1 and 3 (i.e., B is diagonal and A and/or C are orthogonal) we arrive at the **singular value decomposition**,

$$D = U \Sigma V^H, \tag{8}$$

where U and V are different but both are orthogonal, Σ is a diagonal matrix containing the 'singular values', and H denotes the Hermitian transpose. Since the equality constraint (number 2) is not enforced, SVD may be applied to any rectangular matrix.

Case 3: Schur Decomposition

If we enforce constraints 2 and 3 (i.e., A and C are equal and orthogonal) we arrive at the **Schur decomposition**,

$$D = UTU^H, \tag{9}$$

where U and T are square matrices. U is orthogonal, but T will generally not be diagonal, but, at best, be upper triangular.

Note that these three canonical decompositions are equivalent for normal matrices, i.e., matrices that commute with their transpose. For non-normal matrices, they are distinct. Whilst the eigenvalue and Schur decompositions are important in many contexts, the SVD is the most versatile and widespread decomposition for data analysis, and we spend the rest of the first lecture introducing the SVD and discussing its usage.

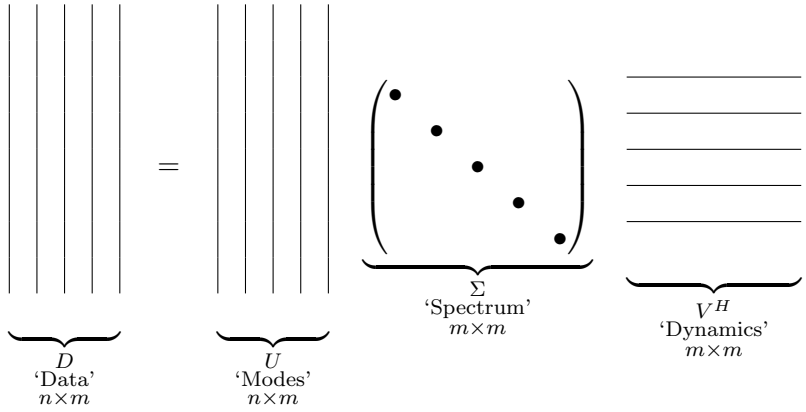
5 SVD for Spectral Analysis

Any triple decomposition with a diagonal scaling matrix B can be thought of as a spectral decomposition of the data. The matrix A contains the basis (the 'modes'), the matrix B contains

the amplitudes (the 'spectrum'), and the matrix C contains the evolution of a given mode in the evolution direction (the 'dynamics'). In the case of the SVD, this looks like,

$$\begin{array}{ccccccc} D & = & U & & \Sigma & & V^H \\ \text{'Data'} & = & \text{'Modes'} & & \text{'Spectrum'} & & \text{'Dynamics'} \end{array} \tag{10}$$

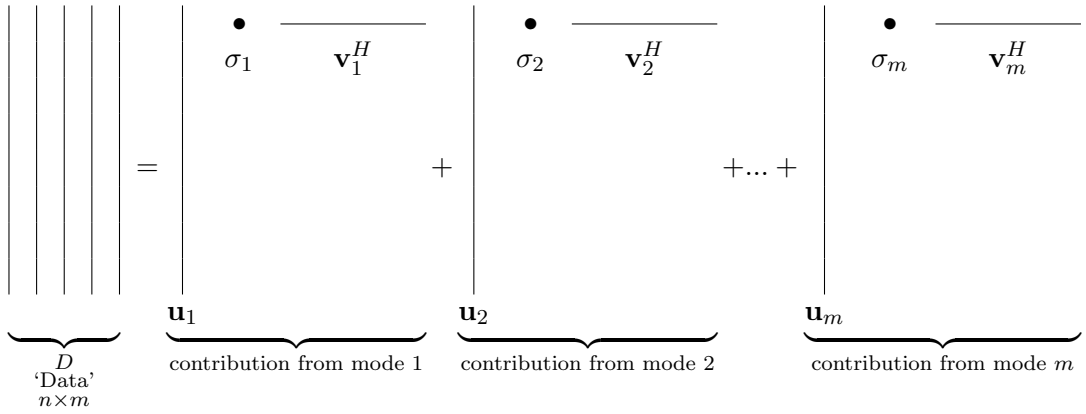
or, pictorially,



(11)

for data containing $m = 5$ system snapshots and $n > m$ measurements per snapshot. The SVD algorithm will construct Σ such that the singular values σ_i are sorted according to descending magnitude (i.e., $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m \geq 0$). The algorithm will also, by design, enforce that U and V are orthogonal matrices, with orthonormalized columns.

We can visually break down (11) further to highlight that our data matrix D has been decomposed into a sum of m rank-1 matrices of the form $\mathbf{u}_i \sigma_i \mathbf{v}_i^H$,



(12)

This formulation helps us see that the $\mathbf{u}_i$ comprise a set of orthogonal structures or 'modes' that each explain a decreasing portion (σ_i) of the spatio-temporal variability in our full data sequence. The $\mathbf{v}_i$ then contain the time evolution of the ith mode's contribution to the full time series, or the 'dynamics' associated with that mode, expressed as a variation about its mean value σ_i.

The outcome of this decomposition is sensitive to our choices during data preprocessing. If known structures with considerable power (e.g., time mean fields, linear trends, seasonal cycles) are not removed from D prior to decomposition, these structures will likely make up the most

important mode $\mathbf{u}_1\sigma_1\mathbf{v}_1^H$ and, due to orthogonality constraints, restrict the possible forms of lesser modes. In other words, the most powerful mode of variability remaining in D after preprocessing has a cascading influence on the entire decomposition. We must be cognizant of this and test the sensitivity of our identified modes to preprocessing choices. The decision to remove powerful modes is somewhat subjective, however, as noted in (§3.2), data structures containing numerous variables should always be normalized prior to decomposition, or else the relative absolute magnitudes of variables will artificially inflate or deflate their importance.

Once our data has been **decomposed** it may then be **reduced** by exploiting the sorted structure of the spectrum Σ. Note that if the singular values σ_i are all non-zero, the matrix D will have rank m. If, on the other hand, some of the singular values σ_i are zero, the matrix D will be rank-deficient and all matrices in our decomposition may be compressed to exclude structures with zero amplitude. If some of the singular values σ_i are small but non-zero, we may choose to approximate the matrix D by a lower-rank approximation according to,

$$\hat{D} \approx U_{:,1:r}\,\Sigma_{1:r,1:r}\,V^H_{1:r,1:r} = \mathbf{u}_1\sigma_1\mathbf{v}_1^H + \mathbf{u}_2\sigma_2\mathbf{v}_2^H + \ldots + \mathbf{u}_r\sigma_r\mathbf{v}_r^H, \tag{13}$$

where $r < m$. Essentially, we can approximately reconstruct our data using only its r most powerful modes. The choice of r is subjective but may be informed by the structure of the Σ matrix. If the magnitude of the σ_i terms drop off rapidly after some index i, that index may be a good choice for r (see Figure 2). We should take care here, this method of data reduction assumes that small amplitude modes are unimportant, which is not necessarily the case in many applications.

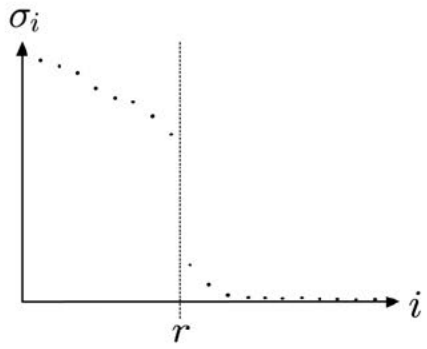


Figure 2: An example of a Σ matrix with a clearly justifiable choice of r index for truncation.

,

A SVD by any other name...

A plethora of field specific terms are used to refer to the SVD algorithm, here we have a (non-exhaustive) list of terms that you may come across in the literature (and in this lecture series...):

Principal Component Analysis (PCA)
Proper Orthogonal Decomposition (POD)
Empirical Orthogonal Functions (EOFs)
Karhunen-Loéve Decomposition
Hotelling Transform

These all refer to the same procedure!

GFD 2022 Lecture 2: Spatio-temporal Decomposition of Timeseries

Laure Zanna; notes by Sam Lewin and Kasturi Shah

"The most that can be expected from any model is it can supply a useful approximation to reality. All models are wrong, some models are useful."
- George E. P. Box (1987 & 2005)

"The purpose of models is not to fit the data but to sharpen the question."
- Samuel Karlin (1983)

1 Motivation for Data-driven Climate Dynamics

Following the data decomposition from linear algebra in Lecture 1, we will consider three examples from climate dynamics:

- Consider timeseries of sea surface temperatures over the globe in Figure 1a: what do we notice? An immediately evident and predictable feature is the seasonal cycle. We also notice regional features, such as the meanders of the Gulf Stream, or the growth of sea ice in polar regions.
- Now, consider sea surface height maps with the mean removed in Figure 1b: what do we notice? A striking feature are the equatorial waves, as well as mesoscale eddies at midlatitudes in the Gulf Stream.
- Finally, consider paths of the Gulf Stream coloured by the kinetic energy at that point in time and space calculated from geostrophic velocities in Figure 1c. What do we notice? The kinetic energy increases as the Gulf Stream detaches and then decreases. Each year's path and meanders are distinct; when the Gulf Stream meanders a lot, it breaks a barrier to mixing.

These examples illustrate that we can deduce information and patterns from looking at data, begging the question: how can we extract patterns and features from data in a robust way?

2 Principal Component Analysis

Principal Component Analysis (PCA) is the "bread and butter" of data-driven and machine- learning techniques. Invented in 1901 by Karl Pearson, it is a remarkable example of a robust and useful technique that has withstood the test of time, despite some limitations. As discussed in Lecture 1, PCA requires stationary data to pick out patterns and features in the data set. PCA is helpful to aggregate data, reduce redundancies and keep only "useful" data, i.e., it is a dimensionality reduction tool.

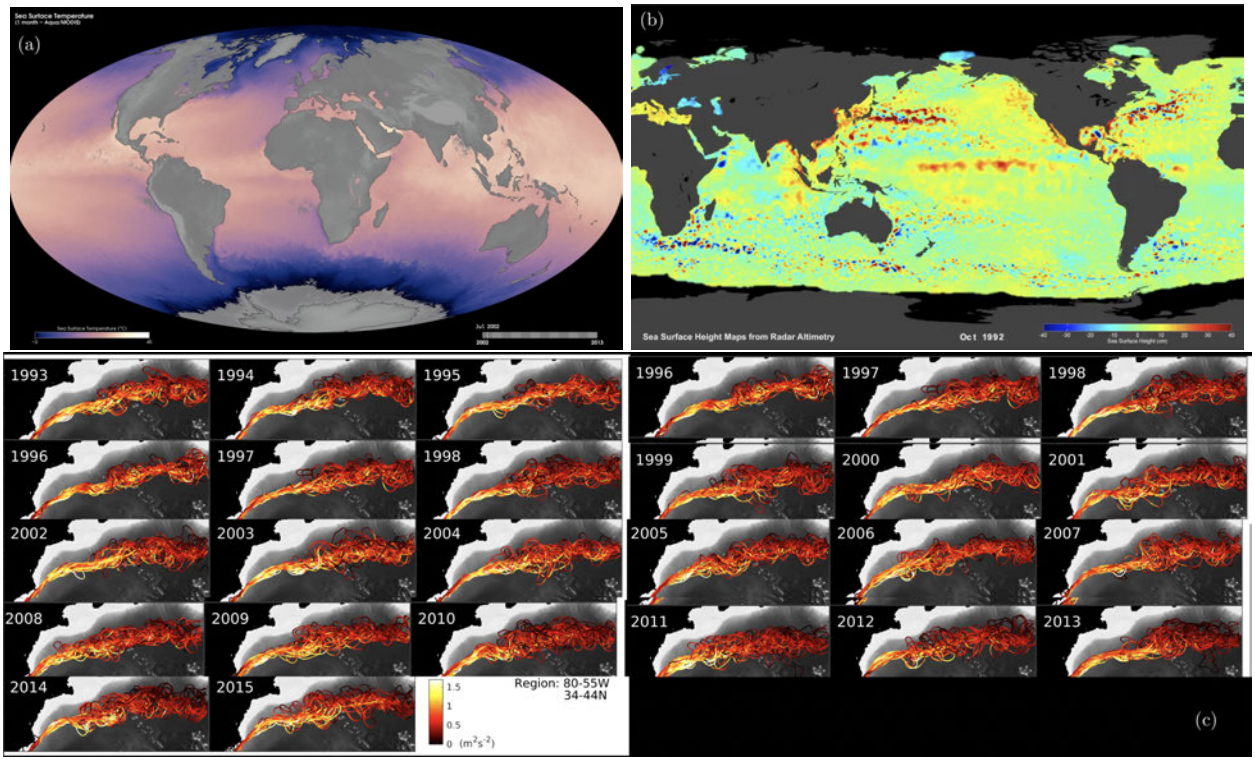


Figure 1: Illustrative examples of climate dynamics as a motivation for pattern extraction-based data-driven techniques. (a) A global map of sea surface temperatures in July 2002 from Aqua/MODIS. (b) A global map of sea surface height anomalies from radar altimetry in October 1992. (c) Paths of the Gulf Stream calculated every two weeks from Aviso satellite altimetry data. Each curve is a continuous contour of Absolute Dynamic Topography (ADT) colocated with the maximum ADT gradient and coloured by the kinetic energy calculated from geostrophic velocities [2].

2.1 Fundamentals of Principal Component Analysis

Consider a dataset D,

$$D_{n\times m} = \begin{bmatrix} | & | & & | \\ \mathbf{d}_1 & \mathbf{d}_2 & \dots & \mathbf{d}_m \\ | & | & & | \end{bmatrix} \tag{1}$$

where spatial evolution is described by the rows and temporal evolution by the columns, such that each column $\mathbf{d}_i$ represents a timeslice. We are looking for a set of orthogonal vectors $\mathbf{u}_j$ that describes the temporal variability of the system. These vectors from the matrix U,

$$U_{n\times n} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_n \\ | & | & & | \end{bmatrix}. \tag{2}$$

We now look for $\mathbf{u}_1$ that maximises the variability $\sum_{k=1}^{m} (\mathbf{u}_1 \cdot \mathbf{d}_k)^2$. Here, $\mathbf{u}_1$ is generally similar to the typical patterns of the columns $\mathbf{d}_m$ of D. We then look for each subsequent eigenvector $\mathbf{u}_j$ such that the variability $\sum_{k=1}^{m} (\mathbf{u}_j \cdot \mathbf{d}_k)^2$ is maximum and orthogonal to $\mathbf{u}_{j-1}$. In general, therefore, we are looking for

$$\max\left[\frac{1}{m}\sum_{k=1}^{m} (\mathbf{u}_j \cdot \mathbf{d}_k)^2\right] = \mathbf{u}_j^T \frac{1}{m} D D^T \mathbf{u}_j = \mathbf{u}_j^T C \mathbf{u}_j, \tag{3}$$

where the covariance matrix C is

$$C = \frac{1}{m} D D^T, \tag{4}$$

C is symmetric and each vector is orthogonal. Drawn schematically,

$$C_{n\times n} = \begin{pmatrix} \diagdown \end{pmatrix} \tag{5}$$

where the diagonal entries are the variance and the off-diagonal entries are the covariance, i.e., they describe how each point jointly varies with every other point. If C has a lot of off-diagonal entries, this suggests a lot of the data is redundant (as they covary) and can be discarded.

We seek to maximize $\mathbf{u}_j^T C \mathbf{u}_j$ subject to the L2 norm of $\mathbf{u}_j$ being 1, i.e., $||\mathbf{u}_j|| = 1$, using Lagrange multipliers. We differentiate with respect to $\mathbf{u}_j$, which leads to

$$\frac{\partial}{\partial \mathbf{u}_j}\left[\mathbf{u}_j^T C \mathbf{u}_j - \lambda_j(\mathbf{u}_j^T \mathbf{u}_j - 1)\right] = 0 \tag{6a}$$

and on taking the derivative,

$$2C\mathbf{u}_j - 2\lambda_j \mathbf{u}_j = 0 \tag{6b}$$

which is an eigenvalue problem

$$C\mathbf{u}_j = \lambda_j \mathbf{u}_j \tag{6c}$$

where $\mathbf{u}_j$ are the eigenvectors of C with eigenvalues λ_j. The eigenvalues λ_j indicate the amount of variance contained by each mode. This fraction of the variance is given by

$$\frac{\lambda_j}{\sum_j \lambda_j} \tag{7}$$

where $\sum_j \lambda_j$ is the total variance, i.e., the trace of C.

Having now attained the spatial patterns, we turn our attention to the temporal evolution of the eigenmodes, which we calculate by taking the projection of the patterns onto the data matrix, such that

$$U^T_{n\times n} D_{n\times m} = V_{n\times m} \tag{8}$$

where $\mathbf{v}_j^T$ gives the temporal evolution of the eigenvector $\mathbf{u}_j$.

In summary,

$$D = UV \tag{9}$$

where D is our dataset, U contains the spatial information and V contains the temporal information.

We conclude this section with a few important notes. First, everything depends on the data matrix D one starts with. This begs the question of how to deal with missing data, a topic that will be covered in a subsequent lecture. In general, however, if we have a lot of missing data, the covariance matrix C can be degenerate or messy and while one could conduct a PCA analysis, the results must be carefully interpreted. Second, the first eigenvector $\mathbf{u}_1$ will describe the maximum temporal variance. Therefore, it is not essential to detrend the data, as the first eigenvector will pick out said pattern. Third, the modes obtained from PCA are *not* the same as the modes of the underlying dynamical system (see §2.4 for a cautionary tale). When dealing with a dynamical system where the system's modes propagate in both space and time, the PCA analysis can be adapted to obtain the spatial propagation of the eigenmodes. For instance, one can choose the evolution pathway as spatial (e.g., the direction of propagation of the Gulf Stream). Alternatively, one can consider complex PCs where the imaginary part is the Hilbert transform. Otherwise, the temporal modes from the PCA also dissipate which may indicate spatial variations.

2.2 Link to Singular Value Decomposition

We can use SVD algorithms to do PCA without the computational expense of having to explicitly calculate the entire covariance matrix C, which could be very high-dimensional. To this end, consider writing

$$\overbrace{D}^{\dim\ n\times m} = \underbrace{U}_{n\times n} \overbrace{\Sigma}^{n\times m} \underbrace{\tilde{V}^T}_{m\times m} \tag{10}$$

Note that this is the *full decomposition*, which is the same as the reduced decomposition from the previous lecture, except that

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \dots & 0 \\ 0 & 0 & \ddots & \vdots \\ 0 & \cdots & \cdots & \sigma_m \\ 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}, \tag{11}$$

where the last $n-m$ rows are zeros (remember we are assuming that $n > m$). This is mathematically the same as the result from §2.1 (i.e., the principal components are given by the columns of U) if we identify

$$V^T = \Sigma \tilde{V}^T. \tag{12}$$

However, the numerical procedures for doing the decomposition are quite different.

2.3 Example: PCA analysis for ENSO

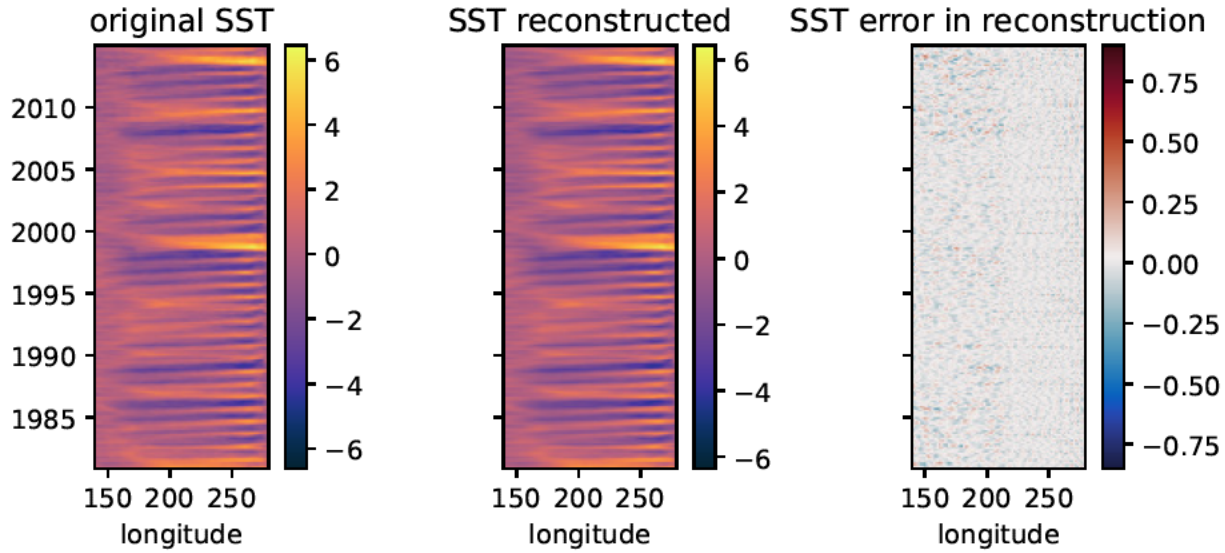


Figure 2: PCA reconstruction of the equatorially averaged SST timeseries. The true time-series is shown in the left-hand panel, whilst a reconstruction obtained using 20 principal components is shown in the center panel. The error is shown in the right-hand panel.

The El Niño-Southern Oscillation (ENSO) is a quasi-oscillatory behaviour of sea-surface temperatures (SSTs) in the central and eastern Pacific Ocean with warming and cooling phases that strongly affect regional weather patterns. It can clearly be observed by plotting the time series of SST latitudinally averaged over the equatorial region, as shown in the left-hand panel of Figure 2.

In this case, the formation of the data matrix D is fairly instructive: $m = 141$ is the number of state vector components (i.e., the spatial dimension: one for each degree of longitude), whilst n is the number of time steps (in this case we take one time step to be one month). By performing SVD on this matrix and reconstructing the data from the first 20 principal components, we obtain the center panel in Figure 2. Qualitatively we can see that the first 20 principal components broadly capture the important features of ENSO. Errors are plotted in the right hand panel of Figure 2 highlighting the reasonable agreement.

We can also do this for a 2-dimensional (2D) latitude-longitude SST field. To form the data matrix D we now need to flatten the image for each time step into a single column with dimensions equal to the number of grid-points/pixels in the 2D field. We can then do PCA as before and look at the spatial structure of the principle components, as well as the evolution of the associated time series. Results are shown in Figure 3. There is a clear structure corresponding to the first principal component whose time evolution experiences high variability over periods of a few years.

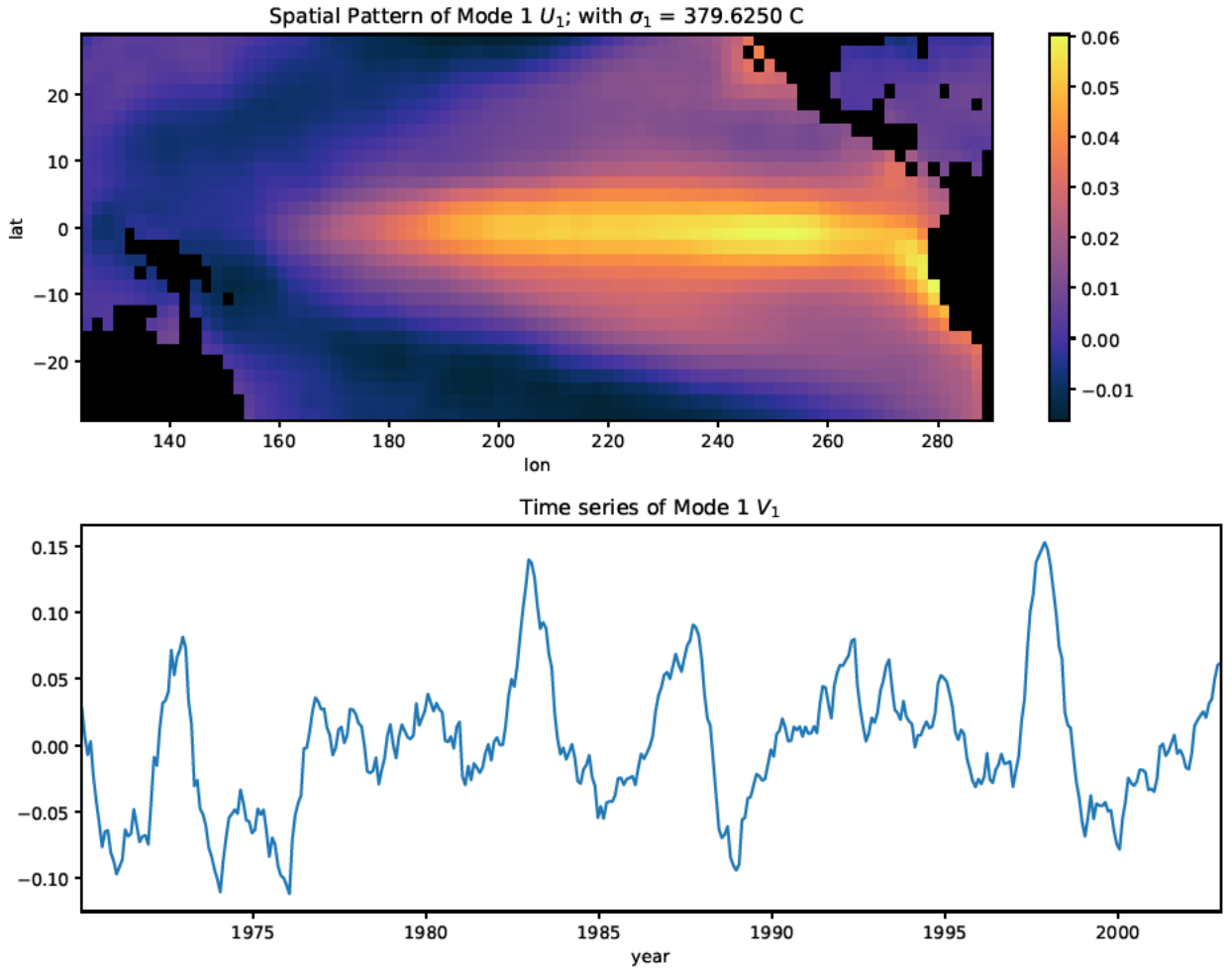


Figure 3: First principal component and associated time series for the PCA of a 2D SST field around the central and eastern equator.

2.4 A cautionary example for linearised systems

Consider the two-dimensional dynamical system

$$\frac{d}{dt}\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = A\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} -0.1 & -0.9\cot\phi \\ 0 & -1 \end{pmatrix}\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} \tag{13}$$

Since the system is already linear, we can compute the eigenvectors $\{\mathbf{b}_1, \mathbf{b}_2\}$ and corresponding eigenvalues $\{\lambda_1, \lambda_2\}$ directly:

$$\mathbf{b}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix},\ \lambda_1 = -0.1; \quad \mathbf{b}_2 = \begin{pmatrix} \cos\phi \\ \sin\phi \end{pmatrix},\ \lambda_2 = -1. \tag{14}$$

In general these eigenvectors are non-orthogonal (the angle between them is just ϕ). But note that the EOFs, which are the eigenvectors of the covariance matrix C, are necessarily orthogonal by construction, so *the EOFs are not the same as the eigenvectors of a corresponding linear system.* A necessary condition for the existence of a set of normal eigenmodes to the linear system is that A be a normal matrix, i.e., $AA^T = A^TA$ [1].

3 Discussion

PCA is a powerful tool for dimensionality reduction, enabling efficient noise removal and compression of complex and high-dimensional datasets, as well as informative visualisation. The decomposition $D = UV$ allows us to construct reduced order linear 'dynamical systems' associated with the temporal evolution of spatial modes (encoded in the rows of V) which might provide useful physical insight. However, as has been alluded to throughout, there are some limitations which are summarized below.

- The underlying assumption of stationarity clearly restricts the sorts of datasets we can use PCA on.
- We are assuming that the principle components are a linear combination of the original data, which may be somewhat restrictive.
- The spatial modes that result from PCA are assumed to be orthogonal: this may not best capture the variability.
- As discussed in §2.4, there is no general link between principal components and the eigenvectors of an appropriate corresponding linearised dynamical system. Even though we can recover a reduced order linear dynamical system for the spatial modes, since these modes are usually some complex combination of the original observables it is often difficult to interpret what this system represents physically.
- Perhaps not necessarily a limitation, but it is worth noting that the data needs to be sufficiently prepared for the algorithm to work as intended: for example, detrending, rescaling, dealing with missing data points, removing outliers etc.

References

[1] Farrell, B.F. and Ioannou, P.J 1996. Generalised stability theory. Part I: Autonomous operators. *J. Atmos. Science*, 53 (14), 2025-2040.

[2] Bolton, Thomas 2019. A data-driven investigation into the behaviour and parameterisation of mesoscale eddies. *PhD Dissertation*, University of Oxford.

GFD 2022 Lecture 3: Transfer Operator for Data Analysis (Part 1)

Peter Schmid; notes by Iury Simoes-Sousa and Tilly Woods

1 Introduction

This lecture comes in two parts. Firstly, we will look at **ensemble averaging** as a method for extracting information from rare or intermittent events in a dataset. That will conclude the first part of the lecture series on **dimensionality reduction**. We will then move on to the second part of the lecture series, covering the approximation of **dynamical operators** from data. In this lecture, we will focus on one such operator: the **Koopman operator**.

2 Ensemble Averaging

The following technique of ensemble averaging will be particularly useful for gaining insight into data sets capturing rare or intermittent events, such as acoustic bursts in high speed jets [1] or intermittent vortex shedding in the ocean. The correlation matrix in these cases is created by taking the mean over statistically independent samples.

$$C = \mathbb{E}_e[\, \mathbf{d}(\mathbf{x}, t)\, \mathbf{d}(\mathbf{x}', t')\,] \tag{1}$$

which $\mathbb{E}_e$ is the expected value over the product of independent realizations. Suppose we have some spatio-temporally inhomogeneous data displayed in a matrix

$$D = \begin{bmatrix} | & | & ... & | \\ \mathbf{d}_1 & \mathbf{d}_2 & ... & \mathbf{d}_m \\ | & | & ... & | \end{bmatrix}, \tag{2}$$

where each column is our set of data points at a given snapshot in time, with time evolving from left to right.

Let's assume we have rare events that are captured by a point in the space. We then assume that each rare event is an independent sample delimited by a fixed window around the peaks (Figure 1). Once we define the window length and select the samples, we embed the time dependence in a new data matrix in the size of $(l \times n) \times k$, which l is the number of snapshots within each sample, n is the number of spatial points and k is the number of samples. In other words, we stack in the vertical each snapshot within the same sample (blue lines in Figure 1b) and stack in the horizontal each event. Samples can overlap each other in time if we have events happening very close to each other, but all windows must have the same number of snapshots and each window must only contain one rare event.

$$D = \begin{bmatrix} d(t_1 - \Delta t) & d(t_2 - \Delta t) & \cdots & d(t_k - \Delta t) \\ \vdots & \vdots & \cdots & \vdots \\ d(t_1 + \Delta t) & d(t_2 + \Delta t) & \cdots & d(t_k + \Delta t) \end{bmatrix}, \tag{3}$$

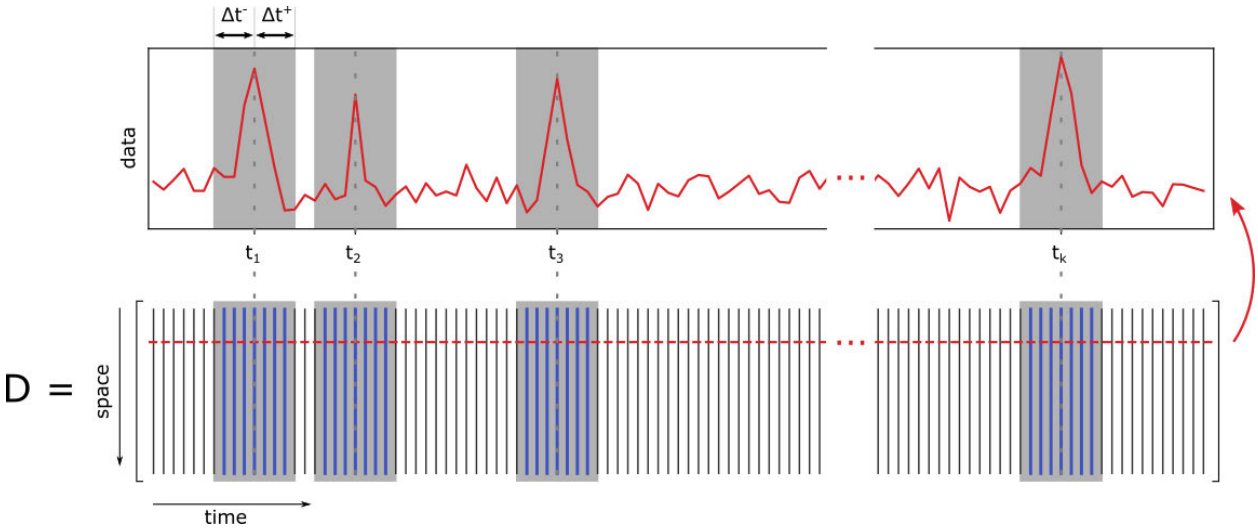


Figure 1: Sampling method for the spatio-temporal EOF. The time window is defined based on the characteristic time-scales of one of the timeseries, and the other data points are used to form the correlation matrix.

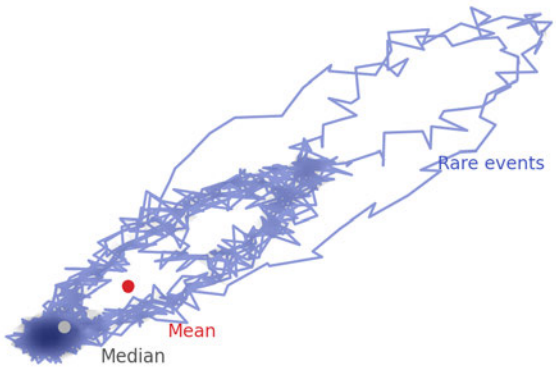


Figure 2: Schematic representation of rare events in a 2D phase space.

The corresponding correlation matrix $C = DD^H$ averages over the samples and the resulting EOFs have both a spatial and temporal structure. In other words, a single EOF is $(l \times n)$ long that could be unstacked into l and n, thus having both a spatial and temporal structure.

We can understand this system as a Gaussian distribution (close to the median) in the 2D phase space overlapped by some rare events that displace the mean (Figure 2). We basically kept and rearranged each of the rare events into a new matrix to perform the EOF analysis.

The statistical independence is not the only way to define a sample within an arbitrary time-series. We can also define the ensemble correlation matrix by taking the mean over samples that satisfy an arbitrary condition H.

$$C = \mathbb{E}_e[\,\mathbf{d}(\mathbf{x}, t)\,\mathbf{d}(\mathbf{x}', t')\,|\,H] \tag{4}$$

This ensemble average method is extremely useful for extracting information about rare events, which the signal would be masked using common SVD analysis.

3 Summary of Dimensionality Reduction

This concludes the first part of the lecture series on dimensionality reduction, where the aim has been to extract patterns from data by identifying the structures that contribute the greatest variability to the system that generated the data. In other words, we have studied techniques for finding and keeping only the most useful aspects of a dataset. The main technique for doing this is Principle Component Analysis (PCA, Lecture 2) - also known by many other names (see Lecture 1) - which is basically a Singular Value Decomposition (SVD, Lecture 1). These powerful tools will be used repeatedly in the remainder of the lecture course.

4 Transfer Operators: Koopman Operators

We now move on to the second part of the lecture series in which we focus on the approximation of dynamical operators from data. Instead of factorising the data matrix D to get the patterns U, amplitudes Σ and dynamics V^{H}, we can try to find a linear operator, called a transfer operator, to map from one snapshot in time to another. There are two types of transfer operators:

- the Koopman (forward) operator maps from one snapshot to the next. (Where do we go to next?)
- the Frobenius-Perron (backward) operator maps from one snapshot to the previous. (Where did we come from?)

These two operators are adjoints (inverse) of one another.

In this lecture, we focus on the Koopman operator. We suppose there is a nonlinear map F that maps the state vector $\mathbf{d}_i$ at one snapshot in time $t_i = i\Delta t$ to the state vector $\mathbf{d}_{i+1}$ at the next snapshot in time $t_{i+1} = (i+1)\Delta t$,

$$\mathbf{d}_{i+1} = F(\mathbf{d}_i), \tag{5}$$

where the state vectors are the columns of the data matrix D. To gain insight into the behaviour of this nonlinear mapping, we would like to reduce the problem to a linear system that we can more easily analyse. Traditionally, this has been done using the Poincaré method, which is to

1. Find the equilibrium states of the nonlinear system.
2. Linearise the system about the equilibrium states.
3. Analyse the linear dynamics (limit cycles, Poincaré maps, bifurcations etc.)

Koopman proposed an alternative approach centred around data rather than the nonlinear mapping. The aim is to choose observables $\varphi(\mathbf{d})$ (which are functions of the state variables) such that the nonlinear dynamical system becomes linear when written in terms of the observables. The Koopman method is to

1. embed the dynamical system high-dimensionally in an observable space, i.e., choose observables that linearise the system;
2. find the linear mapping between observables at one snapshot in time and the next;
3. analyse the linear mapping.

That is, we want to find suitable observables $\varphi(\mathbf{d}_i)$ such that there is a linear operator $\mathcal{K}$, called the Koopman operator, satisfying

$$\mathcal{K}\varphi(\mathbf{d}_i) = \varphi(\mathbf{d}_{i+1}) = \varphi(F(\mathbf{d}_i)). \tag{6}$$

To make more sense of this abstract description of the Koopman method, we look at a couple of examples. Firstly, an example of what we mean by an observable is the shadows created by shining a light onto a three-dimensional object. By combining the shadows (observables) produced by shining light from infinitely many angles around the object, we can reconstruct the full three-dimensional object (nonlinear system).

A simple numerical example is the two-dimensional dynamical system

$$\begin{aligned} \dot{x}_1 &= \mu x_1, \\ \dot{x}_2 &= \lambda(x_2 - x_1^2), \end{aligned} \tag{7}$$

where μ and λ are constants. Here, we choose observables

$$\varphi_1 = x_1, \quad \varphi_2 = x_2, \quad \varphi_3 = x_1^2. \tag{8}$$

This choice has been made so that the nonlinear system written in terms of these observables becomes a linear system,

$$\frac{d}{dt}\begin{pmatrix}\varphi_1\\ \varphi_2\\ \varphi_3\end{pmatrix} = \begin{pmatrix}\mu & 0 & 0\\ 0 & \lambda & -\lambda\\ 0 & 0 & 2\mu\end{pmatrix}\begin{pmatrix}\varphi_1\\ \varphi_2\\ \varphi_3\end{pmatrix}. \tag{9}$$

Note that the cost of making our nonlinear system linear is an increase in the dimension of the system. The philosophy of the Koopman method is that it is better to have a large linear system than a small nonlinear system. In general, the linear system produced will be infinite-dimensional, but there are numerical methods to pick a finite number of observables φ_i to approximate the infinite linear system.

Our aim is to use the data we have available to reduce our nonlinear system to a linear system, as in (6). There are two key questions that we need to answer in order to do this:

1. How do we pick the observables φ to make our nonlinear system linear?

2. How do we work out the linear Koopman operator $\mathcal{K}$ from our data?

We will focus on the first question first. One approach for finding suitable observables for a dynamical system formed of polynomial terms is to use polynomials. This is refered to as Carleman linearisation. The idea is to, for example, choose one observable to be a quadratic function if there are quadratic terms in our nonlinear system. This gets rid of the quadratic terms, but might introduce cubic terms in the evolution equation of the quadratic observable. Choosing a cubic observable to remove the cubic terms might introduce quartic terms, and so on. Hence an issue with the Carleman linearisation method is that it can lead to the observables outrunning the observable space, giving a closure problem.

An alternative (better) approach which avoids the closure problem is to choose the observables φ to be invariants of $\mathcal{K}$, so that applying $\mathcal{K}$ to the observables keeps them in the observable space. Therefore, we pick φ to be eigenfunctions of $\mathcal{K}$:

$$\mathcal{K}\Phi = \Phi\Lambda, \tag{10}$$

where Φ is a matrix whose columns are the eigenfunctions $\{\boldsymbol{\phi}_1, ..., \phi_m\}$ of $\mathcal{K}$ and Λ is a is a diagonal matrix of the eigenvalues $\{\lambda_1, ..., \lambda_m\}$ of $\mathcal{K}$. At each snapshot in time, we calculate the value of the observables at that snapshot from the data $\mathbf{d}_i$ at the snapshot, using the notation

$$\varphi_i = \varphi(\mathbf{d}_i) \tag{11}$$

for the value of the observables at snapshot i. Note that both φ_i and $\mathbf{d}_i$ are column vectors, with φ_i (length $\tilde{n}$) being longer than $\mathbf{d}_i$ (length n) since we have more observables than original state variables. Using (6) and (11), we can write our observables φ_i at each snapshot in time in terms of the observables φ_0 at time zero,

$$\varphi_i = \varphi(\mathbf{d}_{0+i}) = \mathcal{K}^i \varphi(\mathbf{d}_0) = \mathcal{K}^i \varphi_0. \tag{12}$$

We can then express the observables φ_0 at time zero in terms of the basis formed of eigenfunctions of $\mathcal{K}$, writing

$$\varphi_0 = \Phi\xi, \tag{13}$$

where ξ is an $m \times 1$ column vector containing the coefficients of the observable in the terms of the basis of eigenfunctions. Hence, substituting (13) into (12), we can express the observables at each snapshot in time as

$$\varphi_i = \mathcal{K}^i \Phi\xi = \Phi\Lambda^i\xi, \tag{14}$$

for $i = 0, 1, ..., m-1$, where the final equality comes from using the eigenequation (10). Next, we use our observables to build a new 'data matrix', but with the columns being observables φ_i instead of state vectors $\mathbf{d}_i$. Call this new $\tilde{n} \times m$ matrix $\tilde{D}$ to distinguish it from our original $n \times m$ data matrix D, where we define

$$\tilde{D} = \begin{pmatrix} | & | & & | \\ \varphi_0 & \varphi_1 & \cdots & \varphi_{m-1} \\ | & | & & | \end{pmatrix} \tag{15}$$

which we can rewrite as, using (14),

$$\tilde{D} = \Phi[I \ \Lambda \ \Lambda^2 \ ... \ \Lambda^{m-1}](I \otimes \xi) = \Phi \text{diag}(\xi) C, \tag{16}$$

where I is the $m \times m$ identity matrix and

$$C = \begin{pmatrix} \lambda_1^0 & \lambda_1^1 & \lambda_1^2 & \cdots & \lambda_1^{m-1} \\ \lambda_2^0 & \lambda_2^1 & \lambda_2^2 & \cdots & \lambda_2^{m-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \lambda_m^0 & \lambda_m^1 & \lambda_m^2 & \cdots & \lambda_m^{m-1} \end{pmatrix} \tag{17}$$

is a Vandermonde matrix. By thinking of the eigenvalues λ_i as complex numbers of the form $\lambda_i = Re^{i\theta}$, we can see that λ_i^0, λ_i^1, λ_i^2,... all represent the same complex frequency, with the amplitude increasing (decreasing) along the row if $|\lambda_i| > 0$ ($|\lambda_i| < 0$).

To summarise the Koopman method so far, we have managed to decompose our $\tilde{n} \times m$ observables matrix $\tilde{D}$ into the product of the $\tilde{n} \times m$ eigenfunction matrix Φ, the $m \times m$ diagonal matrix of eigenvalues Λ and the Vandermonde matrix C. However, we still do not know the values of Φ, Λ or C - all we know is that $\tilde{D}$ can be written as a product of such matrices. There is not an obvious way to calculate this factorisation from $\tilde{D}$, but we can make progress by using the fact that Vandermonde matrices diagonalise companion matrices. Companion matrices are defined as

square $k \times k$ matrices with all entries zero apart from having ones on the subdiagonal and non-zero entries in the k^{th} column,

$$S = \begin{pmatrix} 0 & & & & a_1 \\ 1 & 0 & & & a_2 \\ & 1 & 0 & & a_3 \\ & & \ddots & \ddots & \vdots \\ & & & 1 & a_k \end{pmatrix}, \tag{18}$$

and they can be diagonalised by a Vandermonde matrix C as

$$S = C^{-1}BC, \tag{19}$$

where B is diagonal. Using this, and taking $B = \text{diag}(\xi)$ as our diagonal matrix, we can rewrite our decomposition of the observables matrix $\tilde{D}$ as

$$\tilde{D} = \Phi \text{diag}(\xi) C = \Phi C S. \tag{20}$$

Hence, an alternative way to factorise the observables matrix is as

$$\tilde{D} = \Phi' S, \tag{21}$$

where $\Phi' = \Phi\xi$. The factorisation is much more helpful because it is easy to understand the effect of the companion matrix S. The subdiagonal ones in S tell us that post-multiplying by S shifts the columns to the left by one. Therefore, we can infer what Φ' must be, by noting that

$$\begin{pmatrix} | & | & \dots & | \\ \varphi_2 & \varphi_3 & \dots & \varphi_m \\ | & | & \dots & | \end{pmatrix} \approx \begin{pmatrix} | & | & \dots & | \\ \varphi_1 & \varphi_2 & \dots & \varphi_{m-1} \\ | & | & \dots & | \end{pmatrix} \begin{pmatrix} 0 & & & & a_1 \\ 1 & 0 & & & a_2 \\ & \ddots & \ddots & & \vdots \\ & & & 1 & a_k \end{pmatrix}. \tag{22}$$

Introducing the notation $\tilde{D}_i$ to mean the matrix formed of columns i through to $i + m - 2$ of $\tilde{D}$, we can rewrite this as

$$\mathcal{K}\tilde{D}_1 = \tilde{D}_2 \approx \tilde{D}_1 S. \tag{23}$$

Note that the final equality here is not exact. This is because φ_m is not one of the columns of $\tilde{D}_1$, so we approximate $\varphi_m \approx a_1\varphi_1 + ... + a_{m-1}\varphi_{m-1}$ as a linear combination of the columns of $\tilde{D}_1$, where the coefficients a_i are determined using a least-squares approximation.

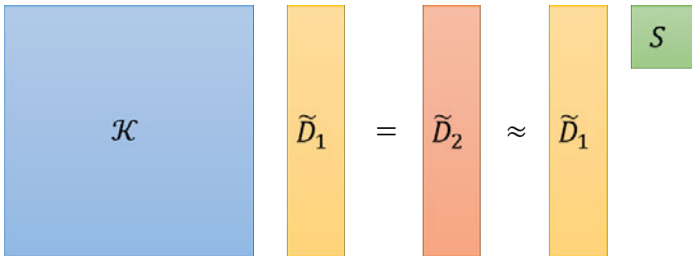


Figure 3: Graphical representation of (23).

This tells us that the smaller $(m-1)\times(m-1)$ companion matrix S contains the same information as the larger $n \times n$ Koopman operator $\mathcal{K}$. Therefore, we can calculate S and recover all the

information we need from S without ever having to calculate $\mathcal{K}$. We can use the the observables matrix $\tilde{D}$ to calculate S (in particular the unknown non-zero entries in the right-hand column) as

$$S = \text{pinv}(\tilde{D}_1)\tilde{D}_2, \tag{24}$$

where pinv means the pseudo-inverse. That is, we can reconstruct S from the data. Then the eigenvalues Λ of $\mathcal{K}$ are given by the eigenvalues of S, and the eigenfunction matrix Φ of $\mathcal{K}$ are related to the eigenfunction matrix X of S by $\Phi = \tilde{D}_1 X$.

4.1 Discussion

The Koopman method is the first technique we have looked at for using data to approximate the dynamics of the studied system. The Koopman operator is an approximate *linear* map from one snapshot in time to the next, which can give useful insight into the governing dynamics. However, the linear nature means that the Koopman method will struggle to capture complicated nonlinear dynamics.

The Koopman method is based around choosing observables such that the nonlinear dynamical system becomes linear when expressed in terms of the observables. With regards to this, we are yet to answer the following question:

How do we choose the observables? Note that all our analysis here has been done using the matrix $\tilde{D}$ of *observables* φ, rather than the matrix D of *data* $\mathbf{d}$. The transformation from data to observables is an important step of the Koopman process since this is what makes the nonlinear dynamical system describing the physical process linear. However, the question of how to choose the suitable observables that will produce linear system still remains.

In practice, the observables are often chosen simply to be the data, so $\varphi = \mathbf{d}$. If we have a large enough number of data points (large enough n), we can have enough degrees of freedom that using the data as the observables will produce a linear system by brute force. This is the case provided that the number of data points n is significantly larger than the number of equations in the dynamical system describing the physical process that our data has come from. (Note that we will likely not know what these equations are.) However, the resulting observables matrix (equal to the data matrix) will be large, making the computations costly. The cost is manageable if we are looking at relatively short timescales with only a few snapshots in time (small m), but the cost becomes too much for long timescales with lots of snapshots (large m). In this case, we need to think carefully about what observables to choose to make a smaller linear system.

References

[1] O. T. Schmidt and P. J. Schmid, *A conditional space-time pod formalism for intermittent and rare events: example of acoustic bursts in turbulent jets*, J. Fluid Mech., 867 (2019).

GFD 2022 Lecture 4: Transfer Operators for Data Analysis — Frobenius-Perron Operator (Part 2)

Peter Schmid; notes by Claire Valva and Rui Yang

1 Introduction

In this lecture, we continue the discussion on approximating of dynamical operators from data, specifically the transfer operator. The transfer operator encodes information about an iterated map and is frequently used to study the behavior of dynamical systems. In the previous lecture we discussed the Koopman operator. Here we will discuss the Frobenius–Perron Operator.

In the following sections, we will describe the Frobenius–Perron operator, including methods of approximation, information about the continuous ($\Delta t \to 0$) limit of transfer operators, as well as give examples of its use in fluid dynamics.

2 Frobenius–Perron Operator

The Frobenius-Perron operator $\mathcal{P}$ is a linear, infinite-dimensional representation of a finite-dimensional dynamical system that describes the propagation of probability densities over one time step. It is defined as

$$\int_A \mathcal{P} f d\mu = \int_{\Phi^{-1}(A)} f d\mu, \tag{1}$$

where μ denotes a probability measure, f a density of a random variables, and Φ the forward mapping over one time step. The above expression thus expresses the propogation of the probability density over one time step: the integrated density at the previous time step (right-hand term) is equivalent to the integrated density propagated by $\mathcal{P}$ (left-hand term).

It is the adjoint (dual) of the Koopman operator $\mathcal{K}$. The information contained in this transition operator encapsulates a statistical description of the turbulent fluid motion, expressed as transition probabilities from one state to another. This approach is also closely related to a probability density analysis within a Fokker-Planck description of a dynamical system.

3 Ulam's Method: Frobenius–Perron Operator Approximation

We now use this defnition to find an approximation of the operator $\mathcal{P}$, the most popular approximation method is Ulam's method, also called Ulam–Galerkin approximation.

Practically, when given a data matrix D (with size $n \times m$ where n is the spatial dimension and m is the time), we will try to reduce the dimensionality of the phase space. The flow can be embedded to a lower-dimensional phase space with basis mods, such as POD modes, onto which the dynamic system is projected. For example, we apply SVD to the original data

$$D = U\Sigma V^H \approx U[:, 1:r]\Sigma[1:r, 1:r]V^H[:, 1:r], \tag{2}$$

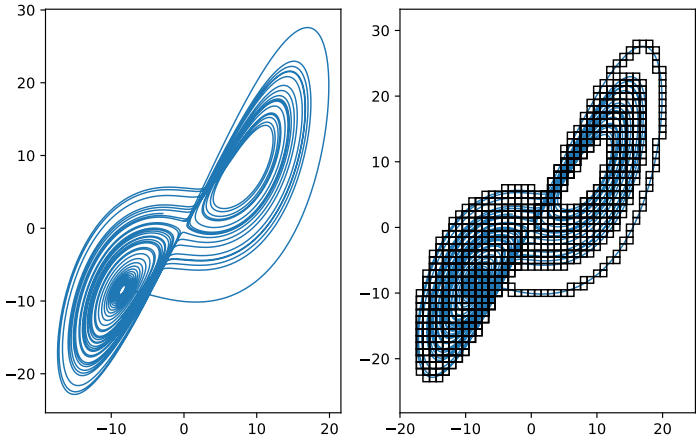


Figure 1: The left Figure shows the phase space of the embedded attractor. The right Figure shows a covering of boxes $\mathbb{B}_i$ which will be used to estimate the action of the dynamical system in those cells $\mathbb{B}_i$ in which the attractor is embedded.

where r is the truncation number of the reduced dimension.

Then, we need to tessellate this phase space to estimate transition probabilities. A very simple and efficient tiling of the region of the phase space is to use rectangular boxes and the counting measure. More formally, denote the set of the box elements from tesselation by $\{B_1, ..., B_k\}$ and write the indicator function for B_i as

$$\mathbb{I}_{B_i(x)} = \begin{cases} 1, & \text{if } x \in B_i, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

A Galerkin projection of $\mathcal{P}$ onto a subspace spanned by the indicator function is expressed by

$$\int \mathbb{I}_{B_j} \mathcal{P} \mathbb{I}_{B_i} d\mu = \mu(\Phi^{-1}(B_j) \cap B_i) \text{ where } \mu \text{ is the counting measure.} \tag{4}$$

The approximation P of the Frobenius–Perron operator $\mathcal{P}$ can be defined as

$$P_{ij} = \frac{\mu(\Phi^{-1}(B_j) \cap B_i)}{\mu(B_i)}, \ i, j = 1, ..., k \tag{5}$$

where k is the number of trajectory boxes. This expression gives P_{ij} as the ratio of the number of trajectory points previously (one time step earlier) in B_j, are now in B_i and the total number of trajectory points in B_i. An illustration is plotted in Figure 1. The matrix P is referred to as Ulam's method [4].

The approximation P denotes a Markov process and is row-stochastic, i.e. each row of P sums to one, and $\lambda_{max}(P) = 1$. The diagonal of P denotes the resting probability for each state. High resting probability will correspond to high persistence of a state in time. Correspondingly, a low resting probability corresponds to a fairly unstable state. This method is often found to be suitable to describe rare events and bimodal/multimodal states.

There are a few practical considerations including the box size: if the boxes are too big, detailed information inside boxes will be lost. If the boxes are too small, there are few trajectory points in the boxes, which makes the probabilities P_{ij} less reliable to compute.

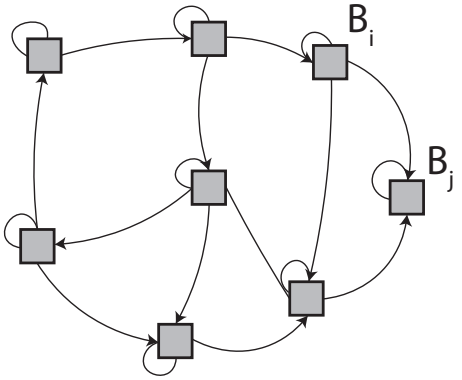


Figure 2: An example drawing of the graph, where nodes are represented by the boxes B_i, and edges are represented by the connections between two boxes B_i, B_j.

4 Graph Theoretic Algorithm: Community Clustering

The matrix P can also be interpreted as an adjacency matrix of a weighted directed graph, where nodes are represented by the boxes B_i, and edges are represented by the connections between two boxes B_i, B_j (see Figure 2). The transition probability P_{ij} represents the weight on the edges.

This graph can be consolidated into communities of similar clusters. A community is defined as a collection of nodes from the graph that shows strong intraconnectivity (inside the community) and weak interconnectivity (between communities). Several algorithms exist for detecting communities, one of which is referred to as modularity, defined as

$$Q = \frac{1}{k} \sum_{i,j} \left[P_{ij} - \frac{k_i^{in} k_j^{out}}{k} \right] \delta_{c_i, c_j}, \tag{6}$$

where k_i^{in}, k_j^{out} are the in and out-degrees of the nodes, c_i the community i, $\delta_{i,j}$ the Kronecker symbol. From this definition, we seek for a division of the graph into communities such that maximizes Q. This optimization problem can be realized by using methods such that include a greedy algorithm proposed by Leicht & Newman [1]. An example of the pattern of the transition probability matrix is shown in Figure 3 and more details can be found in [3].

5 Continuous Limits

The continuous limit of Koopman K operator is

$$\lim_{t \to 0} \frac{K_t \phi - \phi}{t} = \mathcal{L} \phi \tag{7}$$

$$\frac{d\phi(\mathbf{x})}{dt} = \nabla \phi \cdot \frac{d\mathbf{x}}{dt} = \nabla \phi \cdot \mathbf{f}, \tag{8}$$

where $\mathbf{f}(\mathbf{x}) = d\mathbf{x}/dt$ and ϕ is an observable. Thus, we can obtain

$$\mathcal{L} = \nabla(\cdot) \cdot \mathbf{f}, \tag{9}$$

which is called Lie-operator.

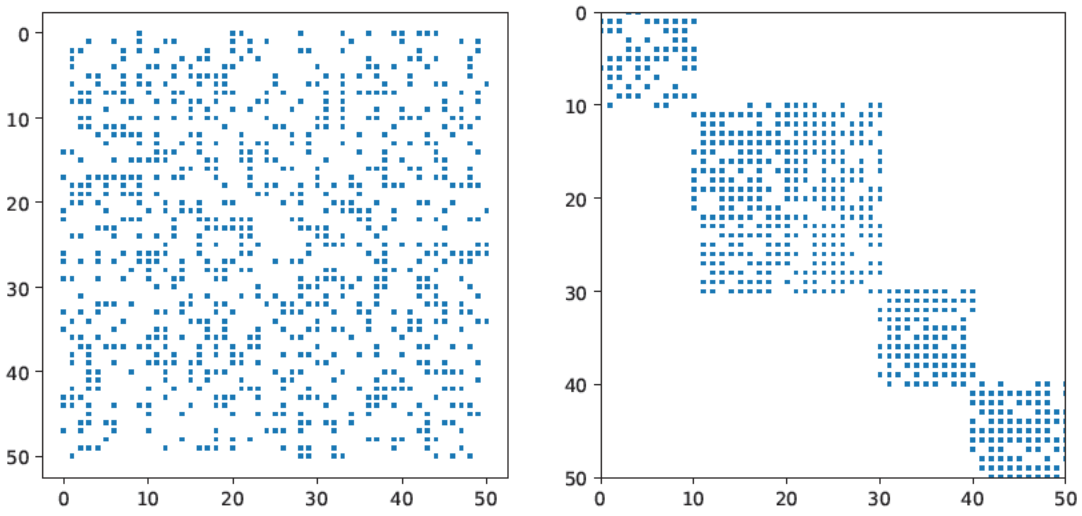


Figure 3: An illustration of the fill pattern of the transition probability matrix P. Left: The transition matrix before sorting, in arbitrary configuration; Right: The transition matrix after sorting to reveal a block-diagonal form.

Primal	Dual/adjoint	Type
Koopman operator K	Frobenius–Perron operator P	(discrete)
Lie operator $\nabla(\cdot)\cdot \mathrm{f}$	Liouville operator $\nabla\cdot((\cdot)\mathrm{f})$	(continuous)

6 Examples

One example is [3], where Frobenius–Perron Operator was used to analyze slow-fast dynamics of turbulent shear flows. It has been applied to a simple nine-dimensional model of the self-sustaining process in wall-bounded flows and has shown its capability in isolating a hierarchy of dynamic communities. Figure 4 shows the model system.

Another example is from [2], where the authors found that the polar vortex can be identified as a quasi-invariant set of the Frobenius–Perron operator. It shows that Frobenius–Perron operator has wide potential application in detecting and analyzing mixing structures in a variety of atmospheric, oceanographic, and general fluid dynamical settings.

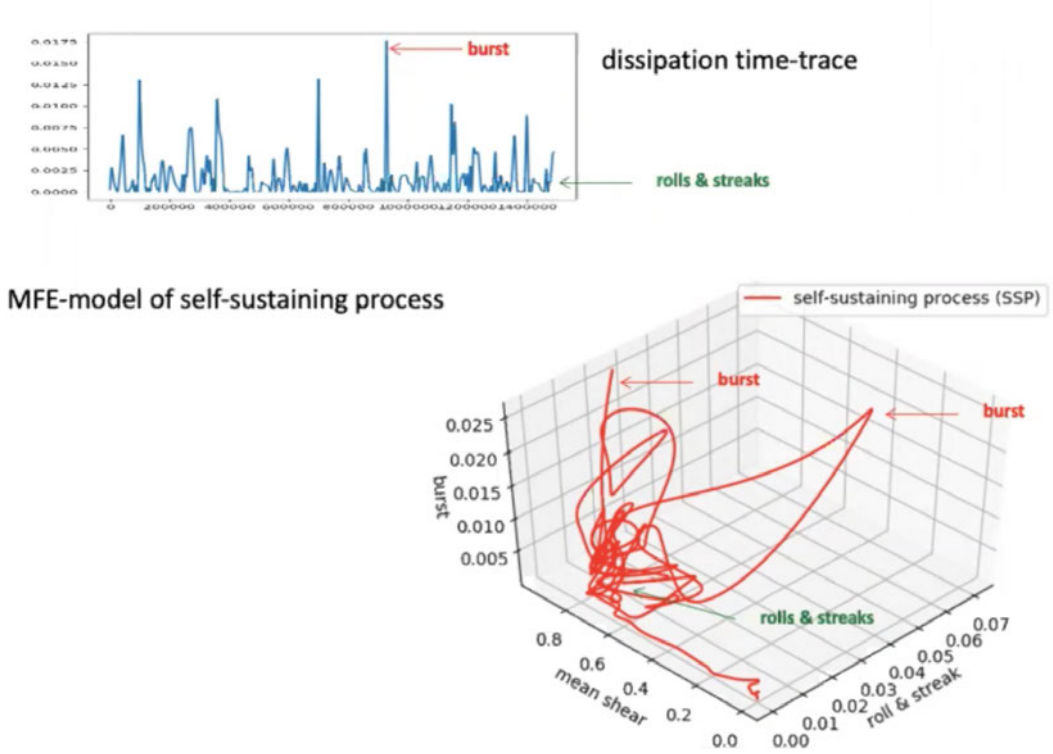


Figure 4: Dissipation as a function of time and the energy-dissipation phase-space trajectory

References

[1] E. A. Leicht and M. E. Newman, *Community structure in directed networks*, Physical review letters, 100 (2008), p. 118703.

[2] N. Santitissadeekorn, G. Froyland, and A. Monahan, *Optimally coherent sets in geophysical flows: A transfer-operator approach to delimiting the stratospheric polar vortex*, Physical Review E, 82 (2010), p. 056311.

[3] P. Schmid, O. Schmidt, A. Towne, and M. Hack, *Analysis and prediction of rare events in turbulent flows*, Proceedings of the Summer Program 2018, (2018), pp. 139–48.

[4] S. M. Ulam, *A collection of mathematical problems*, no. 8, Interscience Publishers, 1960.

GFD 2022 Lecture 5: Linear Inverse Modeling and Linear Response Theory

Laure Zanna; notes by Ludovico Giorgini and Ruth Moorman

1 Introduction

In this lecture we continue to discuss methods of extracting dynamical operators from data. As outlined in Lecture 3, we are seeking linear operators, called transfer operators, that map one system snapshot to another, either forwards (Koopman operator) or backwards (Frobenius-Perron operator) in time. Here we return our focus to forward propagating transfer operators or Koopman operators.

The utility of identifying or approximating Koopman operators in GFD problems is significant; they can be used pragmatically for forecasting, but they also have the potential of uncovering linear dynamical modes imbedded in the system, a more rewarding but elusive target than the modes of variability (essentially a time series of a pattern, sometimes without obvious physical meaning) produced by PCA. However, the formalism underlying Koopman analysis requires us to significantly increase the dimensionality of our system through the inclusion of a large number of observables φ. In fact, we technically require infinite φ to approach the linear limit of a nonlinear system, though very large finite φ suffice. In the methodologies introduced in Lecture 3, we generally take our data as observables such that $\varphi = \mathbf{d}$. As such, **Koopman analysis requires very large amounts of data**! For some questions and some systems these data requirements are simply impractical and other, less rigorous, methods of approximating transfer operators are called for. In the first part of this lecture (§2) we discuss a popular alternative, a subset of Koopman analysis called Linear Inverse Modeling (LIM). In the second part of the lecture (§3) we broaden some of the ideas introduced with LIM to estimate the response of nonlinear systems to forcing.

2 Linear Inverse Modeling (LIM)

We consider a multiscale autonomous dynamical system whose state x, is described by

$$\frac{dx(t)}{dt} = \underbrace{\mathcal{F}\big(x(t)\big)}_{\text{nonlinear dynamics}} + \underbrace{\sigma}_{\text{small noise}} \overbrace{\eta(t)}^{\text{Gaussian}}, \tag{1}$$

where F imparts the full nonlinear dynamics on state vectors $x(t)$, while $\sigma\eta(t)$, is zero mean Gaussian noise η with amplitude σ, describing small perturbations imparted to the system. Usually, $x(t)$ is taken as a **reduced** form the state of a high-dimensional system whose k first principal components are retained (PCA, see Lecture 1), neglecting redundant higher order modes.

We now separate timescales by studying the fluctuations of x around its time average $\bar{x}$, such that $x' = x - \bar{x}$ (1) will then approximately satisfy

$$\frac{dx'(t)}{dt} = \underbrace{Bx'(t)}_{\text{linearized dynamics}} + \underbrace{\sigma}_{\text{small noise}} \overbrace{\eta(t)}^{\text{Gaussian}}, \tag{2}$$

with $B = \frac{\partial \mathcal{F}}{\partial x}\big|_{x=\bar{x}}$ is a linear operator. **It is critical to note that in LIM we explicitly linearize our system dynamics at this early stage and allow any nonlinearity to be subsumed into the noise term.** This is where we diverge from the more data-intensive Koopman analysis of previous lectures, which attempts to approximate the nonlinear dynamics of the system via the retention of a large number of observables, whereas in LIM we generally reduce our data with PCA at the outset but still assume a linear operator B. The methodology is not completely removed from Koopman analysis however, and may be interpreted as the limit of the Koopman operator to purely linear dynamics.

Say that we can observe $x'(t)$ but want to forecast $x'(t+\tau)$, i.e., propagate our system forward in time. Integrating (2) from t to $t+\tau$ gives

$$x'(t+\tau) = e^{B\tau}x'(t) + \int_t^{t+\tau} e^{B(\tau-s)}dW(s), \tag{3}$$

with $\eta(t) = \frac{dW}{dt}$. The predicted state of the system at time $t+\tau$ will simply be

$$x'(t+\tau) \approx E[x'(t+\tau)|x'(t)] = e^{B\tau}x'(t). \tag{4}$$

To predict the future state of the system we need to construct B, the linear dynamics, from data. To do this we note that the lagged covariance matrix of the system $C(\tau)$, can be estimated as

$$C(\tau) = \big(x'(t+\tau)\big)\big(x'(t)\big)^T = e^{B\tau}C(0), \tag{5}$$

by right multiplying (4) by $x'(t)^T$. Then we can simply recover B for a given τ as

$$B = \frac{1}{\tau}\log\left[\frac{C(\tau)}{C(0)}\right]. \tag{6}$$

In practice, we would want to investigate many choices of τ to improve our approximation of B, e.g.,

$$B^{-1} = -\int_0^\infty C(\tau)C(0)^{-1}\,d\tau \tag{7}$$

but if our assumptions in (2) are good B should be relatively insensitive to τ. Thus once we have approximated B our assumption of the system linearity should permit forecasting at a range of small propagation timescales. Note that subsuming all neglected nonlinear terms into the noise term in (2) introduces an error that grows with the forecasting timescale such that the predictions becomes less accurate for longer range forecasts. For this reason LIM is only appropriate for short time forecasts. As noted previously, B can also be interrogated for physical insights as it may contain information about the linear dynamical modes of the system operating on short timescales.

We can also back out σ in order to assess the degree to which our forecasts may diverge from reality at long time by considering the second moment equation of our process

$$\frac{dC(0)}{dt} = BC(0) + C(0)B^T + \sigma\sigma^T, \tag{8}$$

obtained by multiplying $\eta = dx/dt - Bx(t)$ by its transpose. Since both B and σ are constant, we obtain

$$\sigma\sigma^T = -BC(0) - C(0)B^T. \tag{9}$$

If our studied system is well described by (2), then the outcome of LIM should (roughly) the following characteristics:

- B should be independent of the chosen τ in (6),
- $\sigma\sigma^T$ (9) should be positive definite,
- system statistics are Gaussian, and
- our forecasts are generally "good" (my some fitness measure) but errors grow with increased forecast time.

These criterion serve as good sanity checks for the applicability of LIM to a given problem.

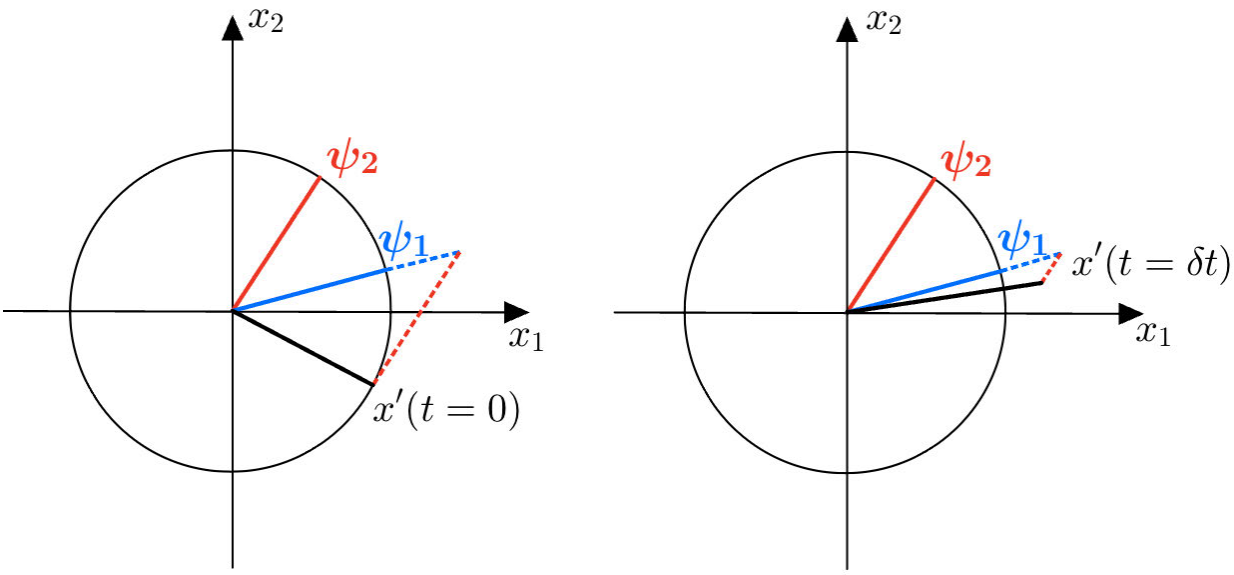


Figure 1: Transient growth of a state vector comprised of two decaying eigenmodes.

Finally, we note that for the approximation in (2) to be valid, all the eigenvalues of B must be negative. Thus, by construction, (2) forces any initial perturbation to relax exponentially toward zero, i.e., a LIM-based propagator should never predict perturbation growth. However, **transient** perturbation growth is possible even when all eigenmodes are decaying if such eigenmodes decay at unequal rates. An example of this process in 2D is depicted in Figure 1 for a system with decaying eigenvectors and eigenvalues $\psi_{1,2}$ and $\lambda_{1,2}$ where the second eigenmode decays at a greater rate than the first, leading to transient growth of the state vector x'. We can extend this logic to determine which initial conditions are precursors to large transient growth (i.e., most growing modes), however, the procedure for doing this is not discussed in the lectures and readers are referred to the papers discussed in (§2.1).

2.1 Example: forecasting sea-surface temperature (SST) anomalies with LIM

There have been numerous LIM studies where PCs of SST anomalies have served at the state vectors $x'(t)$ in (2). An early example, from which much of the explanation above has been adapted, is [1] wherein the time dependence of the first 15 PCs of Tropical Pacific SST was modeled by (2) and the resulting B matrix was used to identify precursors to large ENSO events (Figure 2). A more recent example also interrogating ENSO using LIM of Tropical SST anomalies is [2] which extends this analysis to identify precursors to different flavors of ENSO (Figure 3). Whilst these plots highlight results relating to maximal growth modes (not discussed explicitly in the lecture) both studies present dynamical interpretations of the linear modes and assessments of the predictive timescales associated with El Nino events.

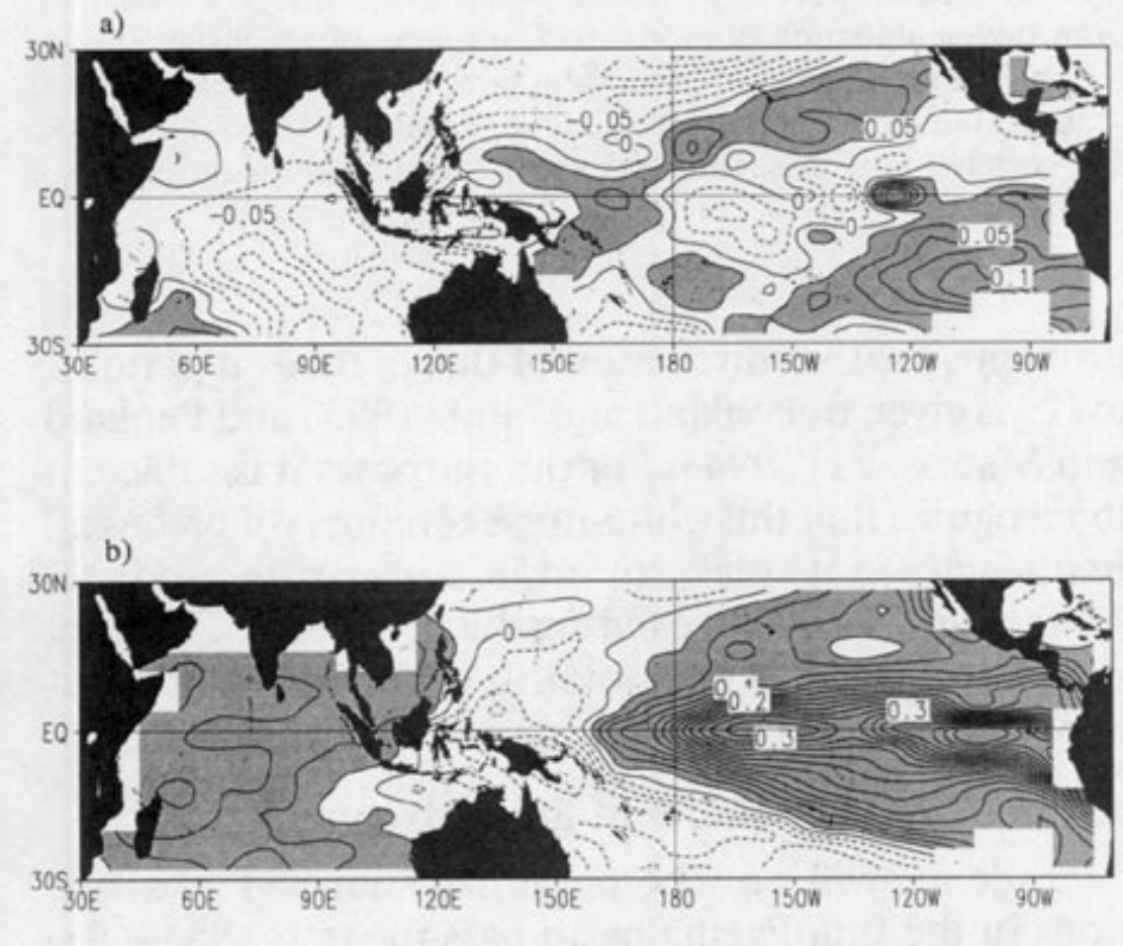


Figure 2: (top) Initial SST perturbation pattern optimized to give maximum amplification of SST anomalies at 7 months. (bottom) SST field resulting from the forward propagation (using LIM) of this initial condition for 7 months. From [1].

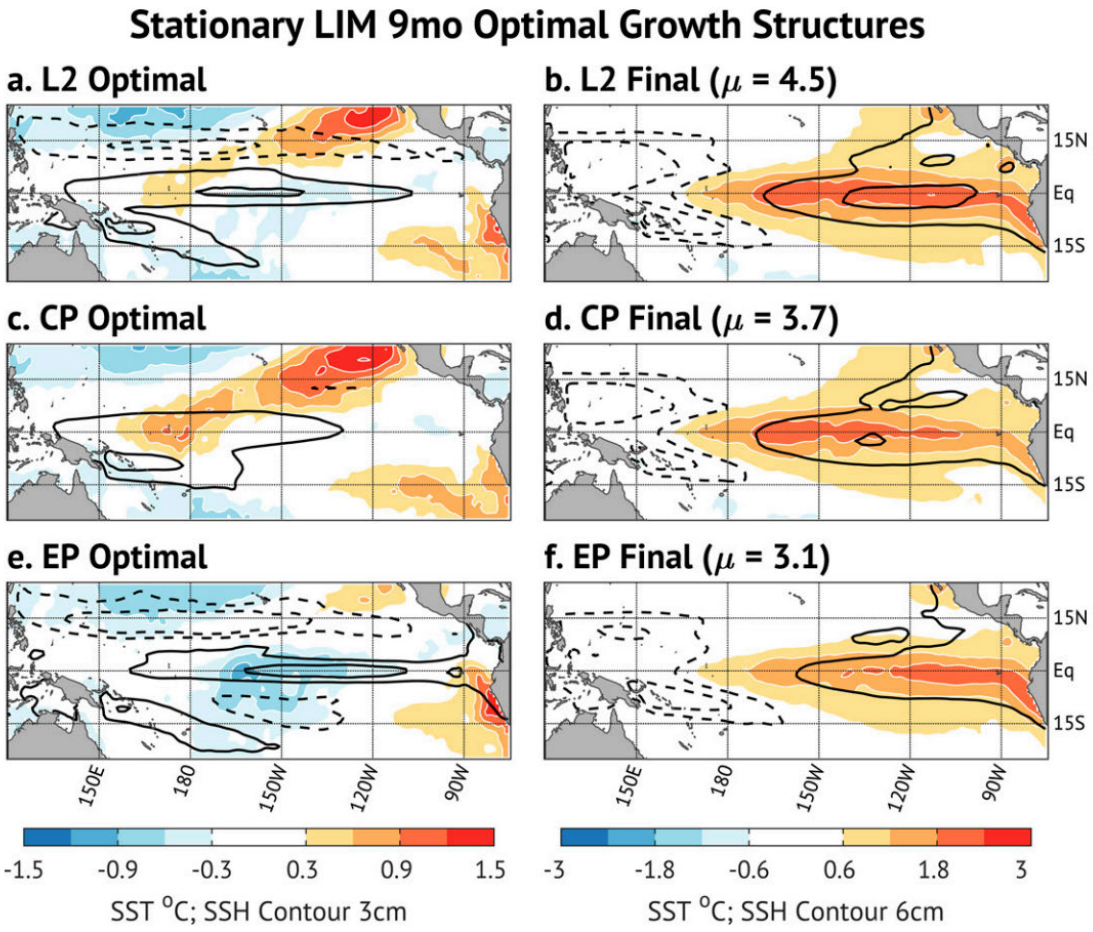


Figure 3: (left) Initial SST perturbations patterns found to give maximum amplification of SST anomalies at 9 months to produce various ENSO patterns (right). L2, CP, and EP refer to norms targeting canonical, central Pacific, and eastern Pacific flavors of ENSO. From [2].

3 Linear Response Theory

Let's return to the dynamical system described in (1) with the addition of a small external forcing perturbation δf

$$\frac{dx(t)}{dt} = \underbrace{\mathcal{F}(x(t))}_{\text{nonlinear dynamics}} + \underbrace{\sigma}_{\text{small noise}} \overbrace{\eta(t)}^{\text{Gaussian}} + \underbrace{\delta f}_{\text{small forcing}} . \tag{10}$$

Here we will not assume the system is linear but will instead consider how the bulk statistics of the full nonlinear system respond to a small force. Within this framing, we say that the original unforced nonlinear dynamical system (1) is associated with some probability distribution function (PDF) ρ, whilst the forced system (10) is associated with the PDF ρ_f.

The time evolution of the forced system PDF is described by the Fokker-Plank equation,

$$\frac{\partial \rho_f}{\partial t} + \nabla \cdot \big((\mathcal{F} + \delta f)\rho_f\big) = \sigma \nabla^2 \rho_f, \tag{11}$$

and similarly for ρ. We then define the mean state of the forced system as,

$$\bar{x}_f = \lim_{T\to\infty} \int_0^T x_f(t)\, dt = \int_{\substack{\text{all}\\ \text{states}}} x_f\, \rho_f(x_f)\, dx_f \tag{12}$$

and similarly for $\bar{x}$.

Now the effect of the perturbation to the mean is expressed as

$$\delta\bar{x} = \bar{x}_f - \bar{x} \approx \mathcal{L}\delta f \tag{13}$$

where $\mathcal{L}$ is the linear response operator. Here we have not linearized the full system, but based on the assumption of a small forcing we are linearizing the system response to δf.

Using a derivation based on the Fokker-Plank theory we can express $\mathcal{L}$ as,

$$\int_0^\tau \int_{\substack{\text{all}\\ \text{states}}} x(t+\tau)\{A[x(t)]\}^H \rho\, dx\, d\tau \tag{14}$$

which is purely a function of the unforced system. We assume ρ is Gaussian and $A = -\frac{1}{\rho}\nabla\rho$, which leads to

$$\mathcal{L} = \int_0^\infty C(\tau)C(0)^{-1}\, d\tau, \tag{15}$$

with $C(\tau)$ as the lagged covariance matrix of the unforced data. Despite the similarities with linear inverse modelling, here we are exploring at the entire phase space (integrating over the whole time for the entire state of the system) instead of single trajectories. In summary, if we know the statistics of the unperturbed system, we can construct the linear-response operator that can be used to estimate the time response of the system to a small perturbation.

Above, we consider the effect of a small perturbation on the mean of the system, but the same can be done to other moments of the data, such as the standard deviation. To give some context, let's assume we have a forced Lorenz system given by

$$\frac{\partial x}{\partial t} = -\sigma x + \sigma y + f\cos(\theta), \tag{16a}$$

$$\frac{\partial y}{\partial t} = -x\, z + r\, x - y + f\sin(\theta), \tag{16b}$$

$$\frac{\partial z}{\partial t} = x\, y - b\, z, \tag{16c}$$

if f is small enough, the variability of the system continues to project into the dominant modes and the effect of the forcing on the statistics is predictable (Figure 4).

In Figure 4 we notice that the time spent by particles in each lobe changes with θ, but the variability is along the same paths as the unperturbed solution. In other words, the addition of forcing does not change the nature of permitted system solutions but does change the frequency with which different solutions are occupied. While this method can be applied to large datasets to investigate the effect of a small forcing into a climate dynamics, it is sensitive to the use of SVD for dimension reduction (note how EOFs would be inappropriately applied to the bimodal distribution in Figure 4).

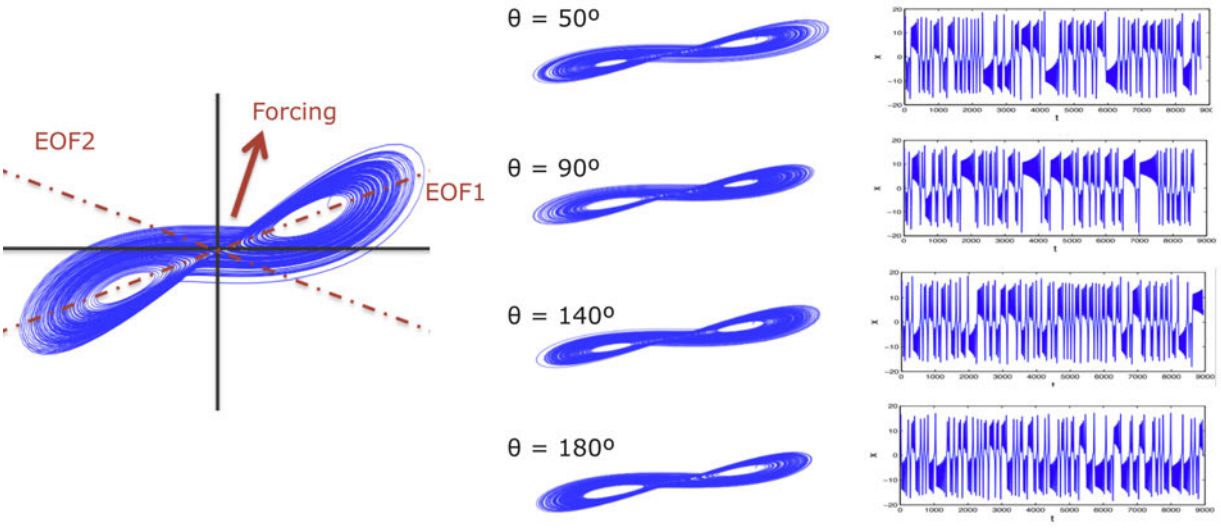


Figure 4: (left) Solution of unforced Lorenz system, the two main orthogonal modes of variability and a schematic representation of a a forcing. (center) Solution of the forced Lorenz system for a small f and different angles θ. (right) Time series of the forced Lorenz system for a small f and different angles θ.

References

[1] C. Penland and P. D. Sardeshmukh, *The optimal growth of tropical sea surface temperature anomalies*, Journal of climate, 8 (1995), pp. 1999–2024.

[2] D. J. Vimont, M. Newman, D. S. Battisti, and S.-I. Shin, *The role of seasonality and the enso mode in central and east pacific enso growth and evolution*, Journal of Climate, 35 (2022), pp. 3195–3209.

GFD 2022 Lecture 6: Real Data and Optimisation

Peter Schmid; notes by Sam Lewin and Kasturi Shah

1 Formulating an Optimisation Problem

1.1 Introduction

Many decompositions of data-streams we have seen are solutions to an optimization problem, subject to some constraints that enforce desired solution characteristics. For example, we can frame SVD as the solution to the following optimisation task:

$$\min_{\hat{D}} ||D - \hat{D}|| \quad \text{such that } \operatorname{rank}(D) \leq r, \tag{1}$$

for some user-specified rank r. In words, we are seeking a low-rank matrix $\hat{D}$ that most closely approximates our starting (potentially very high-dimensional) matrix D. We will get on to exactly what we mean by 'closely approximates' (i.e., how we are defining the norm $||\cdot||$ for matrices) in section §2.1. For interest, we note here the expected (but not necessarily obvious) result that the 'best' low-rank approximation for D is guaranteed to be obtained by SVD by the **Eckard-Young theorem**.

Optimisation algorithms are an extremely flexible framework within which to work because they can be easily modified when we want to impose **constraints**. In this lecture, we will introduce how to formulate an optimisation problem for data analysis, discuss solution characteristics that might be useful to have from a practical, physical or computational perspective, and explain how to modify the optimisation algorithm to impose these constraints.

1.2 Cost functions

In general, the **cost function** (sometimes also called a **loss function)** refers to the object we are trying to minimise, i.e., $||D - \hat{D}||$ above. Optimization algorithms give us flexibility in forcing solution characteristics at the expense of computational convenience: we can impose additional constraints by augmenting the cost function.

A cost function $\mathcal{L}$ can be thought of as being made up of three components:

$$\mathcal{L} = \underbrace{\boxed{\text{data fidelity}}}_{\text{matching term}} + \lambda_1 \underbrace{\boxed{\text{solution characteristics}}}_{\text{e.g. sparse/compact support/low rank}} + \lambda_2 \underbrace{\boxed{\text{algorithmic convergence}}}_{\text{convergence to solution, 'convexification'}} \tag{2}$$

Let us give a qualitative overview of each term in the above equation. The first term ensures the solution stays 'close' to the original data (in terms of some appropriate norm). As we will explore below, the second term can be used to shape the solution to fit desired characeteristics. However, this can introduce problems in that it can give rise to a highly non-convex loss function, i.e., an $\mathcal{L}$ that has many local minima and maxima that are not optimal (and may, in many cases, be very

sub-optimal). This introduces the need for *regularization*, represented by the third term in the equation, which essentially smooths things out to improve convergence. Note that by tuning the free parameters λ_1 and λ_2 (which are often referred to as **hyperparameters** in the machine learning community), we can adjust how strongly we want to enforce the constraints and regularization.

2 Constraints

Solution constraints are closely linked to the choice of **norm**: i.e., how we choose to define the 'size' of the objects we are considering and the 'distance' between them. It is worth pointing out that many of the standard norms for vectors do not extend to matrices in an obvious fashion. Some common choices of norm and their definition applied to matrices are outlined below.

2.1 Choice of norm

Name	Notation	Formula	Notes
2-norm	$\|\|\ \|\|_2$	$= \sigma_{\max}$	The first singular value. It is energy-based, least squares, defined for matrix and vectors.
Frobenius norm	$\|\|\ \|\|_F$	$\text{rms}(\sigma) = \sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_n^2}$	For matching arrays or matrices. Root-mean squared, defined for matrices and vectors.
1-norm*	$\|\|\ \|\|_1$	$\|\|x\|\|_1 = \|x_1\| + \|x_2\| + \cdots + \|x_m\|$	Enforces sparity. Defined for vectors. Can be defined for $\max\left(\frac{\|Dx\|_n}{\|x\|_n}\right)$.
0-norm*	$\|\|\ \|\|_0$	Counts non-zero elements of vector.	Measures cardinality.
Nuclear norm	$\|\|\ \|\|_*$	$= \sigma_1 + \sigma_2 + \cdots + \sigma_m$	Computes the rank. Sum of the singular values of the matrix.
TV norm	$\|\|\ \|\|_{TV}$	$= \|\|\nabla \cdot\|\|_1$	Measures the smoothness of the solution by calculating the gradient of the 1-norm. When the total variation (TV) is low, the solution is smooth.
Huber loss function	$\|\|\ \|\|_H$		Blend of 1-norm and 2-norm (see §2.4)
Hybrid loss function	$\|\|\ \|\|_h$		Blend of 1-norm and 2-norm

* The 1-norm and 0-norm are considered "buddies." The 1-norm is a proxy for sparsity. The 0-norm is the true measure of sparsity, however, it is considered a little exotic and less frequently used.

Before proceeding with the discussion, there are two important messages to convey. First, a judicious choice of norms is always wise. For instance, the L2 norm may not necessarily be the best

choice. Second, adopting an optimization approach often yields the most insight.

2.2 Norms that promote sparsity

Sparsity generally assumes a parsimonious solution. It seeks to represent the solution in terms of the minimal number of modes needed. For example, to find a dictionary of functions for the solution, we want a minimal set to avoid overfitting the solution.

How then, does the L1 norm promote sparsity? The schematic in Figure 1 illustrates this.

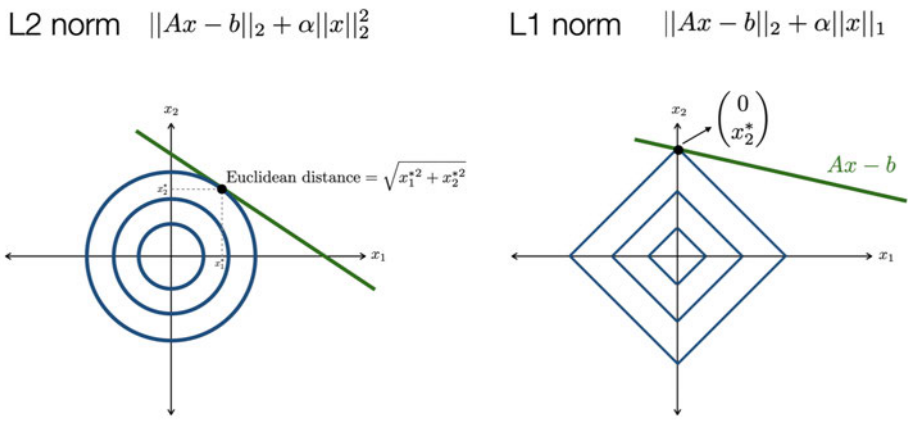


Figure 1: A schematic illustration of how the L1 norm promotes sparsity, compared to the L2 norm.

For general p, Figure 2 indicates how the behaviour of $||\quad\cdot\quad||_p$ changes in limits of p.

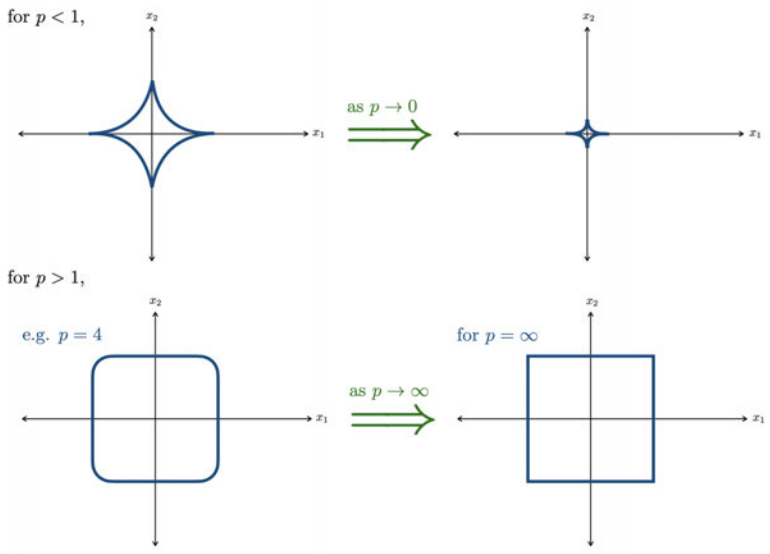


Figure 2: Behaviour of the norms as $p\to 0$ and $p\to\infty$.

2.3 The L_1 optimisation problem

Consider a toy one-dimensional version,

$$ax - b + c\ \text{sign}(x) = 0 \tag{3}$$

where $a \geq 0$ and $c \geq 0$.

Case 1: $x > 0$

$$x = \frac{b-c}{a}, \qquad \text{for } b > c \tag{4}$$

Case 2: $x < 0$

$$x = \frac{b+c}{a} \qquad \text{for } b > -c \tag{5}$$

Case 3: $x = 0$

$$-b - c \geq 0 \geq -b + c \qquad \text{for } -c \geq b \geq c \tag{6}$$

Constructing the full solution

$$x = \frac{S(b,c)}{a} \tag{7}$$

where

$$S(x,\lambda) = \begin{cases} x - \lambda & \text{for } x > \lambda \\ 0 & \text{for } -\lambda < x < \lambda \\ x + \lambda & \text{for } x < -\lambda. \end{cases} \tag{8}$$

$S(x,\lambda)$ is known as the shrink function. It describes "soft thresholding" as it is set to zero between $-\lambda$ and $+\lambda$ (Figure 3).

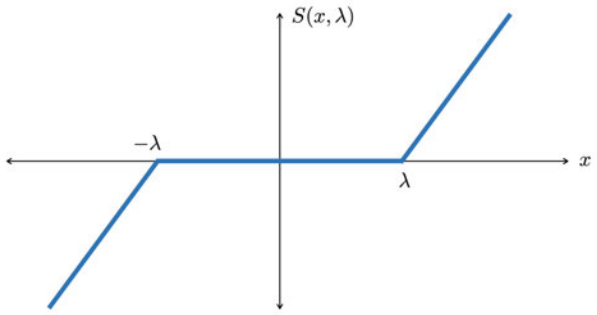


Figure 3: Sketch of shrink function $S(x,\lambda)$ in (8).

2.4 The Huber norm

The Huber loss function gives robust statistics and is insensitive to outliers. It is defined as

$$||x||_{H,\epsilon} = \begin{cases} \frac{1}{2}x^2 & \text{if } |x| \leq \epsilon, \text{ i.e., L2 norm} \\ \epsilon|x| - \frac{\epsilon^2}{2} & \text{if } |x| > \epsilon, \text{ i.e., L1 norm.} \end{cases} \tag{9}$$

The hybrid loss function is also a blend of the L1 and L2 norms,

$$||x||_{h,\epsilon} = \sqrt{1 + \frac{|x|^2}{\epsilon^2}} - 1. \tag{10}$$

3 Examples

Two examples are presented that prudently choose norms to deal with two common issues encountered in real data: outlier removal and filling in missing data.

3.1 Identifying outliers

A simple example of where optimisation with a 2-norm can go wrong is linear regression in the presence of outliers: since the error scales as a square, any outliers will have a strong influence on the solution.

The best method for outlier removal is LS decomposition, where the low-rank component searches for coherent patterns and the sparse component searches for outliers. Consider a matrix $\mathbf{D}$ that we wish to decompose into two components, a low-rank component $\mathbf{L}$ and a component containing sparse events, $\mathbf{S}$, such that $\mathbf{D} = \mathbf{L} + \mathbf{S}$. To do so, we can write down the LS decomposition,

$$\min_{L,S} ||L||_* + \lambda ||S||_1 \tag{11}$$

and reformulate it as a constrained optimisation problem by applying the augmented Lagrange multiplier method and formulating the Lagrangian functional,

$$\mathcal{L}(L,S,Y,\mu) = ||L||_* + \underbrace{\lambda}_{\text{user defined}} ||S||_1 + \frac{\mu}{2} \underbrace{||D-L-S||_F^2}_{\text{convexity improves convergence}} + \underbrace{< Y, D-L-S >}_{\text{inner matrix product improves behaviour}} \tag{12}$$

where Y is an adjoint matrix that enforces the minimization of $D - L - S$ and the second term is a penalty term controlled by the user-defined parameter μ. On doing the first variation of $\mathcal{L}$ with respect to L, S, Y, μ, we obtain four equations for L, S, Y, μ, which we can solve sequentially. Said differently, we do a component-by-component optimization of the Lagrangian functional, while keeping the other components fixed. We eventually obtain the desired spliting $\mathbf{D} = \mathbf{L} + \mathbf{S}$. The component $\mathbf{S}$ can be safely discarded as it contains the outliers and localised disturbances. The remaining data matrix $\mathbf{L}$ can be processed using your decomposition of choice, DMD, PCA, POD etcetera.

3.2 The matrix completion problem

The matrix completion problem is a possible way to fill gaps in data. Consider, for example, a satellite going off-line, the Halley Research Station on Brunt Iceshelf being moved due to the cracks and chasms appearing on the iceshelf itself leading to gaps in the Halley ozone record, clouds blocking features in observed data, etc. The question before us is: how do we fill in this data?

For instance, consider the matrix $\mathbf{D}$ with two missing data points, marked as x's,

$$D = \begin{pmatrix} 1 & 2 \\ x & 6 \\ 2 & x \end{pmatrix} \tag{13}$$

Without more information, it is impossible to fill in the missing data with guaranteed accurate results. However, if we know that $\text{rank}(\mathbf{D}) = 1$, then the completed matrix is clearly

$$D = \begin{pmatrix} 1 & 2 \\ 3 & 6 \\ 2 & 4 \end{pmatrix} \tag{14}$$

In general, we can fill the missing data if the rank of $\mathbf{D}$ is low, indeed, many data filling algorithms fill missing data by minimizing the rank of the matrix.

The missing data problem is also known as the 'Netflix problem,' Netflix posed an open question to the computer science community: given a sparsely filled matrix where the rows are users and the columns movies they have liked/disliked on Netflix, can we fill in the missing data to evaluate what each user's ratings for each movie would be? In other words, how can we fill data in a matrix under the premise that the matrix has low rank? This requires a projection onto the missing points, to avoid overwriting the existing datapoints. Additionally, an algorithm can remove redundancies by considering bulk features to improve predictability. For instance, aggregating data from users displaying preferences to predict the preferences of the group, or similar types of movies.

Finally, we turn our attention to a more complex problem: filling in missing information from a photograph. Here, the matrix entries represent the monochrome pixels of the photograph. The algorithm is implemented in Python (code available in Appendix 3.2). As payback for a cheeky remark earlier, Peter decided to remove pixels from a photograph of Colm and fill them in algorithmically. Much hilarity ensued. The results speak for themselves in Figure 4.

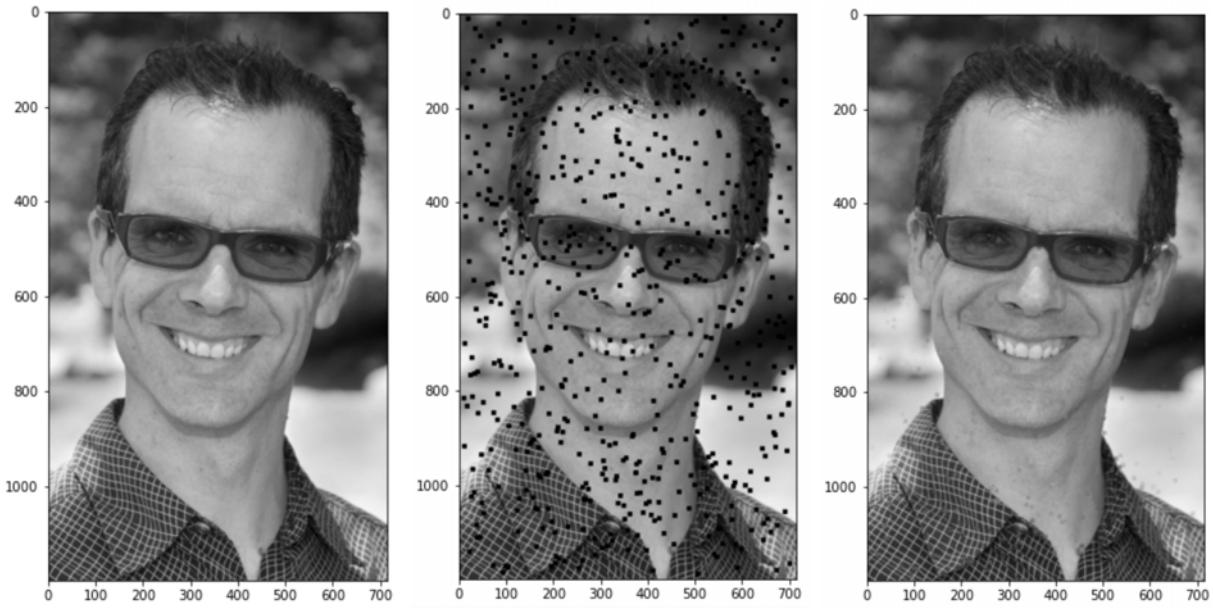


Figure 4: [left] Colm, the original. [center] Colm gone missing. [right] Colm complete.

Appendix A: Jupyter notebook for the matrix completion problem

This appendix provides code snippets for the matrix completion problem.

```
from math import sqrt
import numpy as np
from PIL import Image
import random
from matplotlib.pyplot import imshow,figure
```

Listing 1: Importing python packages

```
def convert2BW(img_in):
    im_file  = Image.open(img_in)                # open the image file
    im_array = np.array(im_file.convert('L'))    # convert to monochrome array
    return im_array

def shrink(A,tau):   # shrinkage operator
    return np.sign(A)*np.maximum(abs(A)-tau,0)

def softT0(A,tau):    # soft thresholding (shrinkage on SVD)
    U,S,Vh = np.linalg.svd(A,full_matrices=False)
    return U@np.diag(shrink(S,tau))@Vh
```

Listing 2: Defining functions

```
def DataRecovery(D,mu,rho): # incomplete alternating Lagrangian method (IALM),
 matrix completion
ep1,ep2 = 1e-8,1e-7                  # thresholds
Dn      = np.linalg.norm(D,'fro')
P       = (D==0).astype(float)       # projector matrix
m,n     = np.shape(D)                # initialization
Y,Eold  = np.zeros((m,n)),np.zeros((m,n))

for i in range(1,1000):                # iteration
    A    = softT0(D-Eold+Y/mu,1/mu)   # soft-threshold
    Enew = P*(D-A+Y/mu)               # project
    Y   += mu*(D-A-Enew)
    resi = np.linalg.norm(D-A-Enew,'fro')/Dn # check residual and (maybe) exit
    if (i%10==0): print(i,' residual ',resi)
    if (resi < ep1): break
    muf  = np.linalg.norm((Enew-Eold),'fro')  # adjust mu-factor
    if (min(mu,sqrt(mu))*(muf/Dn) < ep2): mu *= rho
    Eold = np.copy(Enew) # update E and go back
return A,Enew
```

Listing 3: Incomplete alternating Lagrangian method (IALM) for matrix completion

```
C = convert2BW('Colm.jpeg')    # read in data field
im_file1 = Image.fromarray(C)  # convert back to image
figure(figsize = (80,8))
imshow(im_file1,cmap="gray")
print(np.linalg.matrix_rank(C))
```

Listing 4: Reading in the original image

```
m,n = np.shape(C)                       # rows,columns
PP  = np.ones_like(C)
k   = 600
mm  = random.sample(range(10,m-10),k)
```

```
    nn   = random.sample(range(10,n-10),k)
    for i,j in zip(mm,nn): PP[i-5:i+5,j-5:j+5] = 0
        P       = PP.astype(float)
        Omega   = np.count_nonzero(P)            # number of non-zero elements
        D       = P*C                            # corrupted data matrix
        fratio = float(Omega)/(m*n)
    print('fill ratio ', fratio)

    im_file2 = Image.fromarray(D)
    figure(figsize = (80,8))
    imshow(im_file2)
```

Listing 5: Artificially corrupting the original matrix

```
    mu,rho   = 1./np.linalg.norm(D,2),1.2172 + 1.8588*fratio  # parameters
    AA,EE = DataRecovery(D,mu,rho)                                  # call IALM-
    algorithm
    print('converged')
```

Listing 6: Run the matrix completion problem

```
10  residual  0.14249288024027906
20  residual  0.09429641974212426
30  residual  0.05654256794508527
40  residual  0.033931477320171856
50  residual  0.024496219227798283
60  residual  0.015474261904865864
70  residual  0.010064868362384218
80  residual  0.0072883673915539035
90  residual  0.003892536054872016
100  residual  0.0025857124944492133
110  residual  0.00157624036549276 9
120  residual  0.0010695809976865203
130  residual  0.0008395286481844906
140  residual  0.0005193533407373907
150  residual  0.0002538931901600006
160  residual  0.00012428319688924313
170  residual  7.035634842273233e-06
180  residual  1.2163186400874667e-07
converged
```

Figure 5: Example output of code as it converges.

```
    im_file3 = Image.fromarray(AA)
    figure(figsize = (80,8))
    imshow(im_file3)
```

Listing 7: Displaying the image with data filled in

GFD 2022 Lecture 7: Bayesian and Markovian Approaches to Data Analysis

Laure Zanna; notes by Iury Simoes-Sousa and Tilly Woods

1 Introduction

In this lecture, we shift our focus onto how to deal with uncertainty in data. Most of the methods covered until now assume that the data is the ground truth, but in the real world we are always dealing with error bars associated with the instrumentation, pre-assumptions around the data sampling and numerics. The methods looked at here will enable us to take into account this uncertainty.

2 Probabilistic graph models

Probabilistic graph models are a way to introduce something about 'uncertainty'—something that the methods we have looked at so far have not accounted for. This uncertainty could come from the observations or from the model itself. There can be uncertainty associated with a model either because we do not know an equation (e.g., the equation of state of the ocean is just a Taylor expansion because we do not know what the full equation should be), or because the equations result from some simplifying assumptions.

Probabilistic graph models can also help us deal with nonlinear interactions and mulitmodality, unlike most of the models we have seen so far. For example, in climate, probabilistic graph models enable us to come up with conditional probabilities of an event happening in response to multiple drivers, rather than being restricted to considering a single forcing.

Example: Reanalysis and ensemble Kalman filter If we use time-dependent primitive equations to produce an ensemble of model runs, the difference between the ensemble member will increase over time due to the uncertainty associated with the model. By applying some observations that we have, the ensemble members get brought closer together again (uncertainty is reduced). This idea of using data to improve model predictions is called Kalman filtering. We will go into more detail about these ideas later. Figure 1 shows this process applied to 20th century reanalysis, where an atmospheric model is used to model sea surface pressure evolution through time. The different lines are different ensemble members. The left panel shows the trajectories just before applying observations. The trajectories are spread out, due to uncertainty introduced by the model. The right panel shows how the trajectories are brought closer together when observations are applied.

Example: Multimodality Probabilistic graph models can capture situations which are multimodal, such as the North Atlantic eddy-driven jets shown in Figure 2.

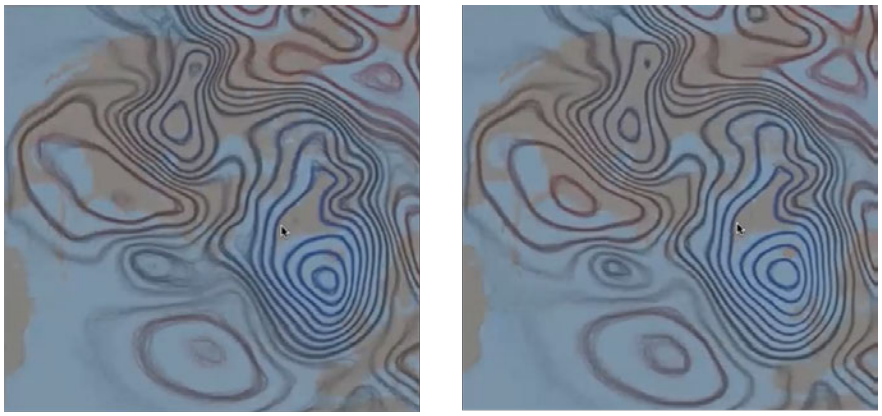


Figure 1: 20th Century Reanalysis data showing sea surface pressure for different ensemble members both before and after applying observations. (From this video: https://vimeo.com/178892173)

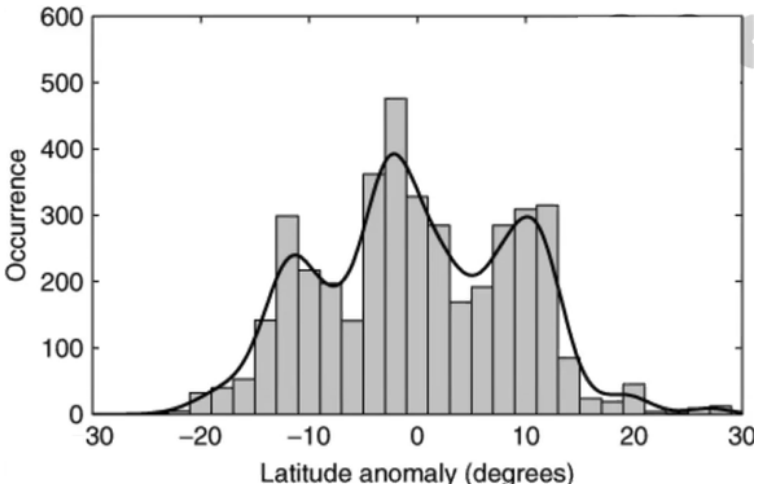


Figure 2: North Atlantic eddy-driven jet [3].

2.1 Probability rules

The methods we will look at here will be based on the following simple probability rules. Let X, Y be random variables. Then

- **Sum rule**
$$P(X) = \sum_Y P(X, Y), \tag{1}$$
where $P(X)$ is the marginal probability of random variable X and $P(X, Y)$ is the joint probability.

- **Product rule**
$$P(X, Y) = P(Y|X)P(X), \tag{2}$$
where $P(Y|X)$ is the conditional probability of Y given X.

- **Symmetry**
$$P(X, Y) = P(Y, X). \tag{3}$$

- **Bayes' theorem (/rule)**
$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} = \frac{P(X|Y)P(Y)}{\sum_Y \big(P(Y|X)P(X)\big)}, \tag{4}$$
where the second equality follows from the sum and product rules. Bayes' theorem can be understood as taking an initial guess (the prior $P(Y)$) for the probability distribution of Y, then bringing in some extra information/data (the likelihood $P(X|Y)$ to give an improved guess of the probability distribution (the posterior $P(Y|X)$, ie. the probability distribution of Y given the extra information represented by X). This is the basis of probabilistic graph models.

2.2 Directed graph models

Here we will consider acyclic directed graph models [2], where the nodes are variables (velocity, sea surface temperature etc.) and the edges are direct influences. The edges are arrows, with the direction showing the direction of influence (Figure 3). A key part of the directed graph model is that we can get all the information about variable x_i from the parents x_{parents} without needing to know anything about the previous ancestors x_{ancestor}. As shown in Figure 3, the parents are the nodes from which direct arrows go to x_i. We need only these direct influences to know the information about x_i, so x_i is independent of x_{ancestor} given x_{parent}.

Note that in this directed graph theory the graph cannot be cyclic.

2.3 Markov chains

A simple example of a directed graph is a Markov chain (Figure 4), which considers the evolution of a variable over time, in discrete steps. Suppose we have variables $x_1, x_2, ..., x_N$, where x_n is the value of a given variable (e.g., temperature) at time n. These variables could be scalars or state vectors. In a Markov chain, the variable at time n depends only on the variable at time $n-1$, as demonstrated in Figure 4. By using our probability rules, the joint probability of having our variables at each time being in a given state is

$$p(x_1, x_2, ..., x_N) = p(x_1) \prod_{n=2}^{N} p(x_n|x_{n-1}). \tag{5}$$

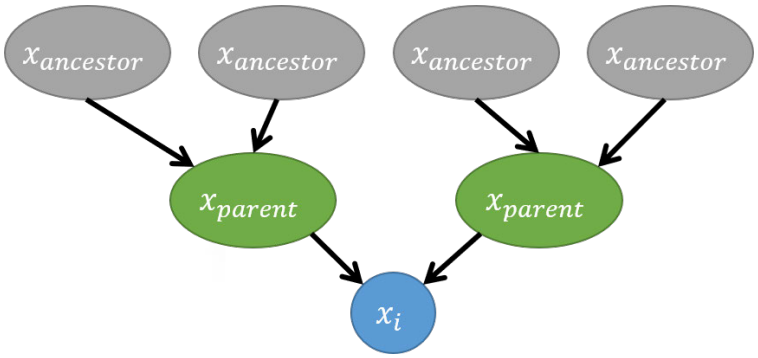


Figure 3: An example of a directed graph.

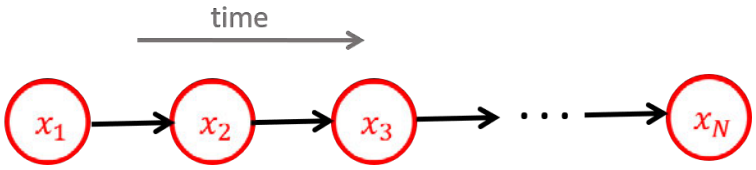


Figure 4: Markov chain.

The $p(x_1)$ term comes from the fact that x_1 has no parents, so is not influenced by any of the other x_n. All the other x_n $(n = 2, ..., N)$ depend on x_{n-1}, leading to the conditional probability terms $p(x_n|x_{n-1})$ in the product.

2.4 Hidden variable/latent variable

The basic Markov chain is very restrictive, making a lot of simplifying assumptions about the dynamics of the system. The major assumption is that the variable (eg. temperature) at time n is only influenced by the state of the variable at time $n-1$. However, reality is often much more complicated than this. To loosen the restriction and introduce some dependence on all past times, we can use **hidden/latent variables**. These are variables that are not directly observed but are inferred from a 'mathematical' model. The resulting model is called a **hidden Markov model**.

Instead of assuming that the observations x_n follow a Markov process, we assume that the hidden variables z_n follow a Markov process, and that the observation x_n and the hidden variable

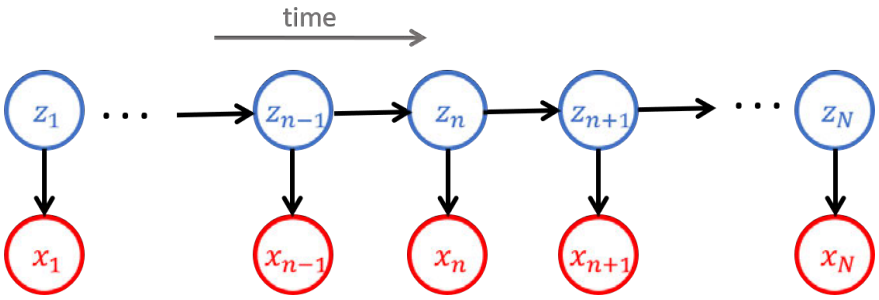


Figure 5: Hidden Markov model. $z_1, ..., z_N$ are the hidden variables, which follow a Markov process, and $x_1, ..., x_N$ are the observations.

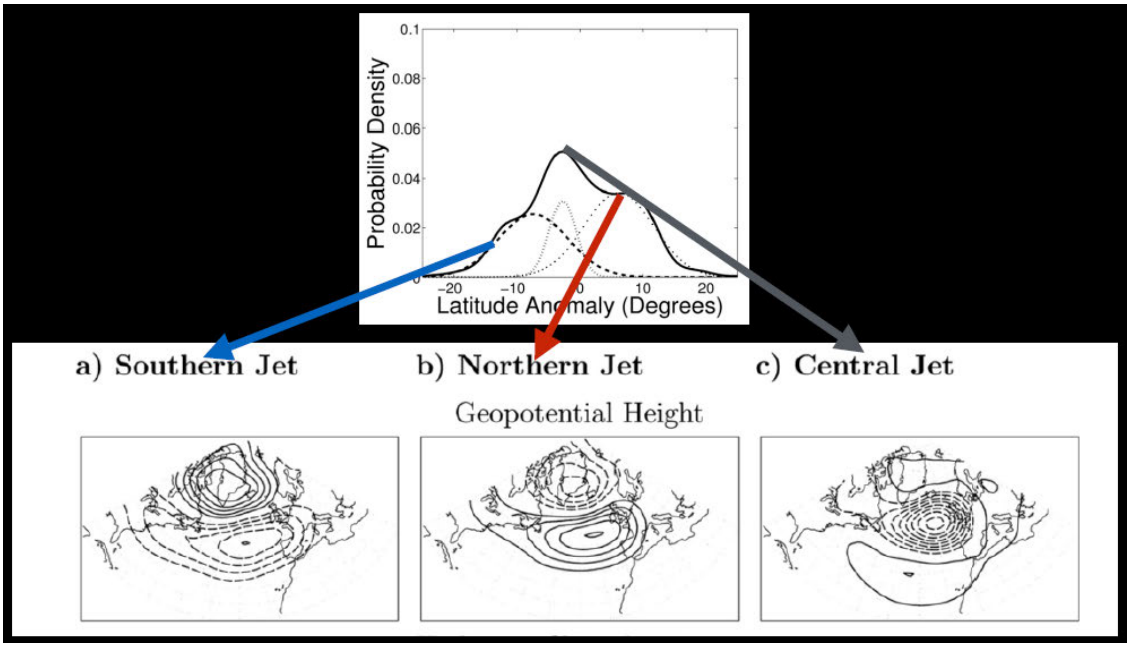


Figure 6: States of North Atlantic eddy-driven jet stream and the probability distributions of the latitude anomaly of the jet in each of these states [1].

z_n influence each other, as shown in Figure 5. Including these hidden variables means that we are allowing observation x_n to depend on more than just the observation x_{n-1} at the previous timestep. Each z_n, and hence x_n, is influenced by all of $x_1, ..., x_{n-1}$, since all of these observations are fed into the hidden variable Markov chain before step n.

At each time n, our hidden variable z_n and observation x_n could be in any one of a number of 'states', corresponding to different states of the system. To clarify what we mean by this, Figure 6 gives an example of the possible states of North Atlantic eddy-driven jet stream. The system transitions between these states with certain probabilities (called transition probabilities - discussed shortly).

To describe the hidden variables model mathematically, we would like to know the joint distribution

$$p(\mathbf{x}, \mathbf{z}) = p(x_1, ..., x_N, z_1, ..., z_N) = p(z_1) \prod_{n=2}^{N} p(z_n|z_{n-1}) \prod_{m=1}^{N} p(x_m|z_m). \tag{6}$$

To find the joint distribution, we need to know the following:

- Transition probabilities A, where $A_{jk} = p(z_n = k|z_{n-1} = j)$ is the probability that the latent variable transitions from state j in one timestep to state k in the next timestep (see Figure 7). We assume that the transition probabilities are the same of each n.
- Emission probabilities $\mathbf{\Phi}$, where $\Phi_{jk} = p(x_m = k|z_m = j)$, which tell us the probability of an observable x_m being in a certain state given given the state of its associated hidden variable z_m.
- Prior/marginal distributions $\mathbf{\Pi}$, where $\Pi_k = p(z_1 = k)$, i.e., the probability that our hidden variable will be in a given state at the initial time.

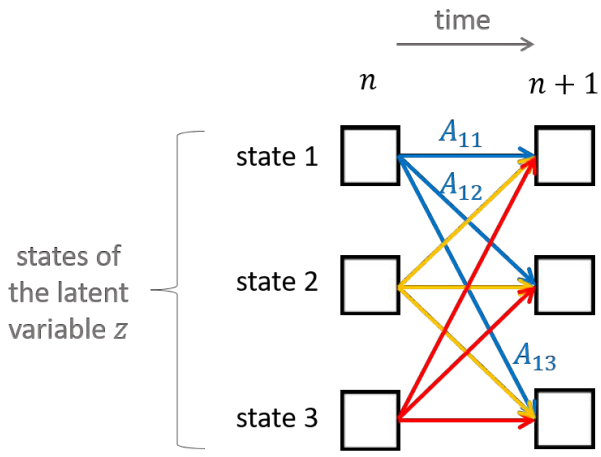


Figure 7: Transition probabilities A_{jk} between different states of the latent variable z, with three states used here for illustrative purposes.

The transition probabilities A, emission probabilities Φ and prior distributions Π—which are currently unknown—completely describe the statistics of the hidden variables. Hence, instead of thinking of the joint distribution as a function of the observations and the hidden variables, we can think of it as a function of the observations $\mathbf{x}$ and the parameters $\theta = \{A, \Phi, \Pi\}$: $p(\mathbf{x}, \theta)$.

In theory, we now have all the pieces of the puzzle, but we still do not know the values of the parameters θ. In order to calculate the full joint distribution $p(\mathbf{x}, \theta)$, we need to find the set of parameter values that will best fit the set of observations we have (given that we are assuming the hidden variables follow a Markov process). We can carry out this optimisation to find $p(\mathbf{x}, \theta)$ using the expectation-maximisation algorithm, as follows:

1. Make an initial guess for the value of the parameters $\theta_{\text{guess}} = \{A_{\text{guess}}, \Phi_{\text{guess}}, \Pi_{\text{guess}}\}$, informed by any existing knowledge we have of the system. In practice, people usually try the algorithm with a few different guesses to check that the result is not too sensitive to the guess and that we do not get stuck in a local minimum.

2. Calculate the expected likelihood.

3. Use a Lagrange multiplier to find the optimum value of the parameter values.

This process is carried out recursively.

The above assumes that we have no knowledge of the hidden variables. However, in many situations, we have some knowledge of the dynamics, for example the Navier-Stokes equation. This knowledge tells us the transitions probabilities A, but there will still be some uncertainty associated with it.

Even with some knowledge of the dynamics, we still need to run recursively through the algorithm, which is effectively performing a discrete version of a Kalman filter: taking a weighted average between the observations and what our model predicts. That is, the observations and our knowledge of the dynamics are used in combination to give us the best prediction of the joint probability distribution we are looking for. Figure 8 demonstrates this weighting graphically. We can use our model for the hidden variables to create a prediction (orange) based on a prior estimate (white). The uncertainty in the model means that applying the model increases uncertainty, so the prediction has greater uncertainty that the prior. To narrow the uncertainty, use the observations

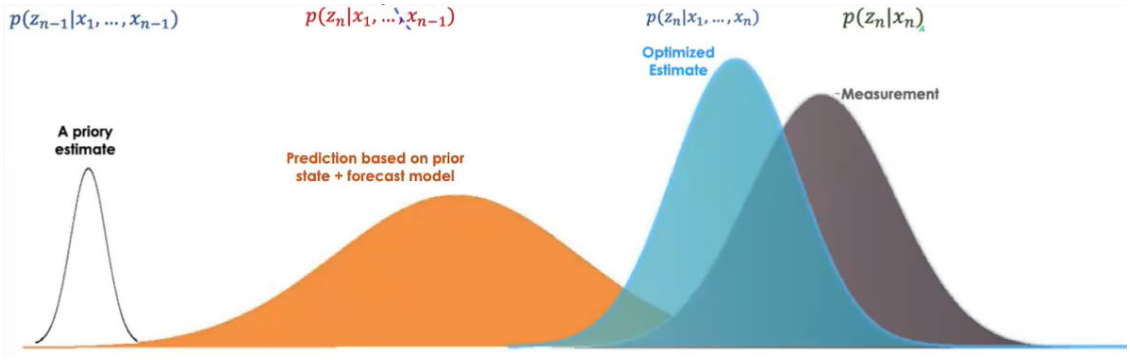


Figure 8: Demonstration of the hidden Markov model.

we have (purple), resulting in an optimised prediction (blue) which is a weighted average of the model and the observations. We can choose what weighting to use depending on how much we want to prioritise the observations vs. the model.

3 Summary

Uncertainty—whether that be uncertainty in the data or uncertainty in a model—can be taken into account by using probabilistic graph theory, in particular Markov chains or hidden Markov models. The key idea behind these methods is to use the data we have to maximise the likelihood (the probability of the system being in a certain state given certain information/observations), narrowing our uncertainty about what state the system is in. This can be done with no knowledge of a model for the physical system of interest, but any model information we do have (e.g., knowing the Navier-Stokes equations for a fluid) can be fed into the probabilistic graph model framework to improve our predictions. That is, modelling and data can be combined to give us a better understanding of the system than we would get by using either the model or data in isolation.

References

[1] C. Franzke and T. Woollings, *On the persistence and predictability properties of north atlantic climate variability*, Journal of Climate, 24 (2011), pp. 466–472.

[2] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann Publishers, 1988.

[3] T. Woollings, A. Hannachi, and B. Hoskins, *Variability of the North Atlantic eddy-driven jet stream*, Quarterly Journal of the Royal Meteorological Society, 136 (2010), pp. 856–868.

GFD 2022 Lecture 8: Sparse Regression—Finding Equations From Data

Laure Zanna; notes by Claire Valva and Rui Yang

1 Introduction

In this lecture, we apply some of the concepts of optimization and sparsity from the previous two lectures to the pursuit of extracting dynamical operators from data. Instead of approximating large dynamical operators from data — such as the Koopman operator (Lecture 3) or LIM (Lecture 5) — we now want to derive sparse and (ideally) easily interpretable **equations** for the dynamics of timeseries directly from data. This is ultimately motivated by the assumption that the physical equations driving the systems we study are essentially given by the laws of physics and thus sparse. We want to enforce sparsity to avoid overfitting data and instead use data to guide us towards physics. Here we summarize two approaches, Sparse Linear Regression (§2) and Sparse Identification of Non-Linear Dynamics (SINDy, §3), and provide an example of a third approach, Sparse Bayesian Regression (§4).

2 Sparse Linear Regression

Suppose we have some $m \times n$ data matrix D with columns $\mathbf{d}_j$ (and entries d_{ji}), where each column is our set of data points at a given snapshot in time, with time evolving from left to right. In sparse linear regression, we are aiming for the best prediction $\hat{\mathbf{y}}_i$ of the 'truth' $\mathbf{y}_i$ from the choice of β_j, given the data matrix D, using minimal nonzero β_j for the following problem:

$$\hat{\mathbf{y}}_i = \beta_0 + \sum_{j=n}^{p} \beta_j d_{ji}$$

To find β_j, the most obvious choice would be to minimize the error function $\epsilon(\beta_k) = \hat{y}_i - \beta_0 - \sum_{j=n}^{p} \beta_j x_{ji}$ in the ℓ^2 sense (the least squares error). However, the ℓ^2 norm does not promote sparsity, leading to the common usage of the following two loss functions for β_k do prioritize sparsity:

- **Ridge regression** is essentially an ℓ^2 minimization with an ℓ^2 penalty on β_k with tuning parameter $\lambda > 0$.

$$\min \left[\left(\sum_{i=1}^{m} (\hat{\mathbf{y}}_i - \beta_0 - \sum_{j=1}^{n} \beta_j x_{ji} \right)^2 + \lambda \sum_{i=1}^{n} \beta_i^2 \right]$$

- **LASSO (Least Absolute Shrinkage and Selection Operator)** is similar, but the penalty on β_k

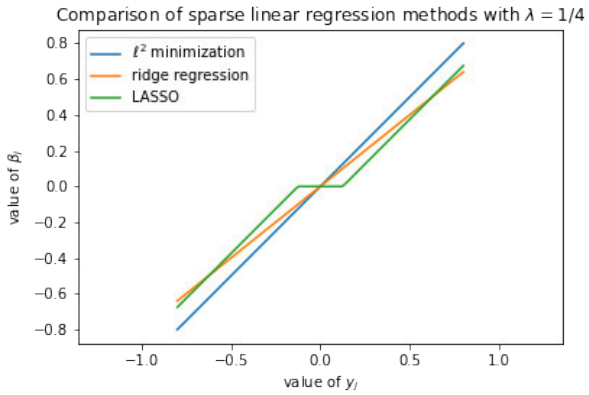


Figure 1: Example LASSO and ridge regression results for the following minimal problem. Let $n = m$ and suppose that X is the identity matrix and $\beta_0 = 0$. Then for ridge regression, we will find that $\beta_j = \frac{y_i}{1+\lambda}$, and for LASSO we get that $\beta_j = y_i - \lambda/2$ if $y_j > \lambda/2$, $\beta_j = y_i + \lambda/2$ if $y_j < -\lambda/2$, and 0 if $|y_i| \leq \lambda/2$.

will be an ℓ^1 penalty.

$$\min \left[\left(\sum_{i=1}^{m} (\hat{\mathbf{y}}_i - \beta_0 - \sum_{j=1}^{n} \beta_j x_{ji} \right)^2 + \lambda \sum_{i=1}^{n} |\beta_i| \right]$$

This norm is also sometimes referred to as 'soft-threshholding'.

3 Sparse Identification of Nonlinear Dynamics (SINDy)

The starting assumption of SINDy is that we are trying to understand some nonlinear dynamical system $\frac{d\mathbf{x}}{dt} = \mathcal{M}(\mathbf{x})$, which can be represented by a finite sum of functions φ_k that depend on $\mathbf{x}$, i.e., we have:

$$\frac{d\mathbf{x}}{dt} = \mathcal{M}(\mathbf{x}) \approx \sum_{k=1}^{p} \varphi_k(x) g_k$$

where g_k are weighting parameters.

We again will have $m \times n$ data matrix D with columns $\mathbf{d}_j$, where each column is our set of data points at a given snapshot in time, with time evolving from left to right. From this, we will want to construct an $m \times n$ data matrix $\dot{D}$ that is an approximation of the temporal derivatives of D, i.e., $\dot{d}_{ji} = \dot{x}_i(t_j)$. We will also construct a function library of φ_k which depend on the data, i.e., we may have that $\varphi_k = x_j^2$ or $\varphi_k = \cos(x_j)$. We will assume that we have p functions in our library. We will then solve the following problem for the vector G, which is the weighting matrix of ϕ_k:

$$\dot{D} = \Phi(D)G = \begin{pmatrix} \phi_1(d_1) & \phi_2(d_1) & \dots & \phi_p(d_1) \\ \phi_1(d_2) & \phi_2(d_2) & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \phi_1(d_m) & \phi_2(d_m) & \dots & \phi_p(d_m) \end{pmatrix}$$

In general, when solving for an optimal G, we will want G to be optimally sparse, usually, the loss function of G will try to minimize $\Phi(D)G - \dot{D}$ in the ℓ^2 sense with some sort of penalty on G (such as ℓ^1) and we can also add additional limitations of G. (SINDy, as implemented in [2] has its own optimization routines that use an expectation-maximization type algorithm.)

See [2, 3] for further examples and explanation, and [5] for an expansion of SINDy to spatial derivatives.

4 Relevance Vector Machines (RVMs)

We will consider an example of the usage of RVM (or iterative sparse Bayesian regression) in hunting for a mesoscale eddy closure.[1].

4.1 Mesoscale eddy closures

Consider x-component of the momentum equation in 2 dimensions with velocity $\mathbf{u} = (u, v)$, dissipation D, and a forcing term $\tilde{F} = -\frac{1}{\rho_0}\partial_x p + F_x$:

$$\partial_t u + \mathbf{u} \cdot \nabla u - fv = \tilde{F} + D$$

A high resolution model of this system (integrating on small scales) will be able to resolve the energy that comes from small scale eddies, however, in climate models we are often working on a much larger grid. To translate from small-scale grids to larger scale grids, the ideal "coarsened" equation (where $\overline{(\cdot)}$ denotes some average) should look like:

$$\overline{\partial_t u} + \overline{\mathbf{u} \cdot \nabla u} = \overline{\tilde{F}} + \overline{D}$$

However, the larger grid models will instead be computing:

$$\overline{\partial_t u} + \overline{\mathbf{u}} \cdot \nabla \overline{u} = \overline{\tilde{F}} + \overline{D} + S_x$$

where S_x, which is used to compensate for the fact that in general $\overline{\mathbf{u} \cdot \nabla u} \neq \overline{\mathbf{u}} \cdot \nabla \overline{u}$. The closure term $S_x = S_x(\overline{\mathbf{u}}, \kappa)$ should depend only on the averaged velocity field $\overline{\mathbf{u}}$ and some parameter κ, and in general, we aim to find some closed from $\hat{S}_x$ that is as close as possible to the "perfect" closure $S_x = \overline{\mathbf{u} \cdot \nabla u} - \overline{\mathbf{u}} \cdot \nabla \overline{u}$.

4.2 RVM usage in mesoscale eddy closures

In [7], Zanna and Bolton use relevance vector machines to seek an eddy closure term of the form $\hat{S_x} = \sum_k \phi_k(\overline{\mathbf{u}}) g_k$ from a library of functions ϕ_k with weights g_k using RVMs (see Figure 1). In this case, the expression from the RVM procedure explained 70% of the variance of the "perfect" closure S_x and extracted the symmetric stress tensor as well as shearing and stretching deformation terms with no a priori knowledge of the relevant physics.

In this example, as well as others, the closures found using the RVM procedure can be interpreted in terms of the expected physics: see [4] for nonlinear gradient models of turbulence or [1] for deformation based-parameterizations.

[1] An explanation of the RVM algorithm was skipped during this lecture in the interest of time. Some helpful references may include [6, 8].

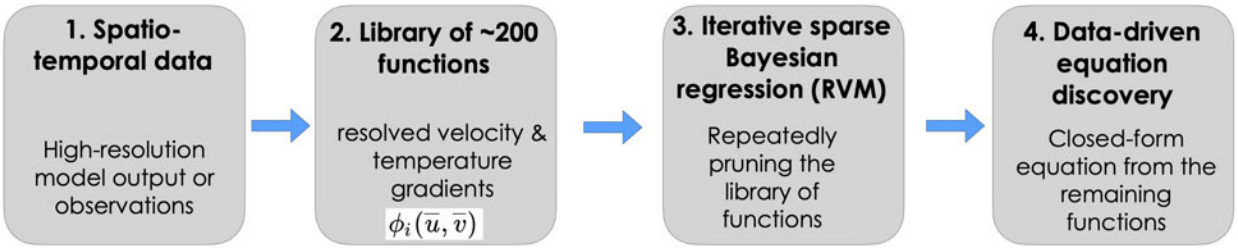


Figure 2: Outline of usage of relevance vector machines (RVMs) to find mesoscale eddy closures, as in [7].

References

[1] J. A. Anstey and L. Zanna, *A deformation-based parametrization of ocean mesoscale eddy reynolds stresses*, Ocean Modelling, 112 (2017), pp. 99–111.

[2] S. L. Brunton, J. L. Proctor, and J. N. Kutz, *Discovering governing equations from data by sparse identification of nonlinear dynamical systems*, Proceedings of the National Academy of Sciences, 113 (2016), pp. 3932–3937. Publisher: Proceedings of the National Academy of Sciences.

[3] K. Kaheman, J. N. Kutz, and S. L. Brunton, *SINDy-PI: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 476 (2020), p. 20200279. Publisher: Royal Society.

[4] B. T. Nadiga and F. Bouchet, *The equivalence of the Lagrangian-averaged Navier-Stokes-α model and the rational large eddy simulation model in two dimensions*, Physics of Fluids, 23 (2011), p. 095105. Publisher: American Institute of Physics.

[5] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, *Data-driven discovery of partial differential equations*, Science Advances, 3 (2017), p. e1602614. Publisher: American Association for the Advancement of Science.

[6] M. E. Tipping, *Sparse bayesian learning and the relevance vector machine*, The Journal of Machine Learning Research, 1 (2001), pp. 211–244.

[7] L. Zanna and T. Bolton, *Data-Driven Equation Discovery of Ocean Mesoscale Closures*, Geophysical Research Letters, 47 (2020), p. e2020GL088376.

[8] S. Zhang and G. Lin, *Robust data-driven discovery of governing physical laws with error bars*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 474 (2018), p. 20180305. Publisher: Royal Society.

GFD 2022 Lecture 9: Sparse Data Reconstruction and Increasing Predictability

Peter Schmid; notes by Ludovico Giorgini, Sam Lewin, Ruth Moorman, Kasturi Shah

1 Introduction

Here we cover two distinct topics building on the concepts of sparsity and forecasting introduced in previous lectures. We start with an inversion of our paradigm of sparsity in data driven methods through a discussion of Compressed Sensing (§2). Whilst the procedures outlined in Lecture 8 **seek** sparser, and thus in some sense more 'physical', dynamical operators from data, Compressed Sensing **exploits** the underlying sparsity of physics to reproduce signals from coarse sampling. The second part of this lecture (§3) revisits the idea of forecasting systems using forward dynamical operators (e.g., LIM and Koopman operators). We interrogate the causes of drift when using such operators to project system behavior forward in time and seek possible remedies for said drift. In other words, we ask the question: how should we truncate our dynamical system when generating forecasts and what are the effects of those truncation choices on predictability?

2 Compressed Sensing

The Nyquist-Shannon sampling theorem states that a signal must be sampled at at least twice its highest frequency for it to be uniquely and exactly reconstructed. Compressed sensing is a signal processing technique for acquiring and reconstructing a signal using fewer samples than the Nyquist-Shannon sampling theorem requires, if certain information regarding the signals sparsity is known.

The essential concept is as follows. Say we have a signal length N that is sparse in, for example, the Fourier basis such that it is well characterized by only $k \ll N$ frequencies. One way we could exploit this sparsity would be to conduct a Fourier transformation of the data and then discard all but the k dominant frequencies. If k is small (i.e., the signal is sparse), this seems quite wasteful, since it requires discarding the vast majority of the generated frequencies. Compressed sensing asks the question: can we use our knowledge of a systems sparsity in a certain basis to avoid such procedural waste? This would allow us to avoid sampling unnecessarily fine sampling of signals and reconstruct signals from coarse sampling.

Now for the mathematical derivation. Let's consider a signal $x \in \mathcal{R}^N$, which is $k-$sparse in a basis $\Psi \in \mathcal{R}^{N\times N}$. For simplicity, we will consider a signal x that is already k-sparse and then $\Psi = I$. We want to reconstruct this signal from $y \in \mathcal{R}^M$, a dense vector of randomly sampled observables with $k \leq M$ and $M \ll N$. The two vectors x and y are related by the sampling matrix $\Phi \in \mathcal{R}^{N\times M}$ as

$$y = \Phi x. \tag{1}$$

Since there are k non zero elements in x, $\text{rank}(\Phi) \geq k$. We don't know,however, where these non-zero elements are located, and we have to then choose Φ such that this condition is guaranteed for any arbitrary k-sparse vector. This is obtained by imposing that any submatrix $\tilde{\Phi} \in \mathcal{R}^{k \times M}$ of Φ has full rank. A wide range of random matrices satisfy this condition, for example, those generated from i.i.d. Gaussian distribution, Bernoulli distribution, subsamples FFT, etc.

Since $M < N$, the linear system in (1) is underdetermined and in order to solve it we have to impose a constraint which, in this case, is the sparsity of x. We then chose x as the vector, which minimizes the following loss function

$$||\Phi x - y||_2 + \lambda ||x||_0. \tag{2}$$

Minimizing the loss function with an l_0 norm is an NP-complex problem that is extremely hard to solve. The l_0 norm can be substituted by the l_1 norm at the price of increasing the minimum size of y, $N = O(k \log(N/k))$ which remains much smaller than N. We have to compute

$$\min_x \left[||\Phi x - y||_2 + \lambda x^T W x \right], \tag{3}$$

with $W = \text{diag}\left(\frac{1}{|x|}\right) \simeq \text{diag}\left(\frac{|x|}{x^2+\epsilon^2}\right)$, which allowed to write the l_1 norm as a l_2 norm. The loss function can be minimized over x iteratively

$$\begin{aligned} W^k &= \text{diag}\left(\frac{|x^k|}{(x^k)^2 + \epsilon^2}\right) \\ x^{k+1} &= (W^k)^{-1}\Phi^T(\Phi W^k \Phi^T)^{-1}\Phi^T y, \end{aligned} \tag{4}$$

and the searched sparse vector x is recovered.

3 Increasing Predictability

During this lecture series, we have spent much of our time seeking reduced order descriptions of high-dimensional dynamical systems. But it is worth considering what happens when we try and make future forecasts using these simplified models. One of the primary issues is that errors arising from the reduction of dimensionality will be propagated forward in time, leading to 'drift' away from the true trend and possibly even instability and blow-up.

To this end, consider a data-matrix D whose rows represent the temporal evolution of measurements of a particular observable. We can perform an SVD:

$$\overbrace{D}^{\dim\ n\times m} = \underbrace{U}_{n\times m} \overbrace{\Sigma}^{m\times m} \underbrace{\tilde{V}^T}_{m\times m} \tag{5}$$

in the usual manner. A reduced order description of the system is obtained as usual by *truncating*, or choosing the first r principal components, in which case Σ becomes an $r \times r$ matrix and we only retain the first r columns of U: let us denote the truncated principal components matrix

$$U_{\text{trunc}} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_r \\ | & | & & | \end{bmatrix}, \tag{6}$$

where the $\mathbf{u}_i$ are the principal components. Any state vector $\mathbf{x}$ (note this does not necessarily have to one of the observed state vectors in D!) can be approximated by some linear combination of

the principal components: $\mathbf{x} = U_{\text{trunc}}\mathbf{a}^T$ for some vector of coefficients $\mathbf{a}$. Suppose our dynamical system evolves according to the equation

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}). \tag{7}$$

Then we have approximately

$$\frac{d\mathbf{a}^T}{dt} \approx U_{\text{trunc}}^T \mathbf{f}(U_{\text{trunc}}\mathbf{a}^T). \tag{8}$$

Even if we know the vector field $\mathbf{f}$ exactly, the fact that it is acting on the approximation $U_{\text{trunc}}\mathbf{a}^T$ means that truncation errors will inevitably be propagated forward in time. When the columns of D represent gridded observations, we can think of the truncation procedure as essentially picking out the dominant large scale features of the system and removing the smaller scale behaviour. In many physical systems, energy is typically cascaded from the large scales to the small scales. However, if we cut-off these scales then this can result in a build up of energy at the cut-off point which can eventually feed back and cause the system to follow a different trajectory, or even destabilize completely. Below, we discuss **residualisation**, a method for handling the error, or residual, that is propagated through the system. We also introduce a practical method for approximating these higher order error terms know as **time-delay embedding**.

3.1 Residualisation

Consider the linear dynamical system,

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{9}$$

where x_1 represents retained structures and x_2 represents removed structures. The question before us now is whether we can write a dynamical system that is truncated in x_1 only, such that

$$\frac{dx_1}{dt} = A_{11}x_1. \tag{10}$$

To do so, we introduce a memory term. Said differently, we move from the differential equation above to the integrodifferential equation,

$$\frac{dx_1}{dt} = A_{11}x_1 + \int_0^t A_{12}e^{(t-\tau)A_{22}}A_{21}x_1(\tau)d\tau. \tag{11}$$

The inclusion of the memory term means that the integrodifferential equation is not an approximation and is an exact solution to (9). The components of the memory integral can be pieced apart as follows,

$$\underbrace{\int_0^t \overbrace{A_{12}}^{\text{big to small}} \underbrace{\underbrace{\overbrace{e^{(t-\tau)A_{22}}}^{\text{big to big}}}_{\text{dynamics of small scales}} \underbrace{\overbrace{A_{21}\underbrace{x_1(\tau)}_{\text{small scales}}}^{\text{small to big}}}_{\text{effect of large scales on small scales}}}_{\text{propagate small scales}} d\tau}_{\text{bring the effect of the small scales to the large scales}} = \int_0^t k(t-\tau)x_1(\tau)d\tau \tag{12}$$

where $k(t-\tau) = \sum k_1(t)k_2(\tau)$. There are several ways to obtain the k's, including by derivation, by approximation, by identification from data, or by machine learning.

A few extensions of the memory integral are worth mentioning:

- Noise and the fluctuation dissipation theorem

  Consider the noisy linear dynamical system,

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} \eta_1 \\ \eta_2 \end{pmatrix}, \tag{13}$$

  the differential equation in (10) becomes

$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = A_{11}x_1 + \eta_1. \tag{14}$$

  On incorporating noise into the integrodifferential equation, we obtain

$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = A_{11}x_1 + \int_0^t A_{12}e^{(t-\tau)A_{22}+\eta_2}A_{21}x_1(\tau)d\tau. \tag{15}$$

  Therefore, η_2 "adds" to the noise η_1, and we recover the fluctuation dissipation theorem.

- Memory integral using quantum mechanics

  The memory integral can be placed in the context of the very famous paper by Weyl & Feynman, which expressed the memory integral using a forward and backward Fourier transform. This essentially involves changing the limits of the integral to be $\pm\infty$; Weyl & Feynman also applied an absorbing boundary condition at $\pm\infty$. The integrand then has the form

$$e^{-iky}P(k)e^{ikx}dxdy, \tag{16}$$

  where $P(k)$ is the "symbol" of our dynamical system and and the above expression is a representation of the Weyl operator.

A parting note on matrix partitioning. The partitioning of a matrix $\mathbf{A}$ into components A_{11}, A_{12}, A_{21}, and A_{22} can be used to eliminate certain parts of the matrix and is a useful tool. For instance, suppose we want to eliminate A_{12}, A_{21}, and A_{22}. Assuming we know the relationship between x_1 and x_2, we can define the **Schur component**, S, as follows,

$$S = A_{11} - A_{12}A_{22}^{-1}A_{21}. \tag{17}$$

Hence, the Schur component partitions the matrix by lumping A_{12}, A_{21}, and A_{22} onto A_{11}.

3.2 Time-delay embedding

Consider again equation (11). Since $\mathbf{x}_1$ is just a truncation of the state vector, we can identify it with the vector $\mathbf{a}$ as in equation (6). Then, if we discretize the dynamical system in time $\{\mathbf{a}_1, \mathbf{a}_2, \ldots\}$, perhaps the simplest way to incorporate the memory term in equation (11) is to write it component-wise as

$$a_{t+1_i} = \mu a_{t_i} + \lambda a_{t-1_i} + \ldots + \text{residual}, \tag{18}$$

where the residual represents the deviation from the true solution. The goal is then to find a relationship between the state vector $\mathbf{a}^{t+1}$ at timestep $t+1$ and the previous timesteps.

To proceed, we start as usual with the data matrix

$$D_{n\times m} = \begin{bmatrix} | & | & & | \\ \mathbf{d}_1 & \mathbf{d}_2 & \ldots & \mathbf{d}_m \\ | & | & & | \end{bmatrix}, \tag{19}$$

where the columns represent the state vectors and the rows represent their evolution in time. We can form a new matrix by stacking columns 1 to $m-2$ on top of columns 2 to $m-1$, on top of columns 3 to m:

$$D'_{3n\times(m-2)} = \begin{bmatrix} | & | & & | \\ \mathbf{d}_1 & \mathbf{d}_2 & \dots & \mathbf{d}_{m-2} \\ | & | & & | \\ | & | & & | \\ \mathbf{d}_2 & \mathbf{d}_3 & \dots & \mathbf{d}_{m-1} \\ | & | & & | \\ | & | & & | \\ \mathbf{d}_3 & \mathbf{d}_4 & \dots & \mathbf{d}_m \\ | & | & & | \end{bmatrix}. \tag{20}$$

This procedure is called **time-delay embedding** and the matrix D' is, by construction, a **Hankel matrix**, in this case with embedding number 3. We denote $D' = \text{Hank}_3(D)$. Such methods date back to the classical work of Ruelle & Takens [2] and are sometimes called Ruelle-Takens embeddings. Larger embedding numbers r can be used by stacking $m-r$ columns sequentially in the obvious manner analagous to the above, though the scale of the problem becomes large very quickly. We note that there are various ways to handle this, either by using computational algorithms that are parallelized to run on multiple processors simultaneously, or by some efficient and dynamics-preserving compression of the Hankel matrix D' using methods such as locality sensitive hashing (see e.g., [1]).

Why have we done this? Remember from lecture 3 that we can use Koopman methods to find a linear map from column to column of a given data matrix, i.e., $\mathbf{d}_t \mapsto \mathbf{d}_{t+1}$. With our time-delay embedded matrix, the same procedure simultaneously gives us linear maps $\mathbf{d}_t \to \mathbf{d}_{t+1}$, $\mathbf{d}_{t+1} \mapsto \mathbf{d}_{t+2}$ and $\mathbf{d}_{t+2} \mapsto \mathbf{d}_{t+3}$. Thus, we end up with a way of writing $d_{t+3_i} = R_{3_i} d_{t+2_i} + R_{2_i} d_{t+1_i} + R_{1_i} d_{t_i}$ as desired, where the R_{k_i} are the eigenvalues of the Koopman operator matrix, or equivalently the relevant companion matrix S computed using the procedure outlined in lecture 3. This method is sometimes described as 'higher order Koopman.'

3.3 Discussion

Time-delay embedding can be especially effective when we have a long time history of measurements, but only a few observables or measurement points. This means that the data matrix D is long and skinny, that is, $n \ll m$. In a Koopman setting, the system is decomposed into single frequency modes. However, the captured frequencies will be limited by the fact that D is low rank: $\text{Rank}(D) \leq n \ll m$. The higher order Koopman approach using a Hankel matrix consisting of stacked columns of D is often found to be very effective for capturing higher frequency dynamics in the system with limited measurement points.

Finally, we point out that the above method of residualisation was strictly only applicable for linear dynamical systems described by (9). The general approach for nonlinear dynamical systems involves computing a Liouville operator for the system with is associated with the matrix A: such methods are part of the so-called 'Mori-Zwanzig formalism' [3].

References

[1] P. Indyk, R. Motwani, P. Raghavan, and S. Vempala, *Locality-preserving hashing in multidimensional spaces*, in Proceedings of the twenty-ninth annual ACM symposium on Theory

of computing, 1997, pp. 618–625.

[2] D. Ruelle and F. Takens, *On the nature of turbulence*, Les rencontres physiciens-mathématiciens de Strasbourg-RCP25, 12 (1971), pp. 1–44.

[3] R. Zwanzig, *Nonequilibrium statistical mechanics*, Oxford university press, 2001.

GFD 2022 Lecture 10: Machine Learning Tools

Laure Zanna; notes by Iury Simoes-Sousa, Claire Valva, Tilly Woods, and Rui Yang

1 Introduction

In this lecture, we discuss machine learning tools. We include a basic overview of neural networks as well as brief mentions of other methods and examples. As motivation, we will return to the mesoscale eddy closure problem — for an explanation of this idea, see 3.1 in lecture 8. For this problem and in general, we are looking for a sparse representation of the system that is both generalizable and interpretable. We then ask if machine learning can capture what we have lost by in decreasing spatial resolution in the eddy closure problem (see 3).

2 The (in)Famous Neural Network

One of the methods that has been increasingly used to find sparse representations of many systems is the (in)famous neural network, that has been used in fields including computer vision, fraud detection, customer recommendation engines etc.

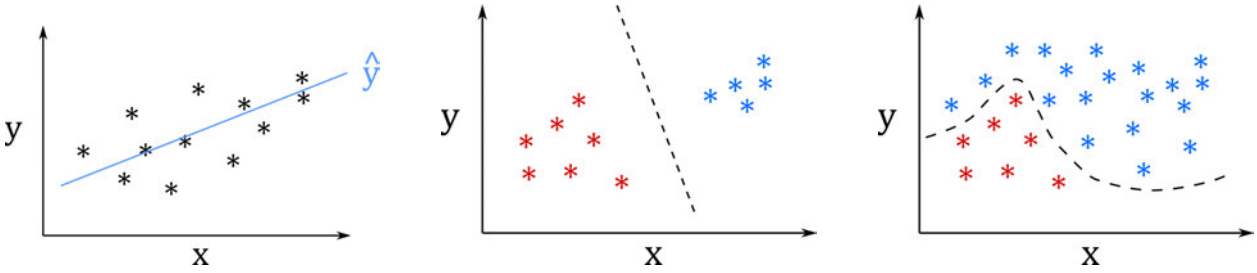
2.1 A basic overview of a single-layer neural networks

Starting from the fact that the main goal with this lecture is to show the whole picture and basic concepts of what a neural network is, we can start with the following representation of a single-layer neural network:

$$\hat{y} = z = \mathcal{G}\underbrace{\left(\sum_{i=1}^{n} w_i x_i\right)}_{\sigma}, \tag{1}$$

where $\hat{y}$ is the prediction, (which in a single layer network is equivalent to z, the result from the single neural network layer), x_i are the input (state variables), w_i are the weights for the summation and $\mathcal{G}$ is the activation function. Several common activation functions are:

- $\mathcal{G}(\sigma) = a\,\sigma$ Linear,
- $\mathcal{G}(\sigma) = \text{sign}(\sigma)$ Step function (classification),
- $\mathcal{G}(\sigma) = (1 + e^{-\sigma})^{-1}$ Sigmoid,
- $\mathcal{G}(\sigma) = \max(0, \sigma)$ ReLU (Rectified Linear Unit).



(a) $\mathcal{G}(\sigma) = a\,\sigma$ (b) $\mathcal{G}(\sigma) = \text{sign}(\sigma)$ (c) $\mathcal{G}(\sigma) = (1 + e^{-\sigma})^{-1}$

Figure 1: Graphical representation of different choice of activation functions for different problems.

For the linear activation function we recall the simple regression problem, graphically represented as in Figure 1a. The choice of the activation function can determine the type of problem. For a classification problem, one may use a step function (Figure 1b). Alternatively a sigmoid function allow a nonlinear classification as those similar to what is represented in Figure 1c.

In another words, the single-layer neural network has an output that will be an activation function applied to the sum of all inputs under tuned weights. We can represent this graphically as in Figure 2.

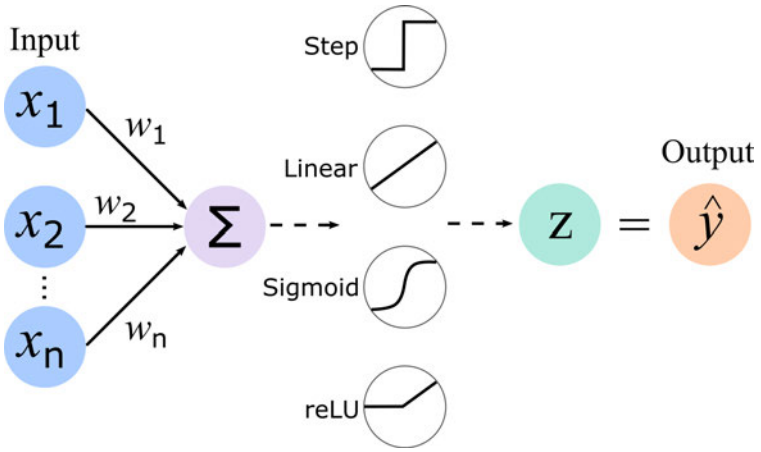


Figure 2: Graphical representation of a one-layer neural network.

2.2 Multi-layer neural networks

Partially owing to their simplicity, single-layer neural networks do not always capture the nonlinear dynamics of the system. More often, multilayered neural networks are employed, expressed as:

$$\begin{aligned} \hat{y}_i &= \mathcal{G}\left(\sum_{j=1}^{d} z_j w_{j,i}^{(z)}\right), \\ z_i &= \sum_{j=1}^{m} w_{j,i} x_j, \end{aligned} \tag{2}$$

where $\hat{y}_i$ are the predictions, z_i are the results the nodes ("neurons") from each hidden layer, m is the number of hidden layers and d is the number of nodes in the hidden layers (kept constant

here). In this case, we use the same activation function for all layers, for the sake of simplicity. We graphically represent a multi-layer network with these objects in Figure 3.

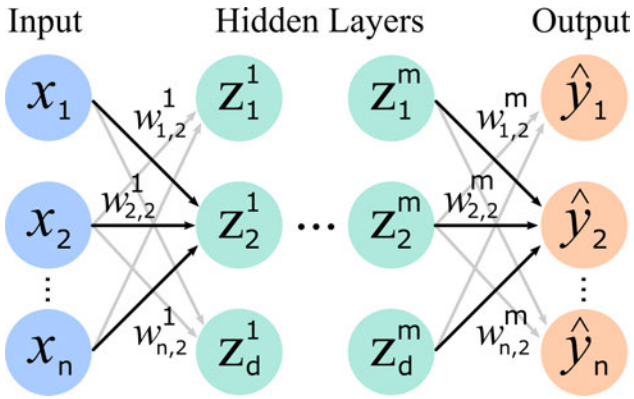


Figure 3: Graphical representation of a multi-layer neural network with the same input and output dimension.

We note that the same method could also be used for the purposes of a reconstruction problem, where the input and output dimension are not the same (despite it being the same in the figure).

Once we construct this "machine", the weighting for the best prediction must be determined. There are many methods for this optimization problem, but the most common is the gradient descent applied to minimize a loss function ($\mathcal{L}$), expressed by the mean squared error of predicted output values compared to the true value.

$$\mathcal{L}(y, \hat{y}) = \sum_i (y_i - \hat{y}_i)^2 \tag{3}$$

3 Mesoscale Eddy Parameterization Examples

Recall that in the eddy parameterization problem, we are looking for a way to low-resolution model output to "match" high resolution model output. Given the momentum equation in 2 dimensions with velocity $\mathbf{u} = (u, v)$, dissipation D, and a forcing term $\tilde{F}$:

$$\partial_t u + \mathbf{u} \cdot \nabla u - fv = \tilde{F} + D$$

When we decrease the resolution of the model the ideal "coarsened" equation (where $\overline{(\cdot)}$ denotes some average) should look like:

$$\overline{\partial_t u} + \overline{\mathbf{u} \cdot \nabla u} = \overline{\tilde{F}} + \overline{D}$$

However, the larger grid models will instead be computing:

$$\overline{\partial_t u} + \overline{\mathbf{u}} \cdot \nabla \overline{u} = \overline{\tilde{F}} + \overline{D} + S_x$$

where S_x which is used to compensate for the fact that in general $\overline{\mathbf{u} \cdot \nabla u} \neq \overline{\mathbf{u}} \cdot \nabla \overline{u}$. As such, the "perfect" closure would be $S_x = \overline{\mathbf{u} \cdot \nabla u} - \overline{\mathbf{u}} \cdot \nabla \overline{u}$.

Comparison of physics- and neural network-based parameterizations In [5], the authors compare physics- and neural network-based parameterizations of subgrid-scale processes in ocean models. In figure 5, we see that the low-resolution model (grey line) did not achieve the wanted energy

spectra, while the machine-learning methods did. However, one should note that the neural-net parameterization was not entirely necessary as the physics-based parameterization (BSCAT) worked just as well. The BSCAT parameterization from [3] writes the closure term as a function of eddy energy q, $S(q) = -\nu\Delta q$, which can be thought of as essentially adding energy backscatter.

Stochastic eddy parameterization In an alternative approach, a stochastic parameterization using a convolutional neural network has been proposed [2]. In this work, Guillaumin and Zanna propose a closure that is stochastic, and try to learn a closure in the form of a Gaussian probability distribution:

$$\overline{u} \to G(\overline{u}|\overline{u}, \overline{q}; \mu, \sigma), \quad \text{where } \mu \text{ is the mean, and } \sigma \text{ is the standard deviation.}$$

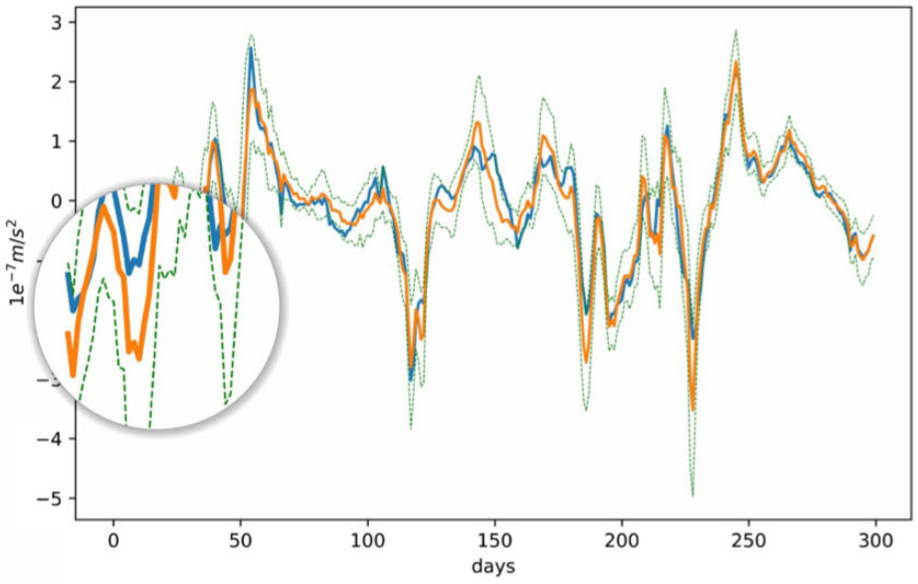
The authors trained the model on surface velocity data from a pi-control GFDL global climate model, and found predictions to perform fairly well 4. This parameterization generalized to generalize to other climate scenarios (such as increased CO_2) relatively well. However, the model does not predict areas with sea ice reliably — which is a sensible result as a model cannot learn what it was not shown. Recently, Zanna and collaborators (Pavel Perezhongin and Cheng Zhang) have begun to try to implement this stochastic closure "online" with the Modular Ocean Model 6 (MOM6) but there are issues that include numerics/stability, coupling, and tuning. (This points to some clear advantages of symbolic-regression that include ease of implementation without any clear performance disadvantages as compared to convolutional neural networks [6].)

4 Other Tools of Interest

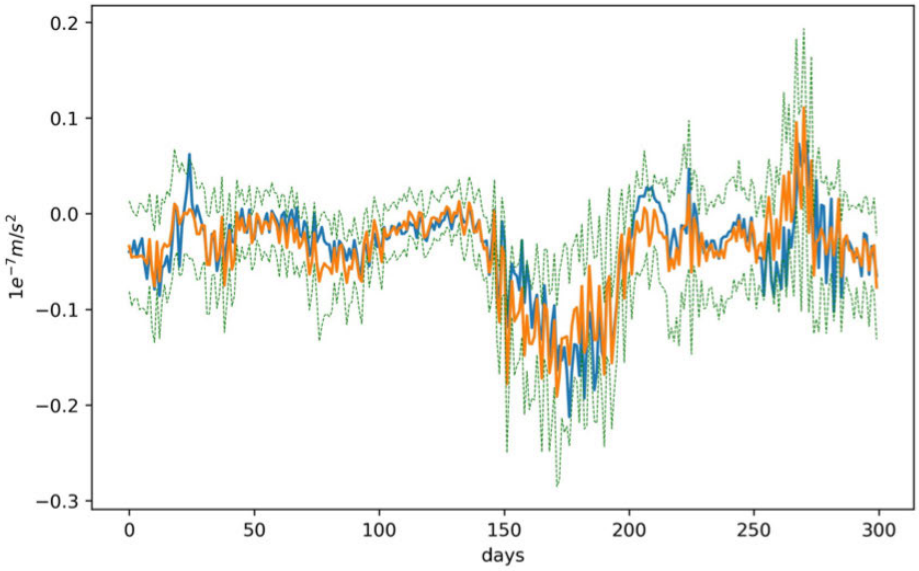
Briefly, two other machine learning tools of interest (among many) were mentioned.

Physics-informed neural networks (PINNs) PINNs are neural networks that are trained to solve supervised learning tasks while respecting any given laws of physics described by general nonlinear partial differential equations, and was initially described by [4]. A key component of PINNs are that the algorithm seeks to minimize two different loss functions: one of which minimizes the error in the prediction (the standard idea of a loss function) and the other that penalizes predictions that do not satisfy a given governing equation. This algorithm appears to both learn predictions relatively well as well as generalize decently.

Genetic programming The idea of genetic programming (a commonly used implementation of which is `gplearn` [1]) uses a similar approach to library-based symbolic regression like SINDy. From the documentation: "The algorithm begings by building a population of naive random formulas to represent a relationship between known independent variables and their dependent variable targets in order to predict new data. Each successive generation of programs is then evolved from the one that came before it by selecting the fittest individuals from the population to undergo genetic operations." Two other notable details that differentiate genetic programming from SINDY are that spatial derivatives are available and there can be human intervention at each successive stage of the algorithm.



(a) turbulent



(b) quiescent

Figure 4: Time series comparison of the zonal component of the subgrid momentum forcing at (a) a location dominated by turbulent behavior and (b) a more quiescent location for 300 days: true forcing (solid blue), mean of the predicted forcing (orange), and 95% confidence interval (green). (Figure from [2].)

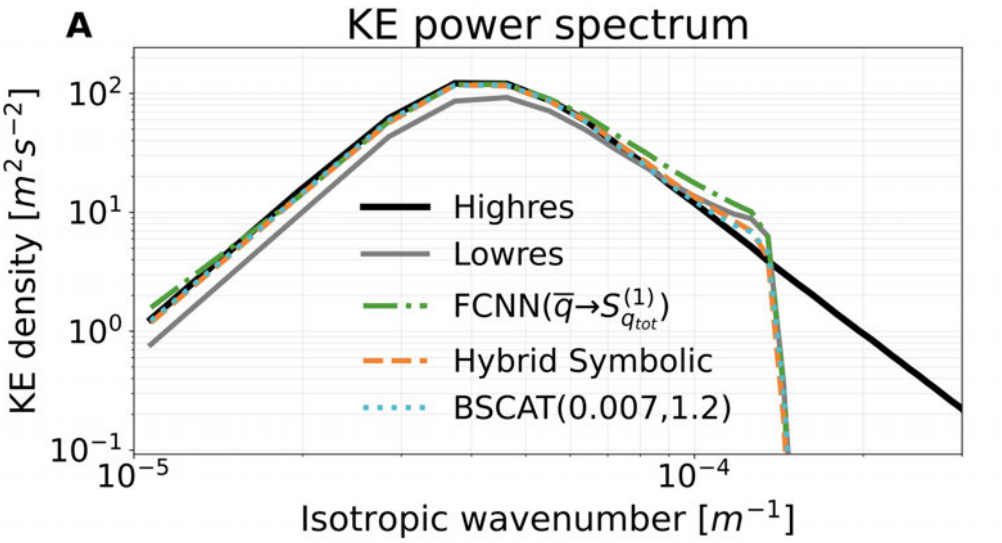


Figure 5: Power spectrum comparison for the eddy-closure problem between physics-based parameterizations (BSCAT) and machine learning parameterizations (FCNN and Hybrid Symbolic) from [5].

References

[1] *Genetic programming in python.* https://github.com/trevorstephens/gplearn.

[2] A. P. Guillaumin and L. Zanna, *Stochastic-Deep Learning Parameterization of Ocean Momentum Forcing*, Journal of Advances in Modeling Earth Systems, 13 (2021), p. e2021MS002534. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2021MS002534.

[3] M. F. Jansen and I. M. Held, *Parameterizing subgrid-scale eddy effects using energetically consistent backscatter*, Ocean Modelling, 80 (2014), pp. 36–48.

[4] M. Raissi, P. Perdikaris, and G. E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, Journal of Computational Physics, 378 (2019), pp. 686–707.

[5] A. S. Ross, Z. Li, P. Perezhogin, C. Fernandez-Granda, and L. Zanna, *Benchmarking of machine learning ocean subgrid parameterizations in an idealized model*, May 2022. Section: Oceanography.

[6] L. Zanna and T. Bolton, *Data-Driven Equation Discovery of Ocean Mesoscale Closures*, Geophysical Research Letters, 47 (2020), p. e2020GL088376.

Continental Shelf Waves Around a Pseudo-Iceland

Ruth Moorman

1 Introduction

Oscillations at frequencies lower than the inertial frequency, with periods of a few days to weeks, are a prominent feature of coastal seas the world over. These oscillations alter coastal conditions and may contribute to mixing and the exchange of tracers between shelf seas and the open ocean [11]. In high latitude oceans specifically, subinertial coastal waves have been noted as a potential driver influencing the overflow of dense waters from shelf seas into the abyssal ocean and the exposure of marine terminating glaciers to ocean heat via the lifting and lowering of thermoclines [7, 8, 22]. In general, improving our understanding of these subinertial coastal oscillations will assist both their characterization as potential drivers of coastal conditions in themselves, and aid their identification and removal from observations when they pose a potential aliasing effect.

Recently, Gelderloos et al. (2021) [7] identified and characterized low frequency coastal waves propagating along the Southeast Greenland shelf in a general circulation model. A feature of their simulations that they noted but did not study was the propagation of subinertial waves around Iceland (Figure 1). Wave modes propagating around the island are set by the circumference of the island, with the lowest alongshore mode (and most prominent mode visible in Figure 1) fitting exactly once into its circumference. Subinertial waves are also evident propagating away from the island along the mid-Atlantic and Greenland-Scotland ridges. We posit that the presence of these sizable ridges abutting the continental slope may scatter or otherwise deflect the energy of some wave modes away from the island, whilst others may remain bound to the island and exhibit a resonance.

The purpose of this work is to interrogate, using the most straightforward model possible, the effect of ridges abutting islands on an island's subinertial wave field. We seek to understand which modes can resonate around an Iceland-like island and which may be influenced by the presence of intersecting ridges. The selected "continental shelf wave" (CSW) model for coastally trapped subinertial waves, derived in Section 2, is linear, inviscid, and barotropic, yet nonetheless presents challenges. In particular, no universal dispersion relation exists for CSWs over arbitrary topography [9] and the few analytical solutions found to date apply to highly idealized topographies [3, 5, 13, 21, for example]. Thus, numerical methods are generally required to study CSWs under more realistic conditions. Prior to the last decade, these numerical methods involved iteratively searching for propagating modes within the 2D dispersion relation parameter space [1, 2, 10], a relatively time-consuming

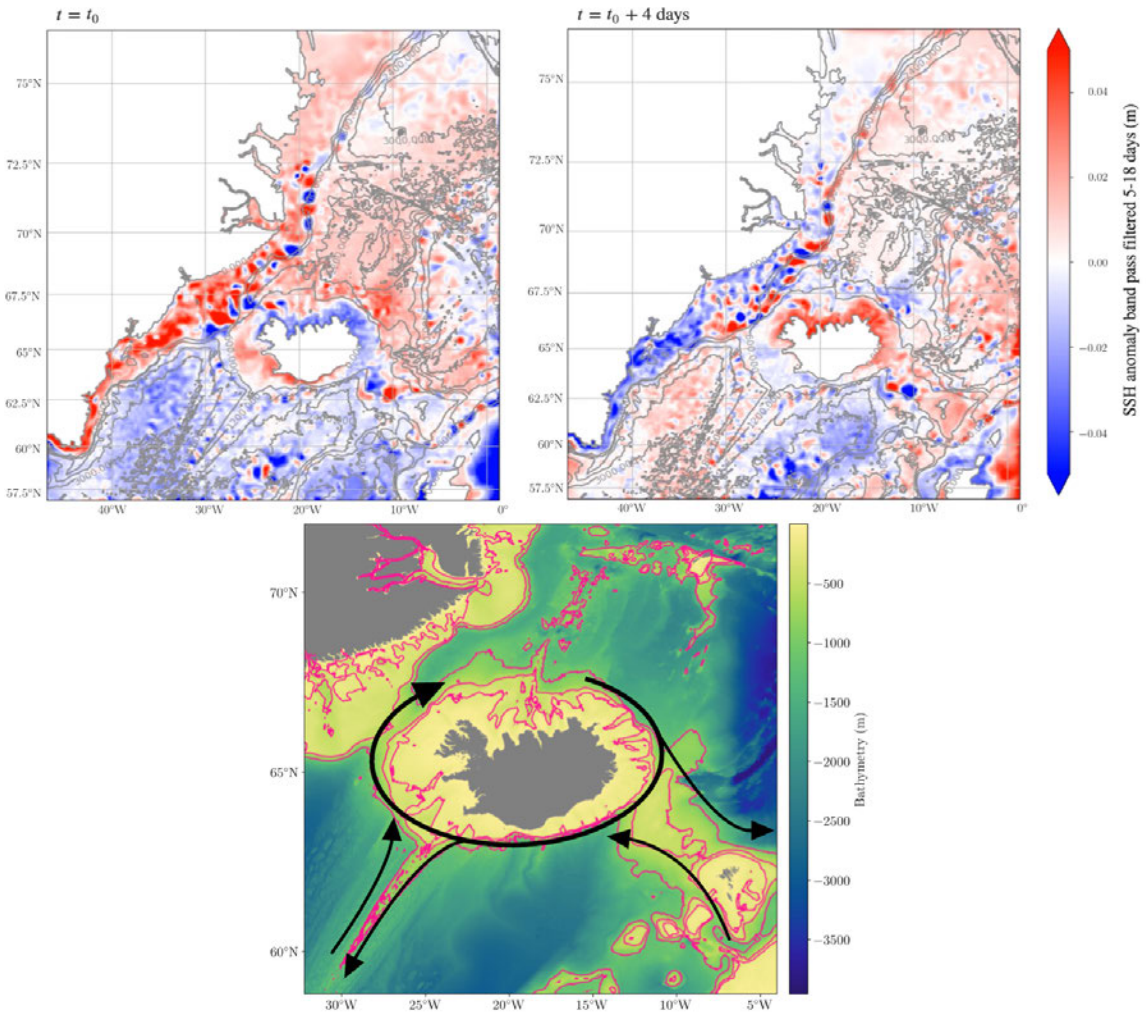


Figure 1: (*upper*) Snapshots of sea-surface height (SSH) anomalies along the coasts of Iceland and Greenland from simulations described in Gelderloos et al (2021) [7]. SSH anomalies have been band-pass filtered with a 5-day upper and 18 day lower frequency cutoff to isolate a subset of subinertial signals. (*lower*) Map showing the bathymetry surrounding Iceland [6]. Pink contours are the 250 m, 500 m and 1000 m isobaths. Black arrows schematically represent the direction of subinertial waves propagating around the Iceland continental slope and along the mid-Altantic (southwest to northeast) and Greenland-Scotland (southeast to northwest) ridges, as identified from the simulations in [7].

procedure with limited accuracy. More recently, Kaoullas and Johnson (2010) [13] demonstrated how spectral numerical schemes may be used to accurately and efficiently compute dispersion relations for CSWs over complex bathymetry without searching by reducing the system to a linear eigenvalue problem. Such methods have since been applied to coastal wave problems of increasing complexity [12, 18, 19, 20, for example] though simpler, analytically tractable problems can still be useful due to their easy interpretability. Here we

employ a combination of analytical and spectral methods.

The geometry considered in this study, reminiscent of Iceland, is an axisymmetric island surrounded by a continental shelf that is intersected by a ridge. We initially consider these structures separately, by analytically obtaining eigenmodes and dispersion relations for CSWs around an axisymmetric island (Section 3) and along an infinite ridge (Section 4). We then use insights from these simpler, analytically tractable geometries to guide the formulation of the combined geometry as a coupled 2D eigenvalue problem (Section 5). Eigenmodes of CSWs around an island abutted by a ridge are then sought numerically using spectral methods, and the influence of ridges on subinertial waves trapped around islands is assessed. Although we motivate this work with the specific setting of Iceland, the results will be widely applicable to many coastal oceans by bringing us closer to an understanding of how alongshore asymmetries influence resonant coastal waves.

2 Linear Continental Shelf Wave (CSW) Theory

Following Buchwald and Adams (1968) [3] and Huthnance (1975) [9] among others, we start with the linearized rotating shallow water equations on an f-plane

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \hat{\mathbf{k}} \times \mathbf{u} = -\nabla h' \tag{1}$$

$$D^2 \frac{\partial h'}{\partial t} + \nabla \cdot (h\mathbf{u}) = 0 \tag{2}$$

where $\mathbf{u}(x, y, t)$ is the horizontal velocity field, $h(x, y)$ is the mean fluid depth, and $h'(x, y, t)$ is the free surface displacement. Here the quantities $\mathbf{u}$, t, $\mathbf{x}$, h, and h' have been non-dimensionalized by the scales U, f^{-1}, L, H, and fUL/g, respectively, such that the non-dimensional parameter

$$D^2 \equiv \frac{f^2 L^2}{gH} \tag{3}$$

compares the continental slope breadth scale L (note this is the breadth of the sloping boundary of the continental shelf, not the continental shelf itself) to the Rossby radius of deformation for barotropic flow $\sqrt{gH}/f$.

In the presence of a coastal boundary and a continental shelf of non-uniform depth, the system defined by (1) and (2) can produce three types of waves that make up the barotropic subset of the more general class of "coastal trapped waves" (CTWs) [3, 9, 15]. The first are "edge waves", an infinite discrete set of high frequency ($\omega > f$) waves trapped near the coast by refraction. The dominant restoring force of these waves is gravity and they resemble rotating shallow water waves propagating in either direction along the coast. At very low wavenumbers the trapping mechanism breaks down and these waves become 'leaky' Poincairé waves. The second type are "continental shelf waves" (CSWs), an infinite discrete set of low frequency ($\omega < f$) waves originally formalized by Robinson (1964) [17] to describe observations off the Australian east coast. The dominant restoring force for these waves is

the conservation of potential vorticity, making them topographic Rossby waves that propagate along coastlines in a 'right-bound' sense (i.e. with the coastline or shallower water to the right). The third and final type is a singular non-dispersive Kelvin wave which is similarly right-bounded and is restored by the Earth's rotation. These three wave types are typical of 'trapped' waves generally and are analogous, for example, to the inertia-gravity waves, Rossby waves, and Kelvin wave sustained within the equatorial waveguide. Here variations in topography, rather than planetary β, provide the waveguide.

Since we are specifically targeting subinertial ($\omega < f$) oscillations around Iceland, we now scale (1) and (2) to isolate CSWs from the other two supported wave types. This may be achieved by assuming the non-dimensional parameter D^2 is negligible, i.e. that the horizontal lengthscale of the slope is much smaller than the Rossby radius of deformation [3, 9]. Through (2) we can see that this is equivalent to taking the 'rigid-lid' limit of the system where the free surface displacement h' has no time dependence. This approximation is justified for the Icelandic continental margin, among other continental margins, where the slope breadth is $\mathcal{O}(10$ km) and the Rossby deformation radius is $\mathcal{O}(100 - 1000$ km) and is perhaps even more appropriate in these high latitude regions where winter sea-ice cover impedes vertical free surface motion. Invoking this limit and cross differentiating (1) reduces our system to

$$\frac{\partial \zeta}{\partial t} + f\nabla \cdot \mathbf{u} = 0 \tag{4}$$

$$\nabla \cdot (h\mathbf{u}) = 0 \tag{5}$$

where $\zeta = (\partial_x v - \partial_y u)$ is the relative vorticity of the flow. Equation (5) suggests a streamfunction of the form

$$hu = -\frac{\partial \Psi}{\partial y}, \qquad hv = \frac{\partial \Psi}{\partial x} \tag{6}$$

which, when substituted into (4) returns the following topographic Rossby wave equation for barotropic CSWs,

$$\nabla \cdot \Big(\frac{1}{h}\nabla \frac{\partial \Psi}{\partial t}\Big) + f\hat{\mathbf{k}} \cdot \nabla\Psi \times \nabla\Big(\frac{1}{h}\Big) = 0. \tag{7}$$

In this work we'll be seeking propagating wave solutions to (7) of the form

$$\Psi(x, y, t) = \Re\{\Phi(x, y) \exp(-i\omega f t)\} \tag{8}$$

where $\Phi(x, y)$ is the spatial structure of the wave and ω (which has been non-dimensionalized by f) is its propagating frequency. Substituting (8) into (7) provides the expression

$$\frac{1}{h}\nabla^2\Phi + \nabla\Big(\frac{1}{h}\Big) \cdot \nabla\Phi + \frac{i}{\omega}\hat{\mathbf{k}} \cdot \Big(\nabla\Phi \times \nabla\Big(\frac{1}{h}\Big)\Big) = 0 \tag{9}$$

which produces CSWs when combined with the following boundary conditions

$$\begin{aligned} \Phi &= 0 \qquad \text{at the coast} \\ \Phi &\to 0 \qquad \text{at large distance.} \end{aligned} \tag{10}$$

The system defined by (9) and (10) is the foundation for all problems considered in this study.

3 An Axisymmetric Island

Consider a circular island of radius $r = R_i$ ($R_i < 1$) with a continental slope extending from a vertical wall at the coast to $r = 1$ (see Figure 2). Let the fluid depth over the slope be given by the polynomial expression

$$h(r) = \begin{cases} r^{2\alpha} & R_i \leq r \leq 1 \,(\text{slope}) \\ 1 & r > 1 \,(\text{far field}) \end{cases} \tag{11}$$

a geometry similar to that studied in [21] but translated from a rectilinear coast to a circular island. A simple concave downwards continental slope could, for example, be represented by a choice of $\alpha = 1$. The fluid depth is minimized at the coast at $h(R_i) = R_i^{2\alpha}$. Thus, for an island with a continental shelf depth h_0, we take $\alpha = \ln(h_0/H)/2\ln(R_i)$ where H is the far field depth. Note that we are only representing the continental shelf margin with this geometry, and that any extended shallow shelf region should be conceptualized as within the bounds of the island. This choice is made to avoid enforcing our assumption that $D^2 \ll 1$ (Section 2) to the wide, shallow inner shelf region where it less applicable [5].

Whilst the simple choice of $\alpha = 1$ is frequently utilized in this study, the bathymetry surrounding Iceland (Figure 1) is better captured by values of $h_0/H \sim 250/1000 = 0.25$ and R_i close to 1, since the breadth of the continental slope is small relative to the combined radius of the island and shelf region. These values suggest a larger α would be more appropriate when comparing analytical results to observations.

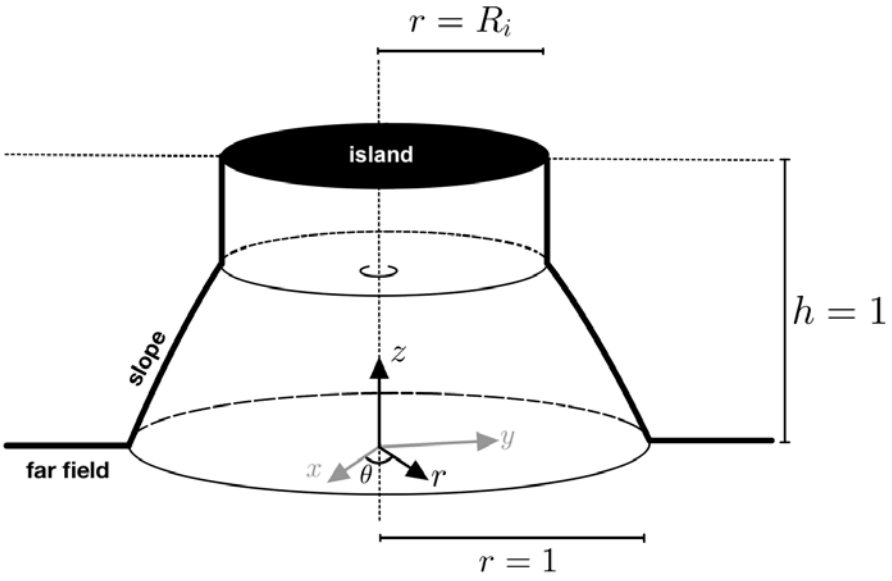


Figure 2: Geometry of the axisymmetric island problem. Dashed line represents the fluid surface and solid lines represent the bathymetry. The system is defined in polar coordinates (r, θ) with Cartisian horizontal coordinates (x, y) and the vertical coordinate (z), which collapses due to the barotropic nature of the problem, shown for reference.

Substituting this slope bathymetry for $h(r)$ in (9) (making appropriate adjustments for the shift to polar coordinates), and seeking solutions of the form

$$\Phi(r,\theta) = F_n(r)\exp(in\theta) \tag{12}$$

gives the Euler equation

$$r^2F_n^{''} + r(1-2\alpha)F_n' - (n^2 + 2n\alpha/\omega)F_n = 0, \qquad R_i \leq r \leq 1. \tag{13}$$

The general solution to (13) takes the form

$$F_n(r) = Ar^{\lambda_1} + Br^{\lambda_2}, \qquad \lambda_{1,2} = \alpha \pm \sqrt{\alpha^2 + n^2 + 2n\alpha/\omega} \tag{14}$$

where A and B are undetermined constants and $\lambda_{1,2}$ are roots of the auxiliary equation of the Euler equation (13).

Substituting the far field, constant bathymetry for $h(r)$ in (9) then provides the constraint that

$$\nabla^2\Phi = 0, \qquad r \geq 1 \tag{15}$$

which has general solution of the form

$$F_n(r) = Cr^{\lambda_3} + Dr^{\lambda_4}, \qquad \lambda_{3,4} = \pm n. \tag{16}$$

We're interested in wave solutions that decay away from the coast, see (10), and thus set $C = 0$. This implies $F_n(r) \propto r^{-n}$, $r \geq 1$ which may be rephrased as the constraint

$$F_n^{'}(r) + nF_n(r) = 0, \qquad r \geq 1. \tag{17}$$

Enforcing continuity of (17) at $r = 1$ provides a boundary condition on (13)

$$F_n^{'}(1) + nF_n(1) = 0, \tag{18}$$

and a second boundary condition is obtained by requiring that the solution goes to zero at the coast

$$F_n(R_i) = 0. \tag{19}$$

The boundary conditions (18) and (19), combined with the general solution form (14), provide the system

$$\begin{pmatrix} R_i^{\lambda_1} & R_i^{\lambda_2} \\ \lambda_1 + n & \lambda_2 + n \end{pmatrix} \begin{pmatrix} A \\ B \end{pmatrix} = 0. \tag{20}$$

This possesses non-trivial solutions if the determinant of the coefficient matrix of (20) (M_{island}, hereafter) vanishes, leading to the dispersion relation

$$\det(M_{\text{island}}) = R_i^{\lambda_1}(\lambda_2 + n) - R_i^{\lambda_2}(\lambda_1 + n) = 0, \tag{21}$$

with $\lambda_{1,2}$ determined as functions of ω by (14). For a given azimuthal wavenumber n, solving (21) numerically provides a discrete set of frequencies $\omega_{n,m}$, $m = 1, 2, \ldots$.

Equation (21) may be manipulated into a more tractable form that permits numerical determination of arbitrarily many roots $\omega_{n,m}$ for a given azimuthal wavenumber n. We consider only spatially oscillatory solutions where

$$\lambda_{1,2} = \alpha \pm i\gamma, \qquad \gamma = \sqrt{-\alpha^2 - n^2 - 2n\alpha/\omega} \in \mathbb{R} \tag{22}$$

such that

$$\omega = -\frac{2n\alpha}{\gamma^2 + n^2 + \alpha^2}. \tag{23}$$

Substituting for $\lambda_{1,2}$ in (21) and rearranging provides

$$\exp\left(2i\gamma \ln(R_i)\right) = \frac{\alpha + n + i\gamma}{\alpha + n - i\gamma}. \tag{24}$$

Noting that $\arg z = \phi$ where $z = r\exp(i\phi)$ allows us to express the above as

$$\gamma \ln(R_i) = \arg(\alpha + n + i\gamma), \tag{25}$$

which, given $\arg(x + iy) = \arctan(y/x)$, simplifies to

$$\tan\tilde{\gamma} = \frac{\tilde{\gamma}}{(\alpha + n)\ln(R_i)} \tag{26}$$

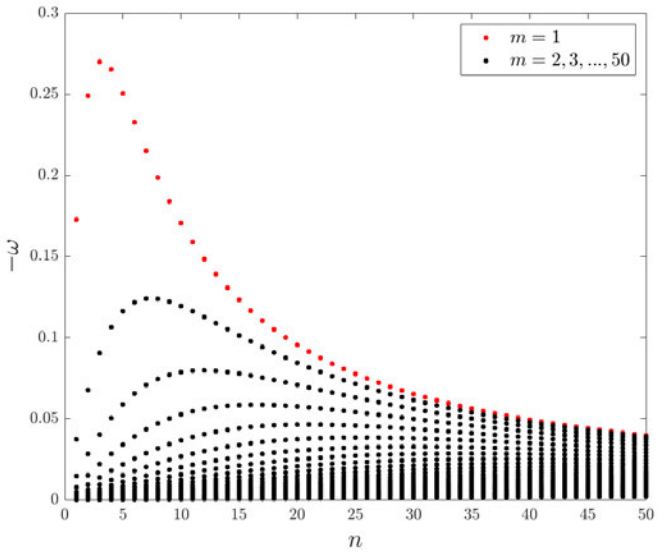


Figure 3: Dispersion relation for continental shelf waves around an axisymmetric island defined by (11) with $R_i = 0.5$ and $\alpha = 1$. Note n and m refer to the azimuthal and radial wavenumbers, respectively, and ω is non-dimensionalized by the Coriolis frequency f.

where $\tilde{\gamma} = \gamma \ln(R_i)$. Numerical determination of arbitrarily many roots of (26) is straightforward since a straight line intersects a tangent function exactly once within each interval $\tilde{\gamma} \in (\pi/2 + j, 3\pi/2 + j)$, $j = 0, 1, 2, \ldots$. The roots of (26), $\tilde{\gamma}_{n,m}$, are then translated into frequencies $\omega_{n,m}$ through (23). Note roots of (26) are sought only where $\tilde{\gamma} > 0$ in order to satisfy (22). Similar methods of finding arbitrarily many roots to CSW equations are employed by [13].

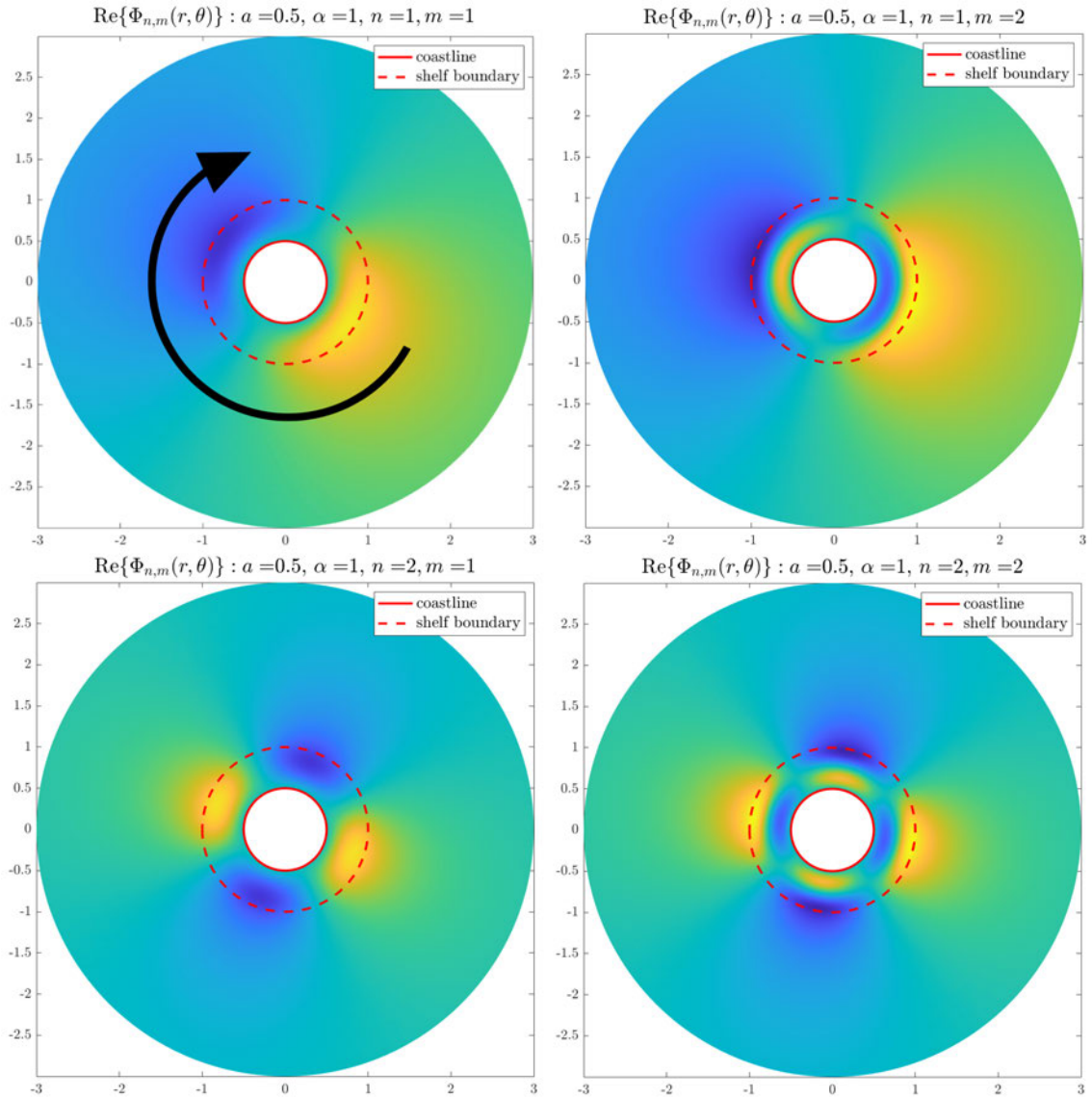


Figure 4: Spatial structure of continental shelf waves sustained around an axisymmetric island defined by (11) with $R_i = 0.5$ and $\alpha = 1$. Waveforms associated with $n = 1, 2$ and $m = 1, 2$ shown. Solid and dashed red lines represent the coastline ($r = R_i$) and slope boundary ($r = 1$), respectively. Black arrow in the first panel indicates direction of wave propagation.

The resulting $\omega_{n,m}$ are negative for $k > 0$ and positive for $k < 0$, corresponding to clockwise or 'right-bound' propagation in this problem geometry. These $\omega_{n,m}$ form a dispersion relation (Figure 3) with a structure typical of Rossby waves but discretized in both azimuthal (n) and radial (m) wavenumbers. For any given azimuthal wavenumber n, the greatest frequencies are associated with the $m = 1$ wave, with frequencies decreasing in magnitude monotonically with increasing m. All sustained frequencies are subinertial (ω has been non-dimensionalized by f), as expected. Decreasing the shelf depth h_0 via α pushes the maximum magnitude frequency towards the inertial limit whilst increasing the shelf steepness via α (while keeping h_0 fixed) changes the shape of the dispersion relation such that the maximum magnitude frequency is associated with a larger n and more modes lie close to the maximum magnitude frequency. Note we find the minimum shelf depth rather than the slope steepness determines the maximum supported frequency, in contrast to [21].

Once the frequencies $\omega_{n,m}$ are generated, we may determine their associated waveforms $\Phi(r, \theta)$ via (12), (14), and (20) (up to a free parameter). Figure 4 shows the spatial structure $\Phi(r, \theta)$ of propagating waves with $n = 1, 2$ and $m = 1, 2$. Finally, these results were confirmed numerically using Dedalus v3 [4] spectral solvers by constructing a 1D eigenvalue problem out of (13), (18), and (19) and discretizing the r coordinate in a Chebyshev basis.

4 An Infinite Ridge

Now consider a rectilinear ridge extending to infinity along the x-axis with profile a function of y alone (see Figure 5). Let the ridge be symmetric about $y = 0$ with half-width W and the fluid depth described by

$$h(y) = \begin{cases} \exp\big(2b(|y| - W)\big) & |y| \leq W \text{ (ridge)} \\ 1 & |y| > W \text{ (far field).} \end{cases} \tag{27}$$

The fluid depth is minimized at the peak of the ridge $h(0) = \exp(-2bW)$. Thus, for a ridge with minimum fluid depth h_0 we take $b = -\ln(h_0/H)/2W$ where H is the far field depth. When considering the Greenland-Scotland ridge abutting Iceland, for example, we have $h_0/H \sim 500/1000 = 0.5$, and a ridge halfwidth $W \sim 100$km implying a very small b. However, for simplicity we will assume a value of $b = 0.5$ in this section. Note that the extension to non-symmetric profiles is immediate with the sole requirement that $b^+W^+ = b^-W^-$ where $b^\pm$ and $W^\pm$ are the values of b and W in $y > 0$ and $y < 0$.

Substituting this ridge bathymetry for $h(y)$ in (9), and seeking solutions of the form

$$\Phi(x, y) = G_k(y) \exp(ikx) \tag{28}$$

bring us to

$$\begin{aligned} G_k'' - 2b\, G_k' - \Big(k^2 - \frac{2bk}{\omega}\Big) G_k = 0, \qquad & 0 < y \leq W \\ G_k'' + 2b\, G_k' - \Big(k^2 + \frac{2bk}{\omega}\Big) G_k = 0, \qquad & -W \leq y < 0 \end{aligned} \tag{29}$$

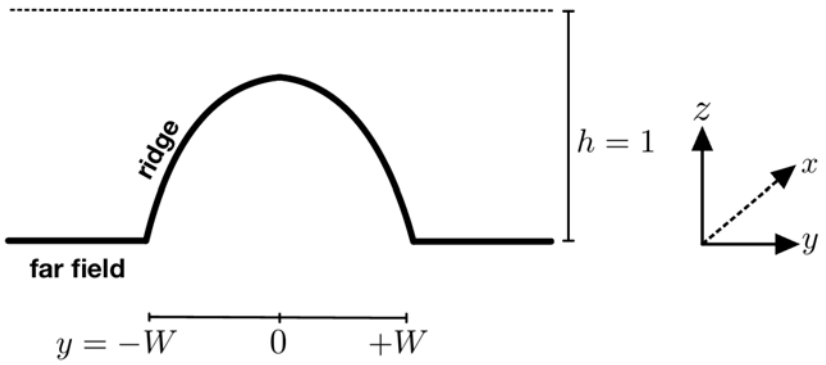


Figure 5: Geometry of the infinite ridge problem. Dashed line represents the fluid surface and solid lines represent the bathymetry. The system is defined in Cartisian horizontal coordinates (x, y) with the vertical coordinate (z), which collapses due to the barotropic nature of the problem, shown for reference.

The general solutions of which take the form

$$\begin{aligned} G_k(y) &= P\exp(\lambda_1 y) + Q\exp(\lambda_2 y), \qquad 0 < y \leq W \\ G_k(y) &= R\exp(\lambda_3 y) + S\exp(\lambda_4 y), \qquad -W \leq y < 0 \end{aligned} \tag{30}$$

where P, Q, R and S are undetermined constants and $\lambda_{1,2}$ and $\lambda_{3,4}$ are roots of the auxiliary equations of (29),

$$\lambda_{1,2} = b \pm \sqrt{b^2 + k^2 - 2bk/\omega}, \qquad \lambda_{3,4} = -b \pm \sqrt{b^2 + k^2 + 2bk/\omega}. \tag{31}$$

Similarly to the axisymmetric island case, taking the decaying solution to the far field system requires $G_k(y) \propto e^{-|k||y|}, |y| \geq W$ which may be rephrased as the constraint

$$\begin{aligned} G_k^{'} + |k|G_k &= 0, \qquad y \geq W \\ G_k^{'} - |k|G_k &= 0, \qquad y \leq -W. \end{aligned} \tag{32}$$

Enforcing continuity at $|y| = W$ provides the following boundary conditions,

$$G_k^{'}(W) + |k|G_k(W) = 0 \tag{33}$$

$$G_k^{'}(-W) - |k|G_k(-W) = 0. \tag{34}$$

Two additional constraints arise from enforcing continuity of G_k and $G_k^{'}$ across $y = 0$,

$$G_k(0^+) = G_k(0^-) \tag{35}$$

$$G_k^{'}(0^+) = G_k^{'}(0^-). \tag{36}$$

Together, the boundary conditions (33)-(36) along with the general solution form (30) provide the system of equations

$$\begin{pmatrix} e^{\lambda_1 W}(\lambda_1+|k|) & e^{\lambda_2 W}(\lambda_2+|k|) & 0 & 0 \\ 0 & 0 & e^{-\lambda_3 W}(\lambda_3-|k|) & e^{-\lambda_4 W}(\lambda_4-|k|) \\ 1 & 1 & -1 & -1 \\ \lambda_1 & \lambda_2 & -\lambda_3 & -\lambda_4 \end{pmatrix} \begin{pmatrix} P \\ Q \\ R \\ S \end{pmatrix} = 0 \quad (37)$$

which has non-trivial solutions when the determinant of the coefficient matrix of (37) (M_{ridge}, hereafter) is zero,

$$\det(M_{\text{ridge}}) = 0. \quad (38)$$

As in the axisymmetric island problem, we manipulate (38) into a form that permits easy numerical determination of arbitrarily many roots. Once again, we are only interested in spatially oscillatory solutions, though in this case solutions may be oscillatory in either the $y > 0$ domain

$$\lambda_{1,2} = b \pm i\sqrt{-b^2 - k^2 + 2bk/\omega}, \;\; \lambda_{3,4} = -b \pm \sqrt{b^2 + k^2 + 2bk/\omega} \quad (39)$$

$$\text{for } 0 < \omega < \frac{2bk}{b^2+k^2}, \text{ and } k > 0$$

or the $y < 0$ domain

$$\lambda_{1,2} = b \pm \sqrt{b^2 + k^2 - 2bk/\omega}, \;\; \lambda_{3,4} = -b \pm i\sqrt{-b^2 - k^2 - 2bk/\omega} \quad (40)$$

$$\text{for } -\frac{2bk}{b^2+k^2} < \omega < 0, \text{ and } k > 0.$$

Substituting (39) into (38) and manipulating into a tangent function provides

$$\tan\tilde{\gamma} = \quad (41)$$

$$\frac{\tilde{\gamma}}{W} \frac{e^{W\zeta}(-b-|k|-\zeta)(-b+|k|+\zeta) - e^{-W\zeta}(-b-|k|+\zeta)(-b+|k|-\zeta)}{e^{Wy}(-b-|k|-\zeta)(\frac{\tilde{\gamma}^2}{W^2}+(b+|k|)(2b-\zeta)) - e^{-W\zeta}(-b-|k|+\zeta)(\frac{\tilde{\gamma}^2}{W^2}+(b+|k|)(2b+\zeta))}$$

where

$$\tilde{\gamma} = W\gamma = W\sqrt{-b^2-k^2+2bk/\omega}, \qquad \zeta = \sqrt{b^2+k^2+2bk/\omega} = \sqrt{2b^2+2k^2+\tilde{\gamma}^2/W^2}.$$

The roots of (41), $\tilde{\gamma}_m$, for a given along slope wavenumber k and bathymetry parameters b and W provide the frequencies $\omega_{k,m}$ of waves propagating along the $y > 0$ side of the ridge with cross slope wavenumbers $m = 1, 2, 3, \ldots$

$$\omega^+_{k,m} = \frac{2bk}{\tilde{\gamma}_m^2/W^2 + b^2 + k^2}. \quad (42)$$

The largest magnitude $\omega_{k,m}$ for a given k is associated with the $m = 1$ wave, the next largest with the $m = 2$ wave, and so on. Unlike in the axisymmetric island case, the right hand side of (41) is not a straight line but takes the form of $1/\tilde{\gamma}$. As such, the numerical root finding procedure should seek two roots in the interval $\tilde{\gamma} \in (\pi/2+j, 3\pi/2+j)$, $j = 0, 1, 2, \ldots$ within which the denominator of (41) goes to zero. This is on account of the discontinuity and associated sign change in the left hand side of (41) permitting an intersection with $\tan\tilde{\gamma}$ in both the positive and negative lobes of $\tan\tilde{\gamma}$.

The symmetries of the problem are such that the oscillating solutions in the $y < 0$ domain (obtained by substituting (40) into (38) and numerically determining roots to a tangent expression, as above) are found to be

$$\omega^{-}_{k,m} = -\omega^{+}_{k,m}. \tag{43}$$

This indicates that perturbations at a given frequency $\omega_{k,m}$ will generate waves propagating in the positive x direction on the $y > 0$ side of the ridge and in the negative x direction on the $y < 0$ side of the ridge, in line with our expectation of 'right-bound' wave propagation. Note that if we take $k < 0$ the sign of the resulting roots flip such that right bounded propagation is retained. The numerically determined $\omega_{k,m}$ form a dispersion relation (Figure 6) with a typical Rossby wave structure. The infinite nature of the ridge considered permits a continuous dispersion relation with respect to k. Once again, all sustained frequencies are subinertial (ω has been non-dimensionalized by f) and decreasing the water depth over the peak of the ridge h_0 via b acts to increase the magnitude of sustained frequencies towards the inertial limit.

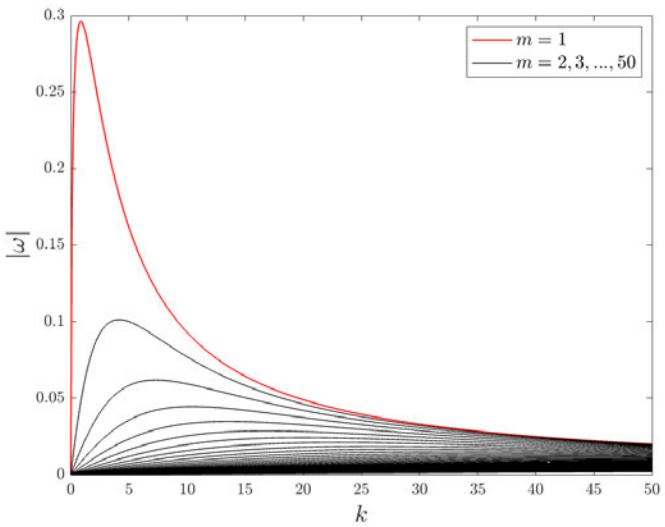


Figure 6: Dispersion relation for topographic Rossby waves along an infinite ridge defined by (27) with $W = 1$ and $b = 0.5$. Note k and m refer to the along slope and cross slope wavenumbers, respectively, and ω is non-dimensionalized by the Coriolis frequency f.

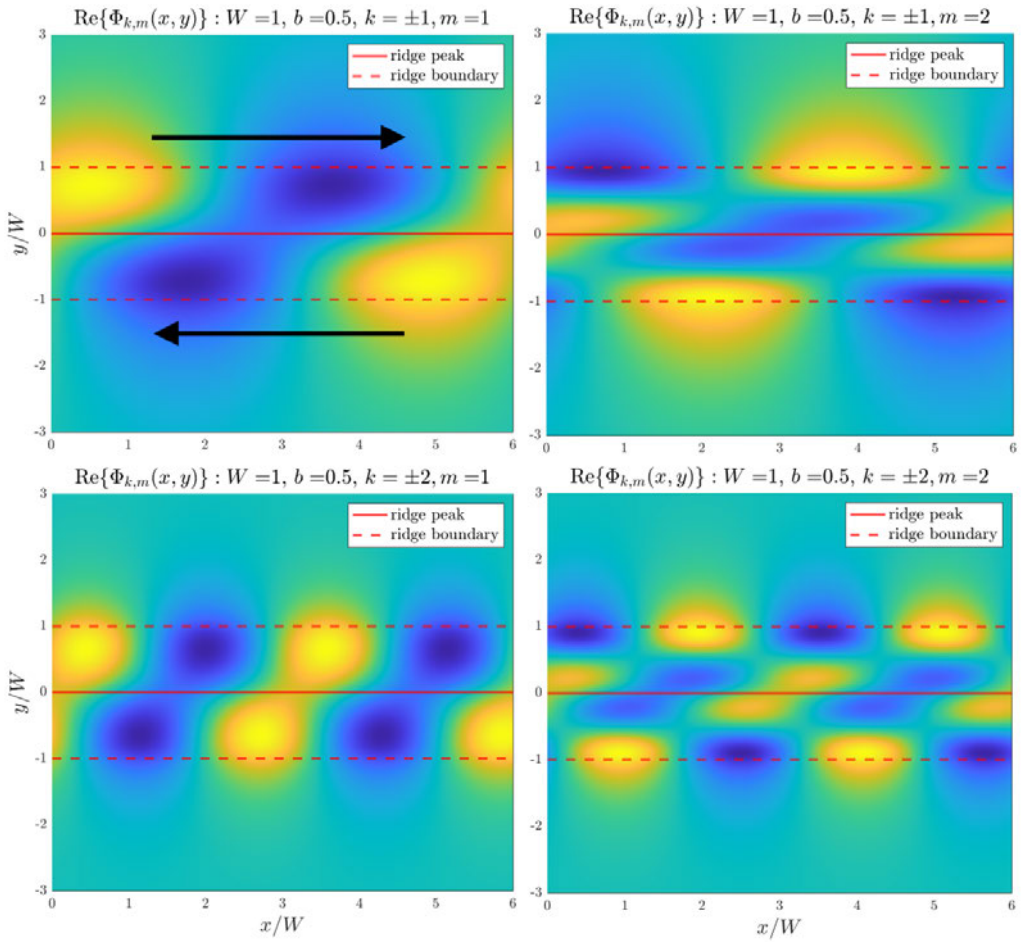


Figure 7: Spatial structure of topographic Rossby waves sustained along an infinite ridge defined by (27) with $W = 1$ and $b = 0.5$. Waveforms associated with $k = 1, 2$ and $m = 1, 2$ shown. Waveforms shown are a linear combination of modes associated with two roots $\omega^{\pm}_{k,m}$ which have mirrored structures across $y = 0$ and propagate in opposing directions as right bound waves.Solid and dashed red lines represent the peak of the ridge ($y = 0$) and slope boundary ($y = \pm W$), respectively. Black arrows in the first panel indicate direction of wave propagation.

Once the frequencies $\omega_{k,m}$ are generated, we may determine their associated waveforms $\Phi(x, y)$ via (28), (30), and (37) (up to a free parameter). Figure 7 shows the spatial structure $\Phi(x, y)$ of propagating waves with $k = 1, 2$ and $m = 1, 2$. Solutions associated with $\omega^{+}_{k,m}$ and $\omega^{-}_{k,m}$ are linearly combined to show symmetrical pairs of waves propagating with the same frequency along either side of the ridge. Again, these results were replicated using Dedalus v3 [4] spectral solvers by constructing a 1D eigenvalue problem out of (29), (33), and (34) and discretizing the y coordinate in a Chebyshev basis.

5 An Island Intersected by a Ridge

The two problems considered in Section 3 and Section 4 can then be combined to probe how intersecting marine ridges affect the set of CSWs that can propagate around islands. The introduction of an abutting ridge breaks the radial symmetry of the island problem, resulting in a non-separable 2D eigenvalue problem unsuited to the analytical approach adopted for the simpler geometries above. Therefore, in this section we utilize spectral methods (Dedalus v3 tools [4]) to numerically approximate the eigenvalues (frequencies, ω) and associated eigenmodes (waveforms, Φ) of propagating CSWs around a 2D combined island ridge geometry.

We formulate the combined problem in polar coordinates, with the fluid depth described by

$$h(r,\theta) = \min\{h_{\text{island}}, h_{\text{ridge}}, 1\}, \qquad R_i \leq r \leq R_o \tag{44}$$

where h_{island} is given by (11), h_{ridge} is a translation of (27) to polar coordinates with a constant angle halfwidth W and peak centered on $\theta = \pi$, and R_i and R_o are the inner and outer domain radii, respectively. In order to simplify the outer boundary condition, we make

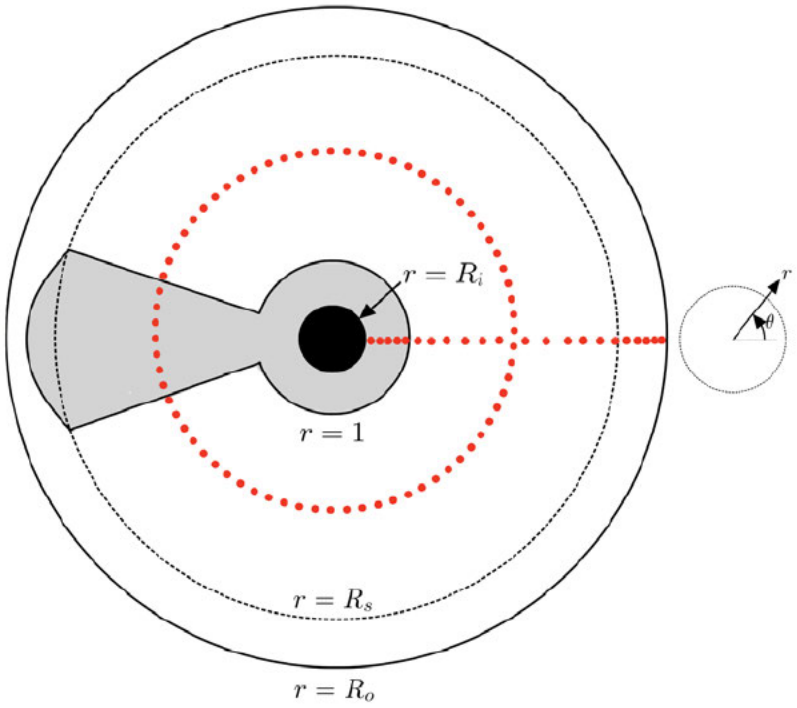


Figure 8: Geometry of the combined island and ridge problem. The central black region is the island, the grey region is the island continental slope and ridge as described by (44), and the white region is the (flat) far field. Key r values are labeled and the red dots schematically represent (not to scale) the choice of Chebyshev and Fourier bases to represent the r and θ dimensions, respectively.

the ridge long but finite, enforcing that it decays exponentially after some radius $R_s \gg 1$ according to a steepness parameter s

$$h_{\text{ridge}}(r,\theta) = \exp(2b(|\pi-\theta|-W))\exp(s(r-R_s)), \qquad R_s \leq r \leq R_o \tag{45}$$

where s is chosen such that $h(R_o,\theta) = 1\,\forall\theta$. This geometry is sketched in Figure 8.

Within the Dedalus v3 framework r and θ are discretized with 64 Chebyshev points and 64 Complex Fourier points, respectively (schematically represented in Figure 8). A 2D eigenvalue problem is then constructed of (9) with a Dirichlet boundary condition setting $\Phi = 0$ at R_i and a Dirichlet to Neumann boundary condition enforcing the decaying solution of $\nabla^2\Phi = 0$ at R_o. Boundary conditions are imposed using the generalized tau method [4, 16]. Due to poor scaling of the 2D eigenvalue problem (required storage and computation time scales like $(N_\theta \cdot N_r)^3$ where N_a is the number of points in the a dimension) all experiments presented here are computed on a 64×64 grid, though greater resolution may be attained at some expense. We set $\alpha = 1$ and $R_i = 0.5$ in all experiments, giving a continental shelf height of $1 - R_i^{2\alpha} = 0.75$. The ridge bathymetry parameter b is set to

$$b = -\ln(1 - 0.75\text{frac})/2W \tag{46}$$

where frac is the maximum height of the ridge as a fraction of the shelf height. Values of $\text{frac} = 0.3, 0.5, 0.7$ are tested and 8 repetitions are run for each value of frac. R_o, R_s, W and s are varied between repetitions. Parameter choices are summarized in Table 1. Note that the island in this combined configuration is identical to the island considered in Section 3 but that the ridge has been modified from Section 4 in three significant ways (i) the ridge is now finite in extent, (ii) the ridge halfwidth now increases with r, and (iii) the ridge now sits in a periodic domain.

α	R_i	frac	R_o, s	R_s	W
1	0.5	$0.3, 0.5, 0.7$	$\{17, 0.2\}, \{20, 0.1\}$	$12, 14$	$\pi/5, \pi/6$

Table 1: Summary of parameters tested in the combined island and ridge problem.

Informed by the results of Sections 3 and 4, we expect the island in this problem to support trapped CSWs at a discrete set of frequencies, whilst the ridge, if made long enough to be well approximated by the infinite problem, should support CSWs at a continuum of frequencies up to a maximum ω_{ridge}. A predicted value for ω_{ridge} can be estimated from the dispersion relation of an infinite ridge given its height (Section 4). We anticipate the eigenvalues of the 2D combined problem will produce a dispersion relation similar to Figure 3 but that eigenmodes associated with frequencies smaller than ω_{ridge} will show waves propagating along the abutting ridge, whilst eigenmodes associated with frequencies larger than ω_{ridge} will be contained to the island slope (Figure 9).

The full solution to the coupled 2D system comprises $\mathcal{O}(N_r \times N_\theta)$ eigenvalues (here 4224, with additional terms arising from the tau method, [4, 16]) and associated eigenmodes. Sorting resolved from unresolved eigenvalues and characterizing their associated eigenmodes, including identifying the azimuthal (n) and radial (m) wavenumbers of waves around the island and determining whether a wave is present along the ridge, is a non-trivial exercise. We take a targeted approach and use a sparse solver inbuilt in Dedalus v3 [4] to seek only 40 eigenvalues in the neighbourhood of $\omega = -0.3$, a value close to the expected maximum frequency around the island. The eigenmodes are then sorted by $|\omega|$ and waveforms are plotted and visually inspected to identify island wavenumbers and determine whether a wave is present on the ridge (see Figure 10 for examples). Note that waves propagating along the ridge need not match the wavelengths of modes trapped around the island, which would result in a single wave propagating around the whole combined structure, instead we generally see island modes acting as a wavemaker for ridge modes of the same frequency but different wavelengths. Further it may be noted that the problem does not produce eigenmodes that exist solely on the ridge. The ridge acts to modify the eigenmodes of the symmetric island problem but does not introduce new eigenmodes with frequencies between the frequencies sustained around the island.

The results of the manual classification are presented in Figure 11. Generally island modes up to $\{n, m\} = \{15, 1\}$ or approximately $\omega_{n,m} = -0.11$ were easily characterized

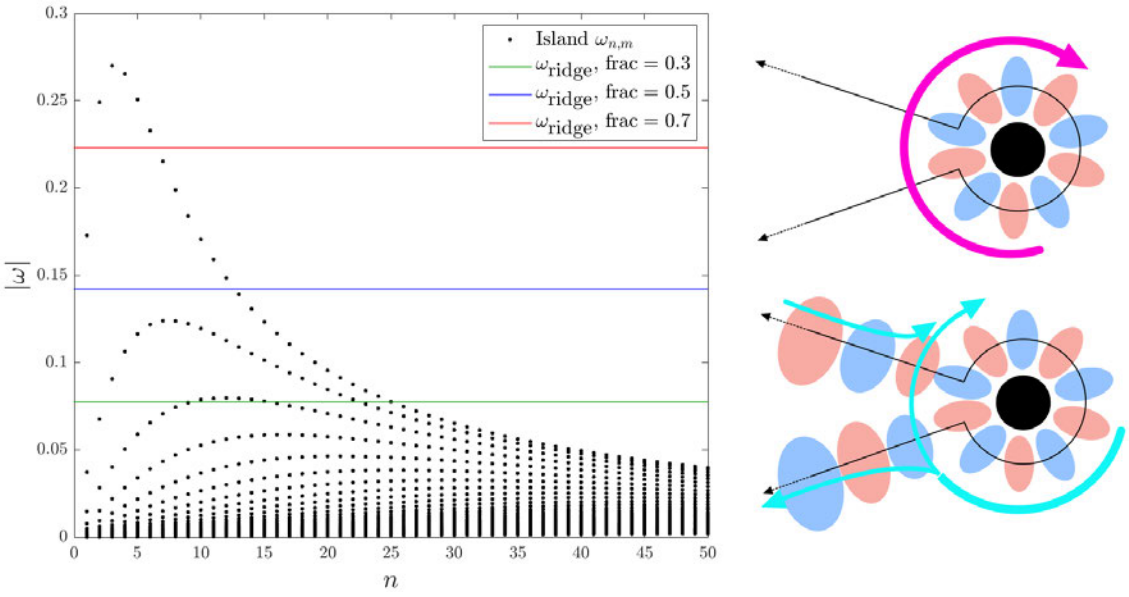


Figure 9: Predicted behaviour of the combined island and ridge problem. (*left*) Dispersion relation for continental shelf waves around an axisymmetric island (as in Figure 3) overlaid with the maximum (in terms of magnitude) frequencies predicted to propagate along infinite ridges of height 0.3 (green), 0.5 (blue), and 0.7 (red) times the island continental shelf height. (*right*) Anticipated behaviour of island wave modes with frequency larger (*top*) and smaller (*bottom*) in magnitude than the maximum frequency sustained by the ridge.

by eye (e.g. Figure 10), though removing the mean $\Phi(r)$ structure, which often contained $n = 0$ signals that do not decay with r, aided classification. In some cases it was difficult to discern whether signals over the ridge were associated with a long wave CSW traveling along the ridge or a low mode structure associated primarily with the terminus of the ridge, these ambiguous cases are noted in Figure 11. Some eigenmodes contained mixed island waveforms where their associated frequencies were similar, in particular $\{n, m\} = \{5, 1\}$ and $\{2, 1\}$ generally appeared in the same eigenmode. Figure 11 supports the hypothesis that CSWs around an island abutted by a ridge change their behavior abruptly across a threshold frequency, determined by the maximum frequency of propagating ridge CSWs. Crucially, almost all eigenmodes associated with frequencies below ω_{ridge} clearly displayed waves along the ridge, suggesting the chosen ridge geometry supports a sufficiently continus spectra of waves to be well approximated by the simpler infinite ridge case. However,

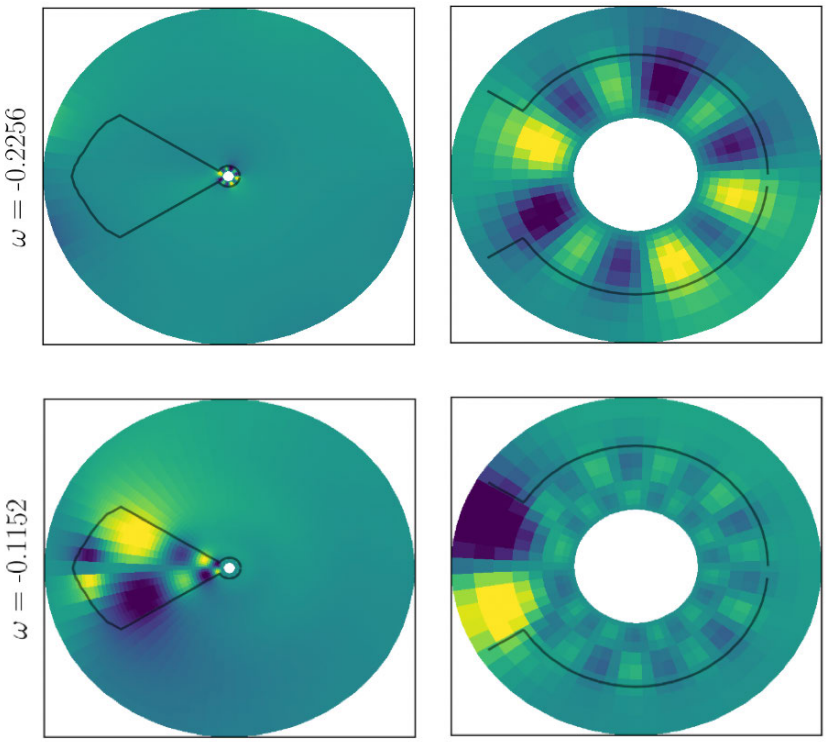


Figure 10: Example eigenmodes of the numerical 2D system categorized as (*upper*) island only modes ($\omega_{n=6,m=1} = -0.2256$ shown), and (*lower*) island and ridge modes ($\omega_{n=10,m=2} = -0.1152$ shown). Panels show $\Phi(r, \theta)$ over the full domain (*left*), and a zoomed subset of the domain to assist identification of island wave numbers (*center*). Black contour shows the boundary of the region with varying bathymetry. Examples taken from the frac = 0.5 experiments. The mean $\Phi(r)$ (averaged over θ) has been removed to filter out the $n = 0$ mode, which is not constrained to decay with r in the numerical problem.

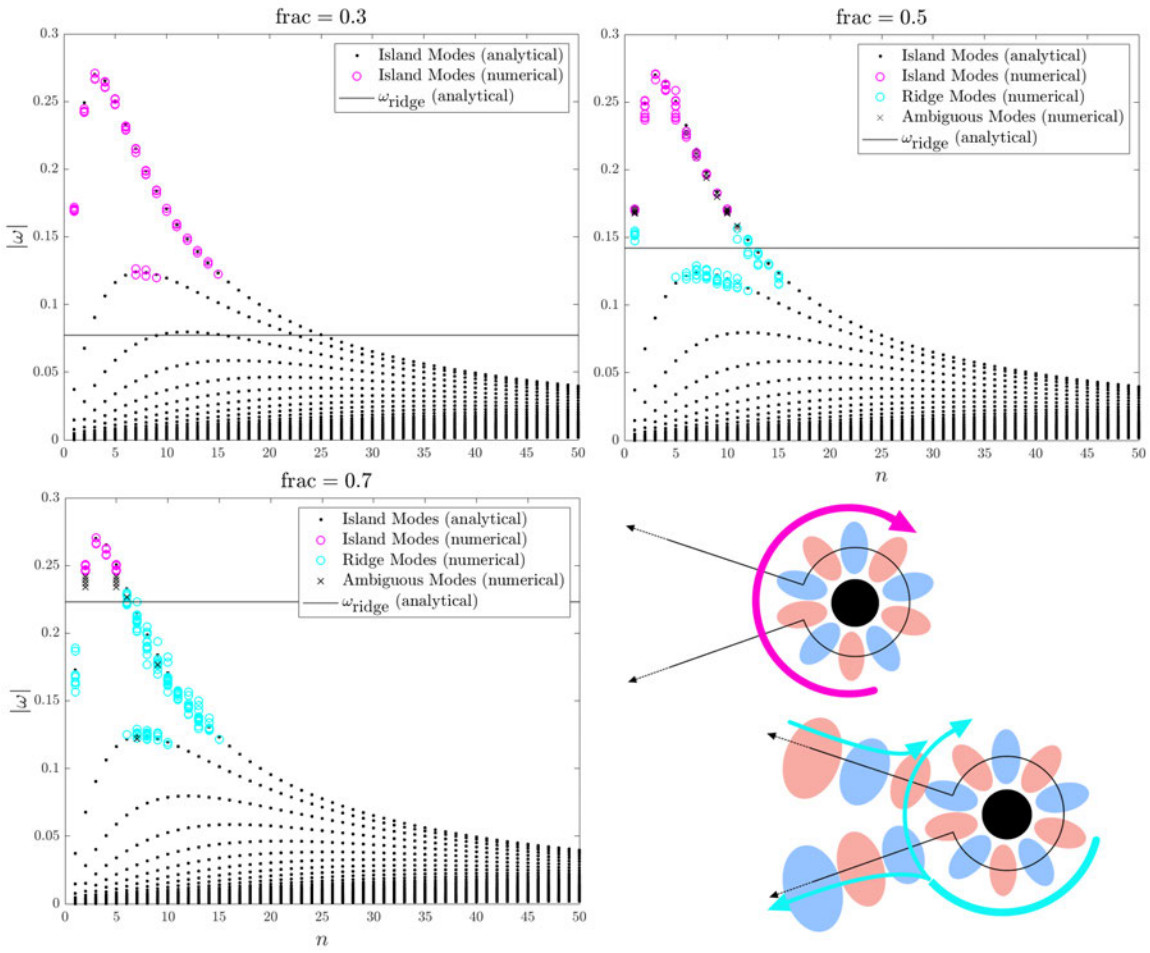


Figure 11: Manual categorization of eigenmodes into island only (magenta circles), island and ridge (cyan circles) and ambiguous (black crosses) modes. Computed eigenmodes associated with frequencies lower (in magnitude) than approximately $\omega = 0.1$ are not shown due to increasing difficulty of identifying n and m values as they become poorly resolved. Illustrations on the lower right visualise the difference between waves that do (magenta) and do not (cyan) propagate along the ridge.

Figure 11 suggests the analytical ω_{ridge} may systematically underestimate the magnitude of this threshold as ridge waves were associated with frequencies greater than ω_{ridge} in both the frac $= 0.5$ and frac $= 0.7$ cases.

As a secondary, more quantitative, indicator of whether a given island mode is associated with a wave along the ridge, we compute the kinetic energy of the wave field

$$KE = \nabla\Phi \cdot \nabla\Phi^* \tag{47}$$

where $\{\}^*$ indicates the complex conjugate, and compare the total KE on the ridge to the total KE off the ridge between $r = 1$ and $r = R_s$. The percentage of KE concen-

trated over the ridge increases as frequency decreases (Figure 12) and, by extension, as island and ridge wavenumbers increase. Modes with frequency $\omega < \omega_{\text{ridge}}$ have, on average, 82%(±12%) of KE between $r = 1$ and $r = R_s$ on the ridge, while for modes with frequency $\omega > \omega_{\text{ridge}}$ have 47%(±16%) of KE on the ridge. The transition across ω_{ridge} is not abrupt, possibly associated with ambiguous modes near the transition and the potential systematic underestimation of ω_{ridge}. However, the concentration of kinetic energy on the ridge at lower frequencies generally supports their classification as ridge modes in Figure 11.

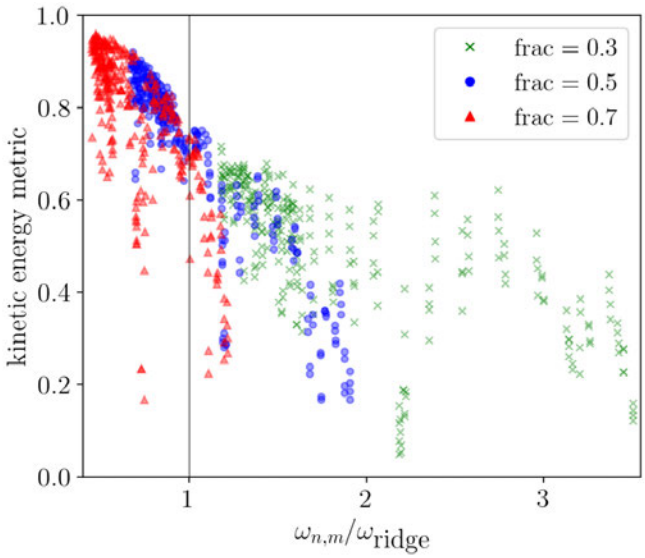


Figure 12: Kinetic energy (47) over the ridge between $r = 1$ and $r = R_s$ as a fraction of the total kinetic energy in that radial band for the first 40 eigenvalues of our system. Frequencies are expressed relative to the analytically predicted maximum frequency permitted on each ridge.

As a final note on the effect of bathymetric parameters, the bathymetry around Iceland (Figure 1) suggests a normalized shelf depth of ~ 0.25, a normalized ridge depth of ~ 0.5 (i.e. frac ~ 0.5), with R_i close to 1. We noted in Section 3 that increasing α whilst holding h_0 constant (an effect of increasing R_i) acts to pull the frequency of island modes upwards towards the maximum magnitude frequency sustained by the island. Thus, we would expect more island modes above ω_{ridge} in a system with more Iceland-like parameter choices (Figure 13).

6 Discussion

Based on the eigenvalue problems considered, we find that a symmetric island can sustain only a discrete set of trapped barotropic CSWs with alongshore wavenumbers that fit precisely into the island circumference, thus exhibiting a resonance (Section 3). By contrast, infinite ridges may sustain CSWs at a continuum of frequencies below some maximum

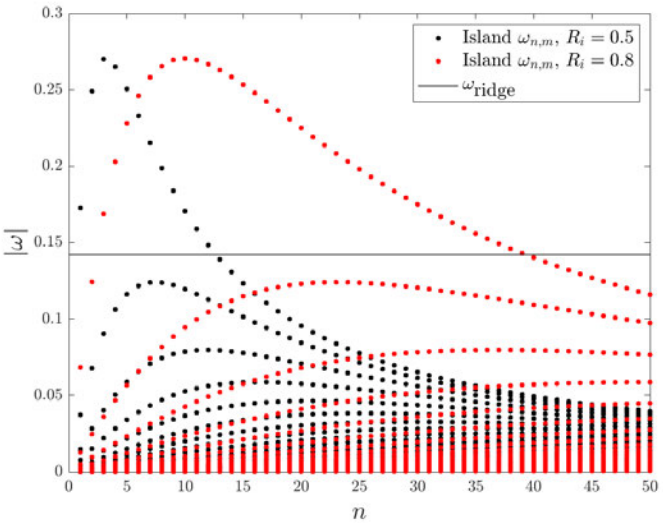


Figure 13: Effect of increasing R_i whilst holding the shelf depth constant, thus making the island shelf more narrow and steep, on the island dispersion relation. Here frac $= 0.5$, the minimum shelf depth is set to $h_0/H = 0.25$, and α is determined as a function of h_0/H and R_i.

value (Section 4). When the continental slope of an island is intersected by a marine ridge with a maximum height less than the island's shelf height, island modes that oscillate at frequencies below the maximum frequency sustained along the ridge act as a wavemaker, generating waves that propagate along the ridge (Section 5). This anticipated behavior is well supported by numerical results, though further investigation is required to understand why waves are supported along the ridge at slightly higher frequencies than the analytical problem suggests possible.

If the intersecting ridge were truly infinite, we would expect these ridge waves to extract all available energy from their associated island modes and propagate said energy into the far field, such that lower frequency modes drop out of the island wavefield. In this case, only a handful of island modes above a threshold frequency would be truly trapped, potentially rendering them easier to identify in observations and general circulation models. In our numerical eigenvalue problem (Section 5) we are confined to studying a long but finite intersecting ridge. Due to its finite extent, along ridge waves propagate around the ridge and back towards the island, returning energy to associated island modes such that these modes do not fall out of the eigenvalue problem. An obvious next step would be reformulate our eigenvalue problem as a time dependent, forced (e.g. by broadband Ekman pumping) model with viscous dissipation, and assess whether or not along ridge waves interrupt the resonance of and extract energy from low frequency island trapped modes, reducing their prominence in frequency spectra.

At this stage, it remains unclear whether the intuition gained from considering an in-

finite ridge translates well to the real geometry of Iceland, where abutting marine ridges are finite. The current interpretation positions the trapped island modes as wavemakers for along ridge waves that may divert energy away from the island. It is possible that the real system behaves more like a convolution of two island-like continental margins of different heights, each with discrete dispersion relations, than an infinite ridge with a continuous spectra abutting a symmetric island with a discrete spectra. In this case we might expect modes present on the 'ridges' to be constrained to fit precisely into the perimeter of the combined island-ridge structure, whilst that does not appear to be a constraint in the geometry considered here. Exploring this possibility by considering shorter ridge lengths within the current framework may be illustrative, and eventually spectra from observations and general circulation models may be compared to end member results of the two geometries.

It should be noted the CSW equations used throughout this study approximate the coastal oceans as inviscid and barotropic (Section 2). Including viscous dissipation may be feasible in the numerical problem [18], however including stratification would inflate the system to a 3D eigenvalue problem, a considerable increase in computational complexity. Both the inviscid and barotropic assumptions might be expected to fail in certain circumstances such as when strong currents pass sharp capes or when the coastal flow is strongly stratified. However, for small amplitude CSWs in quiescent flow along smoothly varying coastlines, viscous separation is negligible. Further, most CTW disturbance energy is concentrated in the modes with the least vertical structure, which are well described by the purely barotropic constant density model [14]. Nonetheless, the results presented here may be incomplete on account of these omissions. The inclusion of stratification, for example, is expected to increase the magnitude of supported CSW frequencies [10].

7 Conclusion

Motivated by the setting of Iceland, we investigated the effect of abutting marine ridges on the infinite discrete set of subinertial barotropic coastal trapped waves expected to resonate around islands. We utilized analytically derived dispersion relations for CSWs along simple geometries to inform our approach to the more complex 2D numerical problem, which was solved using spectral methods. This hierarchical approach substantially aided the interpretation of large, complex 2D eigenvalue problem output. Whilst the geometry considered is highly idealized relative to the bathymetry surrounding Iceland and the CSW model used makes a number of simplifying assumptions, our results suggest a potential mechanism for scattering low frequency CSWs away from Iceland-like islands. This may simplify the subinertial wave field around islands and reduce the number of resonant island trapped wave frequencies to be sought in observed spectra from infinitely many to a handful.

8 Acknowledgements

My sincerest thanks to Renske Gelderloos and Ted Johnson for their guidance and enthusiasm as my advisors on this project, it has been a really interesting problem, and I feel I've

learnt so much! Additional thanks to Keaton Burns for being so generous with his time and for his invaluable assistance in implementing the numerical problem with Dedalus v3 tools. Thanks also to Stefan Llewellyn Smith and Colm-cille Caulfield for their support as Program Directors, to Principal Lecturers Laure Zanna and Peter Schmid for their patience and commitment to our understanding, and to the 7 other fellows, staff, and visitors for keeping it all interesting and fun.

References

[1] K. H. Brink, *The effect of bottom friction on low-frequency coastal trapped waves*, Journal of Physical Oceanography, 12 (1982), pp. 127–133.

[2] K. H. Brink and D. C. Chapman, *Programs for computing properties of coastal-trapped waves and wind-driven motions over the continental shelf and slope*, Woods Hole Oceanographic Institution Tech. Rep., (1987).

[3] V. Buchwald and J. Adams, *The propagation of continental shelf waves*, Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 305 (1968), pp. 235–250.

[4] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, *Dedalus: A flexible framework for numerical simulations with spectral methods*, Physical Review Research, 2 (2020), p. 023068.

[5] M. Drivdal, J. E. H. Weber, and J. B. Debernard, *Dispersion relation for continental shelf waves when the shallow shelf part has an arbitrary width: Application to the shelf west of Norway*, Journal of Physical Oceanography, 46 (2016), pp. 537–549.

[6] GEBCO Compilation Group, *GEBCO 2022 Grid*, 2022. doi:10.5285/e0f0bb80-ab44-2739-e053-6c86abc0289c.

[7] R. Gelderloos, T. W. N. Haine, and M. Almansi, *Coastal Trapped Waves and Other Subinertial Variability alo the Southeast Greenland Coast in a Realistic Numerical Simulation*, Journal of Physical Oceanography, 51 (2021), pp. 861 – 877.

[8] ——, *Subinertial variability in four Southeast Greenland fjords in realistic numerical simulations*, (submitted).

[9] J. M. Huthnance, *On trapped waves over a continental shelf*, Journal of fluid mechanics, 69 (1975), pp. 689–704.

[10] ——, *On coastal trapped waves: Analysis and numerical calculation by inverse iteration*, Journal of Physical Oceanography, 8 (1978), pp. 74–92.

[11] ——, *Circulation, exchange and water masses at the ocean margin: the role of physical processes at the shelf edge*, Progress in Oceanography, 35 (1995), pp. 353–431.

[12] E. Johnson and J. Rodney, *Spectral methods for coastal-trapped waves*, Continental Shelf Research, 31 (2011), pp. 1481–1489.

[13] G. Kaoullas and E. Johnson, *Fast accurate computation of shelf waves for arbitrary depth profiles*, Continental Shelf Research, 30 (2010), pp. 833–836.

[14] P. H. LeBlond and L. A. Mysak, *Waves in the Ocean*, Elsevier, 1981.

[15] L. A. Mysak, *Topographically trapped waves*, Annual Review of Fluid Mechanics, 12 (1980), pp. 45–76.

[16] E. L. Ortiz and H. Samara, *An operational approach to the tau method for the numerical solution of non-linear differential equations*, Computing, 27 (1981), pp. 15–25.

[17] A. Robinson, *Continental shelf waves and the response of sea level to weather systems*, Journal of Geophysical Research, 69 (1964), pp. 367–368.

[18] J. T. Rodney and E. R. Johnson, *Localisation of coastal trapped waves by longshore variations in bottom topography*, Continental Shelf Research, 32 (2012), pp. 130–137.

[19] ——, *Meanders and Eddies from Topographic Transformation of Coastal-Trapped Waves*, Journal of Physical Oceanography, 44 (2014), pp. 1133–1150.

[20] ——, *Localised continental shelf waves: geometric effects and resonant forcing*, Journal of Fluid Mechanics, 785 (2015), pp. 54–77.

[21] L. Z. Sansón, *Simple models of coastal-trapped waves based on the shape of the bottom topography*, Journal of physical oceanography, 42 (2012), pp. 420–429.

[22] A. Wåhlin, O. Kalen, K. Assmann, E. Darelius, H. K. Ha, T.-W. Kim, and S. Lee, *Subinertial oscillations on the amundsen sea shelf, antarctica*, Journal of Physical Oceanography, 46 (2016), pp. 2573–2582.

Theory and Experiments on Deformable Porous Media: Wave Damping and Constitutive Relations

Tilly Woods

1 Introduction

A hydrogel is a soft, poroelastic material formed of a mixture of polymer chains and water molecules. A typical hydrogel starts off as a small (e.g., 1mm diameter), dry, solid bead, forming a network of cross-linked polymer chains. When placed in water, this bead absorbs water and swells, increasing its volume up to about 100 times. The result is a larger, soft, squishy bead referred to as a hydrogel. When a swollen hydrogel is left out in the air, it will gradually deswell, ejecting water and shrinking in size ([2]).

Hydrogels have many applications, due to their soft and absorbent properties, for example in contact lenses, nappies and wound dressings. They can also be used as a slow release of water for plants, and for biomedical applications such as tissue engineering (e.g., [1, 5]).

Another use for hydrogels is to form a laboratory model of a deformable porous medium, to improve understanding of real-world materials. We will focus on two-phase deformable porous media - a mixture of solid and fluid in which the solid structure can deform, for example seabed sediment or a saturated sponge. In a laboratory, a simple way to produce a two-phase deformable porous medium is to mix swollen hydrogels with water, creating a 'pack' of hydrogels. In this setup, the hydrogels are treated as the solid grains (ignoring the fact that they themselves are formed of a mixture of solid and water), and the water is the liquid.

In this work, we first, in section 2, consider the observed phenomenon that the presence of a layer of floating particles causes water waves to come to rest in finite time rather than decaying exponentially (e.g., [15, 16]). We observe this behaviour experimentally, using hydrogels as the floating particles, and attempt to capture the same results theoretically by modelling the layer of floating particles as a two-phase deformable porous medium.

In section 3, we turn to understanding the properties of a pack of hydrogels themselves. A crucial part of writing down a theoretical model for a deformable material, such as that in section 2, is a constitutive relation, which give us information about how the material deforms. That is, it gives a relationship between the material stresses and strains/strain rates, and depends on the specific material being used. Here, we carry out some one-dimensional compression experiments to test the suitability of the particular one-dimensional elastoviscoplastic constitutive relation proposed by [14] for a pack of hydrogels. Along the way, we encounter some interesting behaviour suggesting the potential degradation of hydrogels under repeated loadings.

2 'Sloshing' - Damping of Water Waves in Finite Time in the Presence of a Floating Particle Layer

Incoming ocean waves can cause the break up of sea ice around the poles. Waves can propagate hundreds of kilometres through the marginal ice zone (MIZ), causing break-up of the ice floes forming it. Smaller ice floes get more easily carried around by ocean currents, exposing more ocean to the atmosphere, leading to more heat exchange and increased ice melt ([7]).

To better understand the break up and loss of sea ice, we need to understand how far ocean waves propagate through sea ice. It has been previously assumed that decay is exponential, as is the case for the decay of waves with no floating particles/ice. However, it has more recently been suggested that observations suggest decay is in finite time (e.g., [15]). An everyday example of this is the way in which waves in a cup of water come to a rapid stop when there are ice cubes on top.

Carrying out lab experiments with floating ice is complicated by the fact that ice melts. A simplification is to use floating particles, such as hydrogels, which do not undergo phase change. This is reasonable because the melting of sea ice is unlikely to be important over the wave damping timescales. For example, [16] carried out experiments using hydrogels. These were floated on saltwater in a rectangular tank. Standing waves were then created by lifting and dropping one corner of the tank. Measuring the decay of amplitude of standing waves in time is easier than measuring the decay of propagating waves in space. These experiments showed that the waves, when a floating hydrogel layer was present, decayed in a finite time.

In addition to capturing the finite-time decay experimentally, it is desirable to develop a theoretical framework that can help us explain why the presence of a floating layer changes the rate at which the waves decay. Using a discrete model for the floating particles would be challenging. An easier approach would be to use a continuum model. However, the experiments carried out by [16] used a small number of layers of 'large' hydrogel particles (0.8-1.6 cm diameter), making a continuum approximation questionable. Therefore, we carried out experiments very similar to those done by [16] but with a larger number of layers of smaller hydrogels particles (7.0-7.5 mm), so the floating layer can slightly more reasonably be thought of as a continuum. We found that our experimental setup with smaller hydrogel particles still exhibits finite-time decay of waves, suggesting that a continuum two-phase model might be able to capture the behaviour too.

2.1 Experiments

The experimental setup is shown in Figure 1. Seawater of density 1.022 g/cm^3 was used to fill a rectangular tank of base 18.1×23.9 cm. A small amount of dish liquid was added to reduce surface tension, and green food colouring was added to make the water surface easier to identify. One thousand g of swollen hydrogels (a ratio 19:1 of clear to orange) of diameter $\sim 7.0 - 7.5$ mm were added to the saltwater such that the total depth was 20 cm. The hydrogels were swollen in fresh water so had density close to fresh water (~ 1.009 g/cm^3 for the clear, ~ 1.011 g/cm^3 for the orange), meaning that they formed a floating layer on top of the denser seawater. This layer was about 3.5 cm thick and was formed of

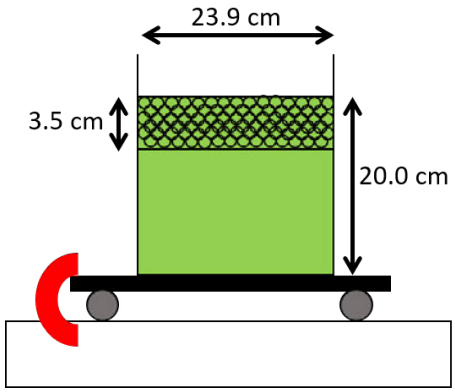


Figure 1: Setup for the 'sloshing' experiments.

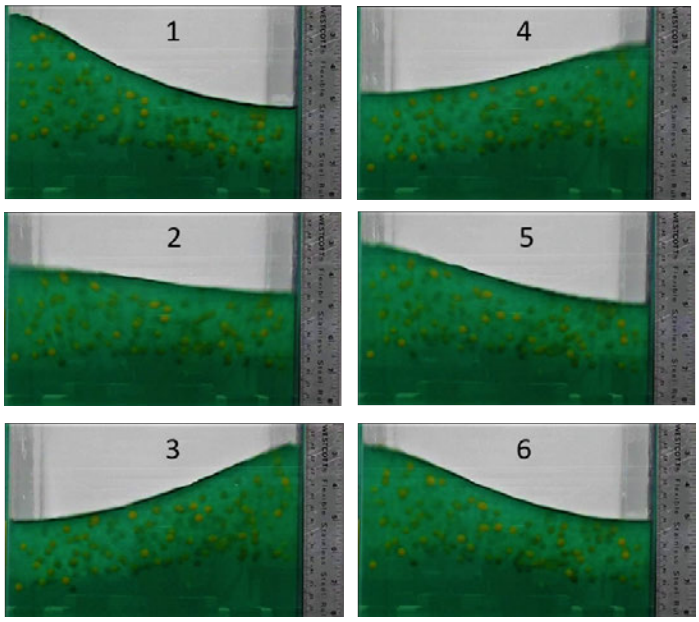


Figure 2: Snapshots of a 'sloshing' experiment taken every 0.125 s starting about 1 s after the tank had been clamping in place. The snapshots are labelled in order from 1 to 6.

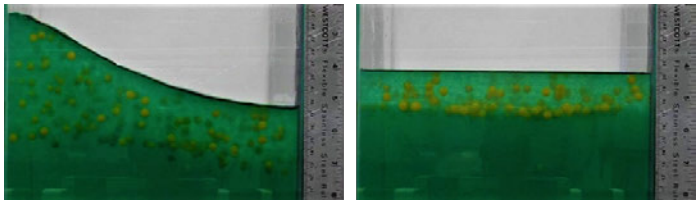


Figure 3: Snapshots taken at the start and end of a 'sloshing' experiment: as the standing wave has just been set up (the same as panel 1 from Figure 2) and once the wave amplitude has decayed to zero.

about 6 layers of hydrogels.

To create a standing wave in the tank, the tank was rocked from side to side in the direction parallel to the longer (23.9 cm) side of the tank. This was done by placing the tank on a platform with wheels that could roll back and forth. To create the waves, the platform was moved side to side by hand for a couple of seconds, with a range of motion of a couple of centimetres. Once a standing wave of appropriate amplitude was formed, the platform was clamped in place to prevent small oscillations of the platform from influencing the waves, and the wave damping was observed. The resulting waves were approximately two-dimensional, with very little variation in the direction parallel to the shorter (18.1 cm) side of the tank. Figure 2 shows some snapshots from one of our experiments, just as the standing wave has been set up. Figure 3 shows snapshots from the start and the end of the same experiment. This clearly shows that the thickness of the floating layer is not uniform in space or time, suggesting that it would be unreasonable to use a theoretical framework that assumes constant layer thickness.

It is worth noting that, over time, the freshwater hydrogels expel water and shrink when in saltwater, due to osmosis. Eventually, the densities of the hydrogels and water layer become equal, and so the hydrogels sink. We found that we had enough time to carry out one or two experiments with each set of hydrogels before they began to sink. A simple test with individual hydrogels in saltwater showed that the orange hydrogels started sinking first after about 8 minutes.

The decay of the wave amplitude was extracted from a video of the experiment taken with a camera about 60 cm away from the tank at approximately the same height as the water surface. The frame rate of the video was 24 frames per second. Using the MATLAB image processing software, the surface of the water (at the top of the hydrogel layer) was extracted from each frame of the video. A simple method to work out the decay rate of the wave is to track the surface height at one point and see how the oscillations of this point decay over time.

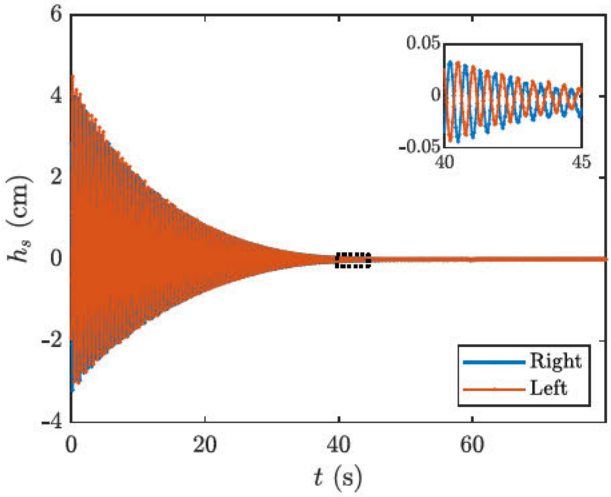


Figure 4: Evolution of the surface height on the left (red) and right (blue) of the tank for the sloshing experiment with floating hydrogels shown in Figure 2.

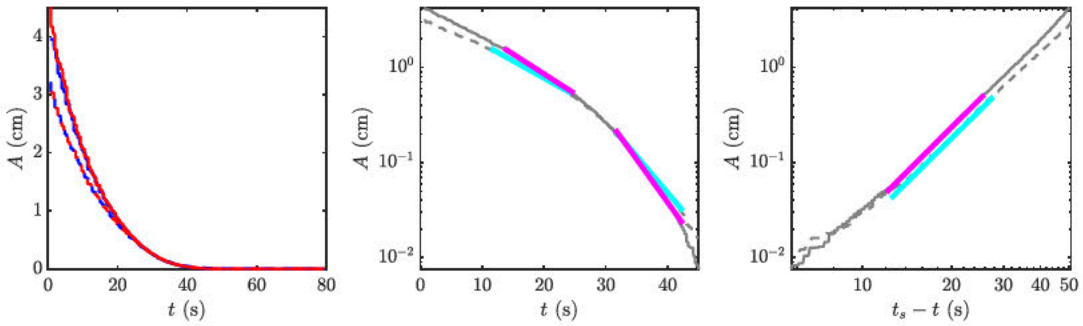


Figure 5: Left plot shows decay of the top (solid lines) and bottom (dashed lines) amplitudes for the experiment in Figure 4. Centre and right plots show mean top and bottom amplitudes (averaged over the left and right surface heights). The vertical axis is logarithmic in the centre plot and the best fit lines for $0.5 < A < 1.5$ cm and $0.02 < A < 0.2$ cm are shown in magenta and cyan. Both axes are logarithmic in the right plot and best fit lines for $0.05 < A < 0.5$ cm are shown in magenta and cyan.

To extract the decay rate of the waves, we looked at small 10-pixel windows on the left and right of the tank (at the start and end of the region where the water surface is not obscured by the tank side walls). Over each of these two windows, we calculated the mean surface height h_s and tracked this over time, as shown for one experiment in Figure 4. We can see that the oscillations of the surface are asymmetrical, at least for earlier times. This is likely due to nonlinear effects when the amplitude is large. To keep track of this asymmetry, we approximate both a 'top amplitude' and a 'bottom amplitude' of the surface oscillations. These two amplitudes are the top and bottom envelopes of the surface height plot, respectively, and are approximated as the maximum/minimum of the surface height h_s over 30 frames. Figure 5 (left) shows the top (solid lines) and bottom (dashed lines) amplitudes for the experiment shown in Figure 4. Once the amplitude gets small enough, the asymmetry between the top and bottom amplitudes disappears, supporting the suggestion that the asymmetry is due to nonlinear effects rather than experimental error.

Since our aim is to see if we can reproduce the decay rate observed by [16] but using smaller hydrogel particles, we calculated the decay rate from our data using a very similar method to [16]. We do this for three experiments: one with a floating layer of hydrogels (Figure 4, Figure 5), one with no hydrogels, just water (Figure 6), and one with a layer of hydrogels on the bottom (Figure 7). The latter will be discussed more later. In all experiments, the total depth was 20cm. In each case, we first fit an exponential

$$A \sim e^{-\frac{t}{\tau}}, \tag{1}$$

where τ is the e-folding time, to the two amplitude ranges $0.5 < A < 1.5$ cm and $0.02 < A < 0.2$ cm. For this fitting, we use the mean of the left and right amplitudes. Figure 5 (centre) and Figure 6 (centre) show the fits (magenta and cyan lines) to the mean top (solid grey line) and mean bottom (dashed grey line) when there are floating hydrogels and no hydrogels, respectively. Note that the vertical axis is scaled logarithmically. The e-folding times τ in each case are shown in Table 1. We see that the e-folding times are shorter when there is a floating hydrogel gel. Furthermore, we see that the amplitude drops off quicker

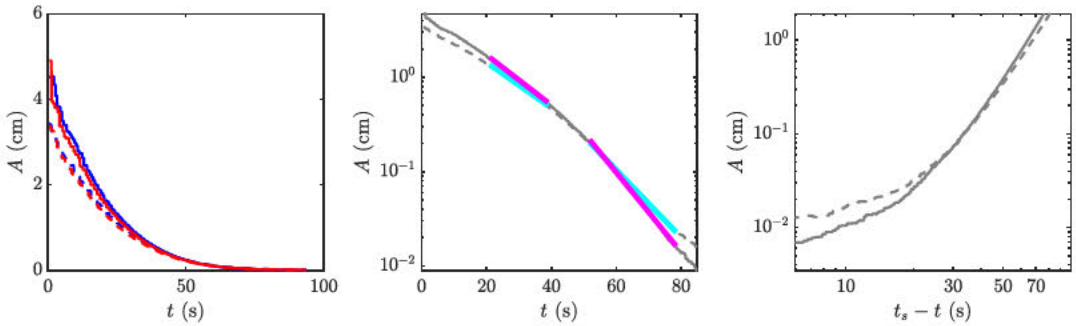


Figure 6: Left plot shows decay of the top (solid lines) and bottom (dashed lines) amplitudes for a sloshing experiment with no hydrogels. Centre and right plots show mean top and bottom amplitudes (averaged over the left and right surface heights). The vertical axis is logarithmic in the centre plot and the best fit lines for $0.5 < A < 1.5$ cm and $0.02 < A < 0.2$ cm are shown in magenta and cyan. Both axes are logarithmic in the right plot and best fit lines for $0.05 < A < 0.5$ cm are shown in magenta and cyan.

when there is a floating layer than when there is just water - the e-folding time halves from large to small amplitudes with a floating layer, compared to decreasing by about a third when there is just water. Once again following [16], we capture this drop off in amplitude at later times by using a fit of the form

$$A \sim (t_s - t)^q \tag{2}$$

for $0.05 < A < 0.5$ cm in the floating layer case, where t_s and q are fitting parameters. Figure 5 (right) shows this fit in magenta and cyan, with both axes scaled logarithmically. For comparison, we have also shown in Figure 6 (right) the water-only experiment with both axes scaled logarithmically (with $t_s = 93.4$s (top) or 94.3s (bottom) coming from trying to fit (2)). This is far from a straight line, unlike the same plot for the floating layer, showing that the finite-time decay fit is not appropriate for the water-only case. Back to the floating layer case, we find that, in the experiment shown in Figure 5, the stopping time t_s is 50.8 s for the top and 52.3 s for the bottom. The exponent q is 3.11 for the top and 3.08 for the bottom. In other experiments, q was closer to 2.5. These are very comparable to the range $2 < q < 3$ found by [16]. The fact that we have been able to reproduce the same decay rate as [16] but with smaller hydrogels suggests that the finite-time decay of waves might be caused by a floating layer rather than specifically floating particles. Therefore, it seems that it might be reasonable to use a continuum model to try and capture the behaviour theoretically, as we will look at shortly.

We now make a final comparison with the case where the hydrogel layer is beneath the water layer, which is similar to previous work on water waves over muddy seabed sediment (e.g., [17, 10]). The setup used is identical to the floating layer case apart from the fact that the seawater is replaced with freshwater so that the hydrogels sink to the bottom of the tank. As shown in Figure 7 (right), the amplitude of the waves decays similarly to the water-only case, without the rapid drop off in amplitude seen in the floating layer case. In fact, there is even less drop off in amplitude than in the water-only case. The e-folding

	Floating hydrogels	Hydrogels on bottom	No hydrogels
$0.5 < A < 1.5$ cm	10.4s (t), 12.0s (b)	12.9s (t), 13.1s (b)	16.3s (t), 17.9s (b)
$0.02 < A < 0.2$ cm	4.8s (t), 5.6s (b)	10.8s (t), 10.4s (b)	10.3s (t), 12.1s (b)

Table 1: The e-folding times τ calculated from the top (t) and bottom (b) amplitudes of the sloshing experiments with floating hydrogels (Figure 5), a hydrogel layer on the bottom (Figure 7) and no hydrogels (Figure 6). τ is calculated from an exponential fit in the two ranges of amplitude A shown.

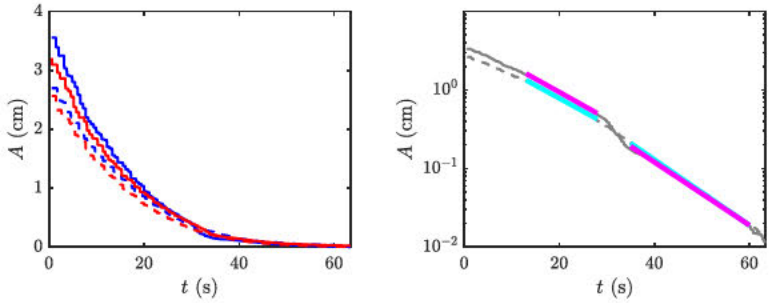


Figure 7: Left plot shows decay of the top (solid lines) and bottom (dashed lines) amplitudes for a sloshing experiment with a hydrogel layer on the bottom. Centre plot shows mean top and bottom amplitudes (averaged over the left and right surface heights). The vertical axis is logarithmic in the centre plot and the best fit lines for $0.5 < A < 1.5$ cm and $0.02 < A < 0.2$ cm are shown in magenta and cyan.

times τ for the bottom layer case are found to be similar to the water-only case, especially for smaller amplitudes (see Table 1). The wave damping seems largely unaffected by having a layer of hydrogels at the bottom.

2.2 Theory

In order to gain insight into why the presence of a floating layer causes water waves to come to rest in finite time, it would be helpful to have a theoretical framework that captures the finite-time decay. We propose modelling the floating layer as a two-phase deformable porous medium floating on water, choosing to use a continuum, rather than a discrete, model for the hydrogels.

We consider the two-dimensional setup shown in Figure 8. A single-phase layer of inviscid water sits on a lower boundary at $z = 0$. The water layer has upper surface $z = h(x,t)$, above which there is a two-phase layer of thickness $d(x,t)$. The upper surface of the two-phase region is at $z = s(x,t) = h(x,t) + d(x,t)$. Typical thickness of the water and two-phase layers are H and D, respectively, and a typical horizontal lengthscale (e.g., wavelength) is L.

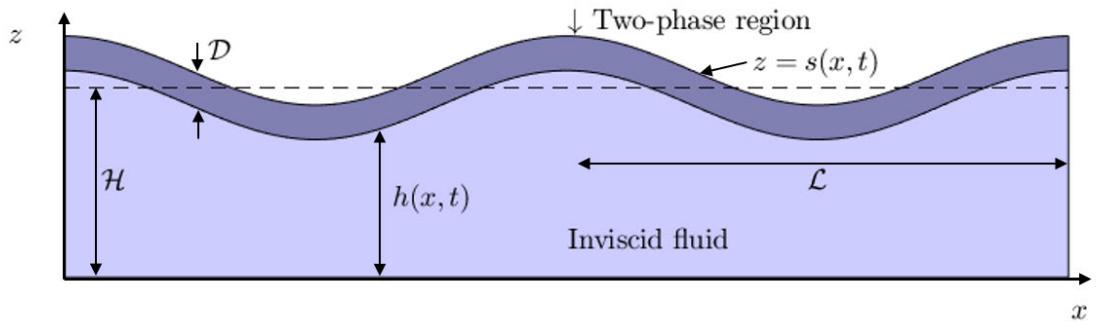


Figure 8: Setup for 'sloshing' theory.

2.2.1 Water layer equations

Assuming that the water layer $0 < z < h(x,t)$ is incompressible, the layer is governed by the incompressible Navier Stokes equations

$$\nabla \cdot \mathbf{u} = 0, \tag{3}$$

$$\rho_f \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla \tilde{p} - \rho_f g \mathbf{e}_z, \tag{4}$$

where $\mathbf{u} = (u, w)$ is the velocity in the water layer, $\tilde{p}$ is the pressure in the water layer, g is the acceleration due to gravity, ρ_f is the density of water, and $\mathbf{e}_z$ is the upward vertical unit vector.

2.2.2 Two-phase region equations

The two-phase region is composed of a fluid phase (water) of density ρ_f and velocity $\mathbf{u}_f = (u_f, w_f)$, and a solid phase of density ρ_s and velocity $\mathbf{u}_s = (u_s, w_s)$. The porosity ϕ is the volume fraction of fluid, with the solid fraction being $\Phi = 1 - \phi$. The pressure of the fluid (the pore pressure) is p.

The mass conservation equations for the fluid and solid phases are

$$\phi_t + \big(\phi u_f\big)_x + \big(\phi w_f\big)_z = 0, \tag{5}$$

$$(1-\phi)_t + \big((1-\phi) u_s\big)_x + \big((1-\phi) w_s\big)_z = 0, \tag{6}$$

respectively. The flux of the fluid relative to the solid is given by Darcy's law,

$$\phi(u_f - u_s) = -\frac{k}{\mu} p_x, \tag{7}$$

$$\phi(w_f - w_s) = -\frac{k}{\mu}\big(p_z + \rho_f g\big), \tag{8}$$

where k is the permeability and μ is the fluid viscosity. The bulk momentum conservation equations are

$$\big(\rho u_b\big)_t + \big(\rho u_b^2\big)_x + \big(\rho w_b u_b\big)_z = \big(\sigma_{xx} - p\big)_x + \big(\sigma_{xz}\big)_z, \tag{9}$$

$$\big(\rho w_b\big)_t + \big(\rho u_b w_b\big)_x + \big(\rho w_b^2\big)_z = \big(\sigma_{zz} - p\big)_z + \big(\sigma_{xz}\big)_x - \rho g, \tag{10}$$

where σ is the solid effective stress tensor, which can be decomposed as $\sigma = -P_e I + \tau$, where P_e is the effective solid stress and τ is the (traceless) deviatoric stress tensor. We have also introduced the bulk density

$$\rho = \rho_f \phi + \phi_s (1 - \phi) \tag{11}$$

and the bulk velocity $\mathbf{v}_b = (u_b, v_b)$, which satisfies

$$\rho \mathbf{v}_b = \rho_f \phi \mathbf{v}_f + \rho_s (1 - \phi) \mathbf{v}_s. \tag{12}$$

Note that the bulk density and velocity satisfy the bulk mass conservation equation

$$\rho_t + \big(\rho u_b\big)_x + \big(\rho w_b\big)_z = 0. \tag{13}$$

2.2.3 Boundary conditions

The top surface $z = s(x,t) = h(x,t) + d(x,t)$ is a common surface for both the fluid and the solid, so we have the two kinematic conditions

$$s_t + u_f s_x = w_f \quad \text{at} \quad z = s, \tag{14}$$

$$s_t + u_s s_x = w_s \quad \text{at} \quad z = s. \tag{15}$$

The surface is also stress-free, so

$$\big(\sigma - pI\big)\mathbf{n} = \mathbf{0} \quad \text{at} \quad z = s, \tag{16}$$

where

$$\mathbf{n} = \frac{(-s_x, 1)}{\sqrt{s_x^2 + 1}} \tag{17}$$

is the outward unit normal to the surface $z = s$. In components, the stress-free condition can be written as

$$\sigma_{xz} - s_x(\sigma_{xx} - p) = 0 \quad \text{at} \quad z = s, \tag{18}$$

$$\sigma_{zz} - p - s_x \sigma_{xz} = 0 \quad \text{at} \quad z = s. \tag{19}$$

In addition to the four boundary conditions (14), (15), (18), (19), we also need an additional stress to be exerted on each phase to ensure that $z = s$ remains a shared material surface for both phases. For example, capillary forces to resist the solid grains (e.g., hydrogels) dropping below the water surface, or the localised body force of gravity to resist grains emerging above the water surface. Whatever the condition is, it means that we cannot impose any other stress conditions, in particular, we cannot impose that the pore pressure p is atmospheric.

The lower surface $z = h(x,t)$ of the two-phase region is the interface between the two-phase region and the water-only region, and is a material surface for the solid but not the fluid. Water can flow into and out of the porous material. The kinematic condition for the solid is

$$h_t + u_s h_x = w_s \quad \text{at} \quad z = h. \tag{20}$$

We also have continuity of pressure,

$$p = \tilde{p} = P(x,t) \quad \text{at} \quad z = h, \tag{21}$$

where, $P(x,t)$ is the shared value of the pore pressure p and the water pressure $\tilde{p}$ at the interface. We also require conditions for the flux of mass and momentum across the interface, which arise due to the flow of water across the interface. These come in the form of jump conditions given by [4]. Firstly, there must be no jump in normal mass flux relative to the interface $z = h$, i.e.,

$$\big[\rho(\mathbf{v}_b - \mathbf{v}_i)\big]_{z=h^+} \cdot \mathbf{n} = \big[\rho(\mathbf{v}_b - \mathbf{v}_i)\big]_{z=h^-} \cdot \mathbf{n}, \tag{22}$$

where

$$\mathbf{n} = \frac{(-h_x, 1)}{\sqrt{h_x^2 + 1}} \tag{23}$$

is the upward unit normal to $z = h$, and

$$\mathbf{v}_i = (0, h_t) \tag{24}$$

is the velocity of the interface. By using the kinematic condition (20), the mass flux condition simplifies to

$$\phi\big(w_f - u_f h_x - h_t\big) = w - u h_x - h_t \quad \text{at} \quad z = h. \tag{25}$$

Similarly, the jump condition for the momentum given by [4] is

$$\big[\rho \mathbf{v}_b(\mathbf{v}_b - \mathbf{v}_i) + \sigma - pI\big]_{z=h^+} \cdot \mathbf{n} = \big[\rho_f \mathbf{v}(\mathbf{v} - \mathbf{v}_i) - \tilde{p}I\big]_{z=h^+} \cdot \mathbf{n}, \tag{26}$$

which simplifies to

$$\rho_f \phi\big(w_f - u_f h_x - h_t\big)(\mathbf{v}_b - \mathbf{v}) + \sigma(-h_x, 1) = 0 \quad \text{at} \quad z = h, \tag{27}$$

by using the kinematic condition (20) and mass flux condition (25).The condition (27) says that when there is a flux of water $\phi(w_f - u_f h_x - h_t)$ into the porous medium, the momentum excess or deficit $\rho_f(\mathbf{v}_b - \mathbf{v})$ exerts a force on the solid. In components,

$$\sigma_{xz} - h_x \sigma_{xx} + \rho_f \phi(w_f - u_f h_x - h_t)(u_b - u) = 0 \quad \text{at} \quad z = h, \tag{28}$$

$$\sigma_{zz} - h_x \sigma_{xz} + \rho_f \phi(w_f - u_f h_x - h_t)(w_b - w) = 0 \quad \text{at} \quad z = h. \tag{29}$$

2.2.4 Shallow water equations

To simplify the problem in the water layer, we assume that the water layer is relatively shallow ($H \ll L$), and hence use the shallow water approximation, in which vertical derivatives are larger than horizontal derivatives ($\partial/\partial z \gg \partial/\partial x$) and vertical velocities are smaller than horizontal velocities ($w \ll u$). It is also assumed that the the horizontal velocity $u = u(x,t)$ is independent of z. The shallow-water water pressure $\tilde{p}$ is hydrostatic,

$$\tilde{p}(x,t) = \rho_f g(h - z) + P(x,t), \tag{30}$$

and the horizontal component of the momentum equation (4) becomes

$$\rho_f\big(u_t + uu_x\big) = -\rho_f g h_x - P_x. \tag{31}$$

Depth-integrating the continuity equation (3) and using that there is no vertical velocity at $z = 0$ gives

$$h_t + \big(hu\big)_x = -w|_{z=h}. \tag{32}$$

We also assume that the two-phase layer is shallow ($D << L$), but we do not make any assumptions about the relative thicknesses of the two layers.

2.2.5 Membrane model

The two-phase equations can be simplified by making a further approximation. We choose to take the membrane limit, but note that this is just one possible approximation (with others including the bending limit, for example). The membrane limit can be thought of as the two-phase layer stretching horizontally and thinning vertically. The layer is dominated by extensional (rather than shear) stresses. We assume that horizontal speeds can be much greater than vertical ones, and vertical derivatives are greater than horizontal ones. This leads to an imbalance in the two components (7) and (8) of Darcy's law. To respect both the derivative and velocity assumptions, only one of the two components can balance. There are two options:

- **Version 1**: horizontal components balance, giving

$$\phi(u_f - u_s) = -\frac{k}{\mu} p_x, \tag{33}$$

$$0 = -\frac{k}{\mu}\big(p_z + \rho_f g\big), \tag{34}$$

  i.e., there is Darcy flow horizontally and the pore pressure is hydrostatic.

- **Version 2**: vertical components balance, giving

$$\phi(u_f - u_s) = 0, \tag{35}$$

$$\phi(w_f - w_s) = -\frac{k}{\mu}\big(p_z + \rho_f g\big), \tag{36}$$

  i.e., there is no slip between the solid and the fluid, and there is Darcy flow in the vertical.

In what follows, we will use version 1.

For slippery particles (like hydrogels), we expect there to be little shear or shear stress (i.e., $|\sigma_{xz}| \ll |\sigma_{xx}|, |\sigma_{zz}|$) and the normal effective stresses to be dominated by the effective pressure (i.e., $\sigma_{xx} \sim \sigma_{zz} \sim -P_e$). The bulk momentum equations (9) and (10) become

$$\big(\rho u_b\big)_t + \big(\rho u_b^2\big)_x + \big(\rho w_b u_b\big)_z = -\big(P_e + p\big)_x + \big(\sigma_{xz}\big)_z, \tag{37}$$

$$0 = -\big(P_e + p\big)_z - \rho g. \tag{38}$$

The stress conditions (18), (19), (28) and (29) become

$$\sigma_{xz} + s_x(P_e + p) = 0 \quad \text{at} \quad z = s, \tag{39}$$

$$-P_e - p = 0 \quad \text{at} \quad z = s, \tag{40}$$

$$\sigma_{xz} + h_x P_e + \rho_f \phi (w_f - u_f h_x - h_t)(u_b - u) = 0 \quad \text{at} \quad z = h, \tag{41}$$

$$-P_e = 0 \quad \text{at} \quad z = h. \tag{42}$$

To close the problem, we need a constitutive law for the solid effective stress P_e. A simple case is to take

$$P_e = P_e(\phi) = m(\phi_g - \phi), \tag{43}$$

for $\phi < \phi_g$, based on the yield stress used by [13] for suspensions of cellulose fibres in water. Here ϕ_g is the value of the porosity at the 'gel point', that is, the porosity at which there is no stress on the solid ($P_e(\phi_g) = 0$). We can simplify the equations further by depth-integrating and assuming that the density difference $\rho_f - \rho_s$ is small. Substituting (43) into the vertical momentum equation (38) and using vertical Darcy's law (34) leads to an equation for the porosity ϕ,

$$m\phi_z = -(\rho_f - \rho_s)(1 - \phi)g. \tag{44}$$

Integrating, using the no-stress condition that $\phi = \phi_g$ at $z = h$, gives

$$\phi = 1 - (1 - \phi_g)e^{\Gamma(z-h)} \approx \phi_g - (1 - \phi_g)\Gamma(z - h), \tag{45}$$

for small density differences $\rho_f - \rho_s$, where

$$\Gamma = \frac{(\rho_f - \rho_s)g}{m}. \tag{46}$$

The vertical Darcy's law (34) tells us that the pore pressure is hydrostatic,

$$p = P(x, t) + \rho_f g(z - h). \tag{47}$$

Integrating the vertical bulk momentum equation (38) gives

$$P_e + p = \int_z^s \rho g \mathrm{d}z. \tag{48}$$

Evaluating this at $z = h$ tells us the shared value of the pressure at the interface,

$$P(x, t) = \left(\rho_f \bar{\phi} + \rho_s(1 - \bar{\phi})\right) g d, \tag{49}$$

where

$$\bar{\phi} = \frac{1}{d}\int_h^s \phi \mathrm{d}z = 1 - (1 - \phi_g)\frac{e^{\Gamma d} - 1}{\Gamma d} \approx \phi_g - \frac{1}{2}(1 - \phi_g)\Gamma d \tag{50}$$

is the depth-integrated porosity over the two-phase region, with the approximation being for small density differences.

We can simplify the form of the horizontal solid velocity u_s by enforcing that the shear stress is small ($|\sigma_{xz}| \ll |\sigma_{xx}|$). Let $\epsilon = D/L$, which we are assuming to be small. It is reasonable to expect that the shear strain rate $\dot{\gamma}_{xz} = \frac{1}{\epsilon}u_{sz} + \epsilon w_{sx}$ should be small compared to the normal strain rate $\dot{\gamma}_{xx} = 2u_{sx}$ when $|\sigma_{xz}| \ll |\sigma_{xx}|$. We can make the shear strain rate small by enforcing that $u_{sz} = 0$, i.e., that u_s is independent of z, $u_s \approx U_s(x,t)$. Using this assumption, the horizontal Darcy's law (33) tells us

$$\phi u_f \approx \phi U_s - \frac{k}{\mu}\Big(P_x - \rho_f g h_x\Big). \tag{51}$$

Finally, depth-integrating the mass and momentum conservation equations (5), (6) and (37) gives

$$\Big(\bar{\phi}d\Big)_t + \Big(\bar{\phi}dU_s - \frac{k}{\mu}\big(P_x - \rho_f g h_x\big)d\Big)_x - h_t - (hu)_x = 0, \tag{52}$$

$$\Big((1-\bar{\phi})d\Big)_t + \Big((1-\bar{\phi})U_s d\Big)_x = 0, \tag{53}$$

$$\begin{aligned}\Big[\rho_f\Big(\bar{\phi}U_s d - \frac{k}{\mu}\big(P_x - \rho_f g h_x\big)d\Big) + \rho_s(1-\bar{\phi})U_s d\Big]_t \\ + \Big[\big(\rho_f\bar{\phi} + \rho_s(1-\bar{\phi})\big)U_s^2 d + \frac{\rho_f k}{\mu}(P_x - \rho_f g h_x)\Big(\int_h^s \frac{1}{\rho_f\phi + \rho_s(1-\phi)}\mathrm{d}z - 2U_s d\Big)\Big]_x \\ + \big[\rho w_b u_b\big]_{z=h}^{s} = -\Big[md(\phi_g - \bar{\phi}) + Pd + \frac{1}{2}\rho_f g d^2\Big]_x + \big[\sigma_{xz}\big]_{z=h}^{s}\end{aligned} \tag{54}$$

By substituting (49) for $P(x,t)$, (45) for ϕ and (50) for $\bar{\phi}$ into (52), (53) and (54), applying the boundary conditions and invoking the small density difference approximation (i.e., assuming Γ is small), we can reduce (31), (52), (53) and (54) to four equations for u, h, d and U_s, giving us a closed system.

2.2.6 Method for understanding the finite-time decay theoretically

We hope that our relatively simple system of equations will provide us with a way to understand the finite-time decay of water waves under a layer of floating particles observed in experiments. Although not done yet, the future steps to take will be outlined here.

Firstly, our theoretical model can be used to look at shallow water waves, relevant to the sloshing water wave problem. Our main aim is to capture the decay rate of the amplitude of these waves. This can be done by formulating an energy equation from our system of equations (without having to solve the equations). Each term in the energy equation will look like some power of the wave amplitude A. If we were to consider the water-only problem without the floating two-phase layer, we would get an energy equation that looks something like

$$\frac{\mathrm{d}}{\mathrm{d}t}A^2 = -\alpha A^2, \tag{55}$$

which tells us that the amplitude decays like $A^2 \sim e^{-\alpha t}$. We hope that the energy formulation from our two-phase floating layer problem will lead to there being an additional term in the right-hand side with a smaller power of A that will dominate for smaller amplitudes.

In particular, the presence of a term of the form $A^{(2q-1)/2}$ will lead to the amplitude behaving like $A \sim (t_s - t)^q$ for small enough amplitudes, which is the behaviour observed in the experiments. The extra term in the energy equation will come from dissipation terms due to the presence of the floating layer. Carrying out the analysis will enable us to say which dissipative processes in particular are causing this presence of the term in the energy equation which leads to the damping of waves in finite time.

3 One-dimensional Constitutive Relation and Compressions

3.1 Constitutive relation model

When modelling a saturated pack of hydrogels, we need a constitutive law to close the system. The constitutive law gives a relationship between the stress σ and the strain e and or strain rates $\dot{e}$. The traditional description of a deformable porous medium is poroelasticity ([3]). This assumes that deformation is purely elastic. However, experiments looking at flow-driven deformation of a pack of hydrogels suggest that (1) the timescales suggested by an elastic model are too short, suggesting there is also a viscous component of the deformation ([6]), and that (2) particle rearrangement (irreversible, plastic deformation) also plays a role ([9]). Motivated by this, one proposed one-dimensional constitutive relation for a saturated pack of hydrogels is the elastoviscoplastic relation for the effective solid stress (compressive) P_e suggested by [14]:

$$\frac{1}{\varepsilon(\Phi)}\frac{\mathrm{d}P_e}{\mathrm{d}t} + \frac{1}{\Lambda(\Phi)}\max\left(0, \frac{|P_e| - P_y(\Phi)}{|P_e|}\right)P_e = -\dot{e}, \tag{56}$$

where $\dot{e}$ is the strain rate (tensile), $\varepsilon(\Phi)$ is the bulk (elastic) modulus, $\Lambda(\Phi)$ is the bulk viscosity and $P_y(\Phi)$ is the compressive, plastic yield stress. The material deforms purely elastically when $|P_e| < P_y$ (the material is unyielded). When $|P_e| > P_y$ (the material is yielded), there is rate-dependent plastic deformation too. The strain rate $\dot{e}$ comes from how we deform the material (how the surface height h changes in the one-dimensional setup shown in Figure 9), and can be expressed as

$$\dot{e} = \frac{\dot{h}}{h}, \tag{57}$$

where h is the thickness of the hydrogel layer. The strain rate can also be expressed in terms of the solid fraction $\Phi = 1 - \phi$ by deriving a relationship between h and Φ from the global one-dimensional solid mass conservation equation. Since the solid fraction is uniform under mechanical compression (e.g., [12]), Φ is independent of z. Hence, the global solid mass conservation is simply

$$h\Phi = h_0\Phi_0, \tag{58}$$

where h_0 and Φ_0 are the initial layer thickness and solid fraction, respectively. Using this relationship, the strain rate in terms of the solid fraction Φ is

$$\dot{e} = \frac{\dot{h}}{h} = -\frac{\dot{\Phi}}{\Phi}. \tag{59}$$

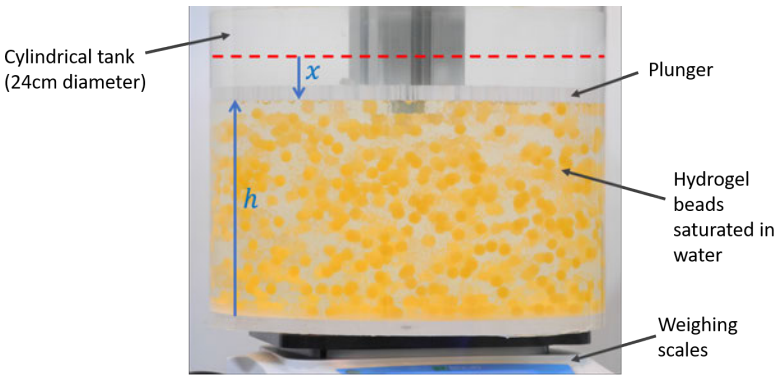


Figure 9: The experimental setup for the compression tests. The red dashed line is an illustration of the starting position of the bottom of the plunger (referred to as the origin, 13.1 cm above the base of the tank). x is the distance of the plunger below the origin. h is the thickness of the hydrogel layer.

For a given deformation, the change in the layer thickness is controlled and known, and hence so is the strain rate $\dot{e}$. On the other hand, $\varepsilon(\Phi)$, $\Lambda(\Phi)$ and $P_y(\Phi)$ are unknown parametric functions of the solid fraction Φ (or equivalently the porosity ϕ) which must be determined empirically. This can be done by carrying out suitable compression tests which isolate elastic, rate dependent and plastic behaviour to calculate ε, Λ and P_y, respectively.

3.2 Compression tests

In this section, we will describe the experimental setup used to fit the parametric functions in the elastoviscoplastic constitutive relation. However, before turning to the experimental protocols used to extract the parametric functions, we first use the same setup (described below) to carry out some simple one-dimensional compression tests to assess the reproducibility of our experiments. The aim was to see if repeating the same experiment leads to the same results and whether there is a dependence on the speed at which the hydrogels are compressed. We will come back to the constitutive relation in section 3.3.

The setup for the compression tests is shown in Figure 9. We put a saturated mixture of hydrogels and water into a cylindrical tank of 24.0 cm diameter. The pack of hydrogels was compressed from above by a plunger: a disc of diameter slightly smaller than the tank. The disc had holes smaller than the diameter of the hydrogels drilled into it. The holes and the gap around the edge act to reduce the resistance of the plunger moving through the water, by letting through water but not hydrogels. The plunger was linked to a stepper motor that could be controlled by a computer, allowing speeds as low as 0.01 mm/s. The tank was placed on top of some weighing scales to allow the load to be measured as the hydrogel pack was compressed by the action of the stepper motor. The scales output readings to the computer every 0.1 s, and had precision of 0.1 g. The load (Pa) is calculated from the mass M (kg) as load $= Mg/A$, where $g = 9.81$ m/s^2 is acceleration due to gravity and $A = \pi r^2$ is the area of the base of the tank (m^2), with $r = 0.12$ m being the radius.

Various batches of hydrogels were used for the compression tests: batches 3, 5 and 6. All are composed of approximately 3450 g of swollen hydrogels of diameter $\sim 7.0 - 7.5$

mm ($\sim$ 3280 g clear, $\sim$ 170 g orange) mixed with 3000 g fresh water. The hydrogels are the same type used in section 2.1. Calculating the mass of the drained, swollen hydrogels accurately is challenging because it is very difficult to ensure that all the water has been drained away, introducing considerable error.

All the compression tests started with the plunger submerged in the water but not touching the hydrogels, at a height of 13.1 cm above the bottom of the tank. This position will be referred to as the origin, with x being the distance of the plunger below the origin. As a note, before the plunger makes contact with the hydrogels, the load looks to increase linearly at a slow rate. This is due to more of the plunger becoming submerged, which raises the water level and hence the centre of mass of the setup on the scales. This in turn slightly increases the load. We have verified that the same linear increase in load is observed when the experiment is repeated with only water, no hydrogels. In all of the compression test plots shown, the water-only load has been subtracted from the load measured in the compression tests.

In the tests with hydrogel batch 3 (see Figure 10), the plunger was then lowered by 20 mm at a fixed speed of 0.1 mm/s, then held in place for 40 seconds before the load was released. Between experiments, the setup was reset by raising the plunger out of the tank, inserting one end of some plastic tubing into the hydrogel pack and blowing air through it to mix up the hydrogels using the resulting bubble motion. The plunger was then lowered to the origin and the hydrogels setup was allowed to settle for 5 minutes (until the reading on the scales stabilised). Once stabilised, the scales were tared and the next compression test commenced. The experiments were carried out at roughly the same temperature to ensure the results were not affected by any temperature dependence.

Figure 10 shows the load against position of the plunger below the origin for the batch 3 compression tests described above. Twenty different runs are shown, labelled in the order they were carried out. We see from this that there is a general trend of the load decreasing as more runs are carried out. The fact that the hydrogel pack is mixed up between successive runs rules out that the particles are getting more and more tightly packed (due to particle rearrangements) each run. An alternative explanation could be that water is being squeezed out the of the hydrogels when they are under load, and that the hydrogels do not have enough time to fully reswell between runs. However, the decreasing trend persists even when the hydrogels are left for multiple days to recover between runs (5 days between solid and dashed lines and 2 between dashed and dotted in Figure 10), which should be plenty of time. Therefore, we propose that this decreasing trend could be due to the hydrogels becoming permanently damaged by the applied loads. In particular, we suggest that the polymer chains forming the solid skeleton of the hydrogels are being damaged. Some hydrogels are designed to degrade in this way, for example for use in biomedical applications. The polymer chains or cross-links between them in such hydrogels break down under certain conditions, such as the presence of certain chemicals (e.g., [11]). The polymer chains themselves can also be chosen to influence the degradation ([8]).

If the hydrogels are getting damaged by loading, it could be that applying a lower load would prevent this damage. To investigate this, we carried out some compression tests at a lower load with a new batch of hydrogels (batch 5). The results are shown in Figure 11. The compressions were the same as for batch 3, but with the plunger only being lowered by 15 mm, not 20 mm. Also, the load was not held before being released for the first 5

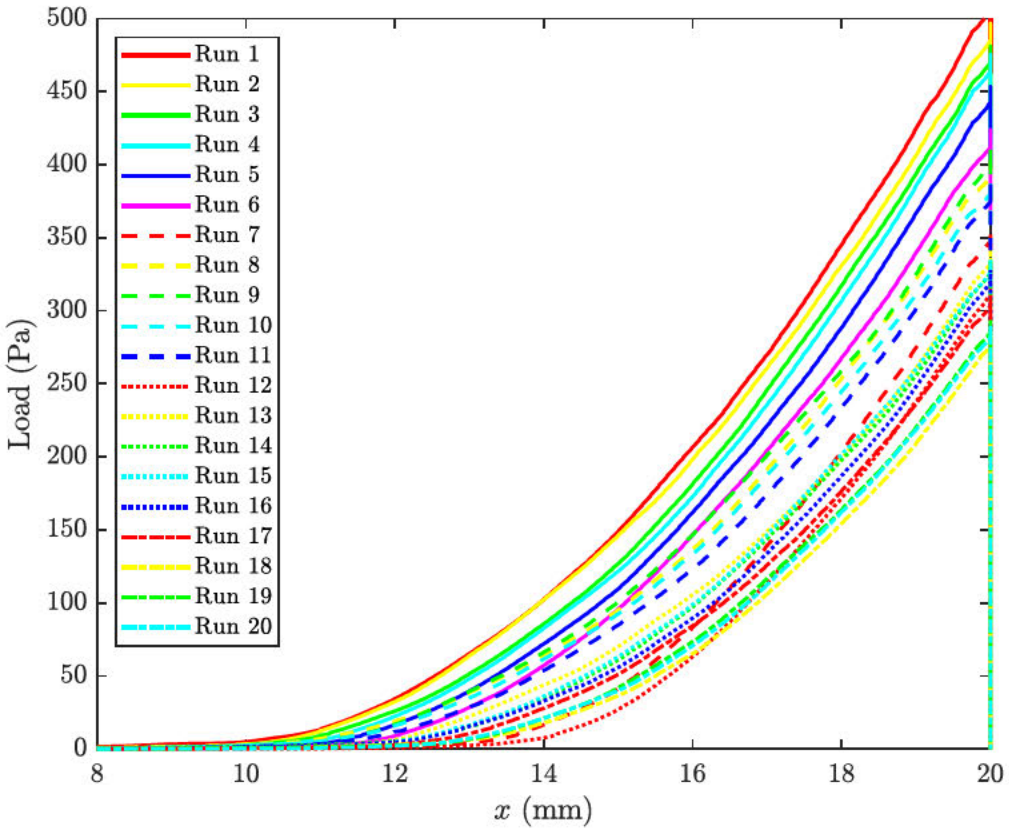


Figure 10: Compression tests with hydrogel batch 3 at 0.1 mm/s. The different line styles represent runs carried out on different days: day 1 (solid), day 6 (dashed), day 8 (dotted and dot-dashed).

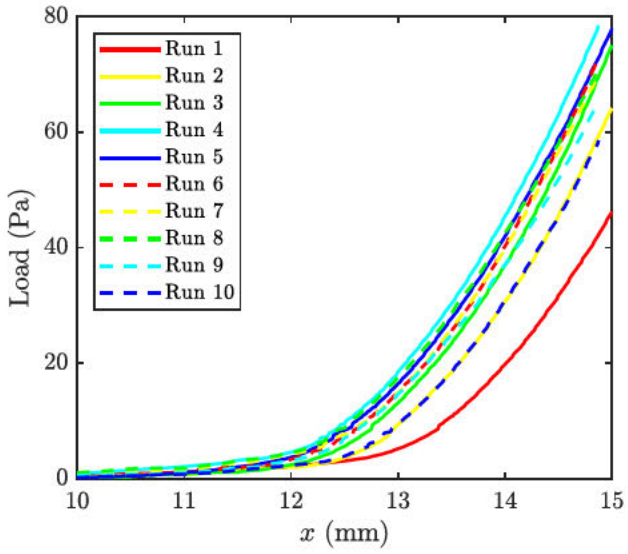


Figure 11: Low load compression tests with hydrogel batch 5 at 0.1 mm/s. All tests were carried out on the same day. The load was released immediately for some (solid lines) and held for 40 seconds before releasing for others (dashed lines).

runs (solid lines) with batch 5. The results for batch 5 do not follow such a clear trend. Curiously, the load increased across successive runs for the first 5 runs. Later runs seem to fluctuate in load, not showing any clear trend. In particular, there is not a clear trend of lowering load over successive runs. This could suggest that lower loads do not damage the hydrogels so much, but results are inconclusive.

Figure 12 (left) shows results from another set of low load compression tests, this time with hydrogel batch 6, lowering by 16 mm. The load was released immediately after each compression, then the hydrogels were mixed and left for a couple of minutes before the next run. This time, there does seem to be the overall trend of decreasing load across runs, even with the lower load. However, the results are still inconclusive; in the first two runs, the load was accidentally increased to about 130 Pa and 75 Pa, respectively, so the hydrogels could have been damaged by these higher loads, making further damage by lower loads possible thereafter. Once again, all the idea that the solid skeleton of the hydrogels is being damaged by applied loads is speculative.

The fact that there is so much variation between compression tests at the same speed makes if difficult to assess whether there is any speed dependence on top of that. Figure 12 (right) shows the results of some further batch 6 compression tests carried out at higher speeds: 0.2 mm/s (blue) and 0.4 mm/s (red), on top of the 0.1 mm/s compression tests (grey). The higher speed compression tests show no discernible speed dependence and lie within the range of variation of the 0.1 mm/s tests, suggesting that there might not be very much speed dependence, but the variation within one speed makes it hard to say anything definitive.

We also investigated the change in load under back-to-back compressions, without mixing up the hydrogels between runs. This was done for batch 3, after the other batch 3

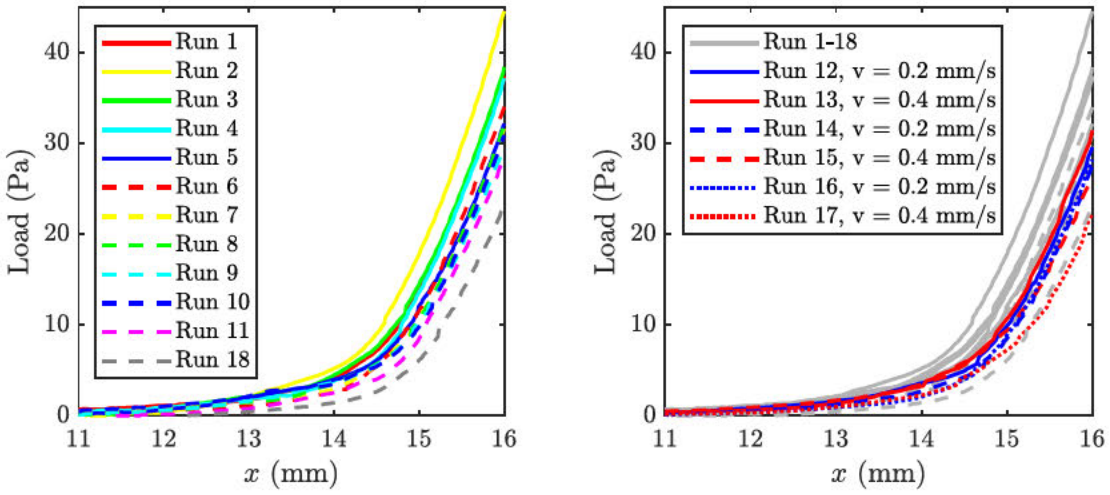


Figure 12: Left panel: low load compression tests with hydrogel batch 6 at 0.1 mm/s. All tests were carried out on the same day. The different line styles are simply to avoid running out of colours. Right panel: the runs from the left panel are shown in grey, tests at 0.2 mm/s (blue) and 0.4 mm/s (red) added on top. Note that the loads in run 1 and run 2 accidentally reached about 130 Pa and 75 Pa, respectively (not plotted).

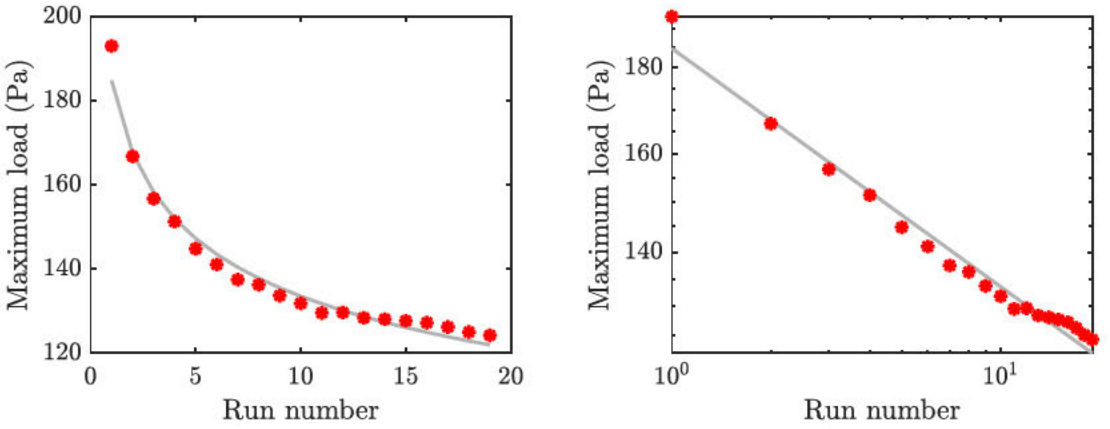


Figure 13: Left panel: the maximum loads reached in 19 back-to-back compressions to 20 mm of batch 3, carried out after the other batch 3 compression tests. The grey line shows the fit from fitting a straight line to the log-log plot shown in the right panel.

compression tests. Figure 13 shows the maximum load (red stars) reached in 19 back-to-back compressions where the plunger was lowered by 20 mm. The load decreases over runs, looking like it might be stabilising as more runs are carried out. The reason for the decrease in load in the back-to-back runs is the hydrogels becoming more and more tightly packed each run, due to irreversible particle rearrangements. The grey line in Figure 13 comes from fitting a straight line to the log-log plot (right panel). This suggests that the maximum load decreases like (number of runs)$^{-0.14}$.

3.3 Determining P_y, ε and Λ

We now move on to look at how more complicated protocols of loading and unloading, rather than just simple compressions, can be used to determine the functions $P_y(\Phi)$, $\varepsilon(\Phi)$ and $\Lambda(\Phi)$ in the elastoviscoplastic constitutive law (56). However, we should treat these results with caution; our compression tests suggest that repeated loading changes the material properties of the hydrogel pack, so the functions we find are likely to be different for the same batch of hydrogels after repeated loadings.

Firstly, we focus on P_y. Under compression, the maximum term in the elastoviscoplastic relation (56) is $\big(P_e - P_y(\Phi)\big)/\Lambda$. If we also consider quasi-steady compressions, the $\mathrm{d}P_e/\mathrm{d}t$ and $\dot{e}$ terms are small and can be neglected, so the constitutive relation becomes

$$P_e = P_y(\Phi), \tag{60}$$

that is, the material follows the yield stress curve. Hence, the load curve that results from quasi-steady compressions (like those studied in section 3.2) gives us the yield function $P_y(\Phi)$. Figure 14 shows an example experimental P_y, with fits (by eye) for smaller and larger Φ shown in black and red. These suggest that $P_y(\Phi) \sim (\Phi - 0.6)$ for smaller Φ and $P_y(\Phi) \sim (\Phi - 0.6)^5$ for larger Φ.

To determine the elastic modulus ε, we consider quasi-steady unloadings: we carry out quasi-steady loading, as for finding P_y, but unload and reload by small amounts along the way, as shown in Figure 15. During the unloading sections, the effective stress P_e drops below the yield stress and the material reverts to being unyielded, exhibiting purely elastic behaviour. The constitutive law (56) becomes

$$\frac{\mathrm{d}P_e}{\mathrm{d}t} = -\varepsilon(\Phi)\dot{e}. \tag{61}$$

Using the expression (59) for the strain rate in terms of the solid fraction, this can be rewritten as

$$\varepsilon = \frac{\mathrm{d}P_e}{\mathrm{d}\Phi}\Phi. \tag{62}$$

Hence, the bulk elastic modulus ε can be calculated from the slope of the unloading sections of the quasi-steady load curve. Carrying out small unloadings at several different values of solid fraction allows the function $\varepsilon(\Phi)$ to be reconstructed. For the example in Figure 15, the rough fit we find in $\varepsilon(\Phi) = 335(\Phi - 0.635)$ kPa.

The bulk viscosity Λ is associated with rate-dependent deformation. Therefore quasi-steady loading will not allow us to extract Λ. One method to include rate dependence is to consider a 'step change': quickly compress the material by a small amount then hold the

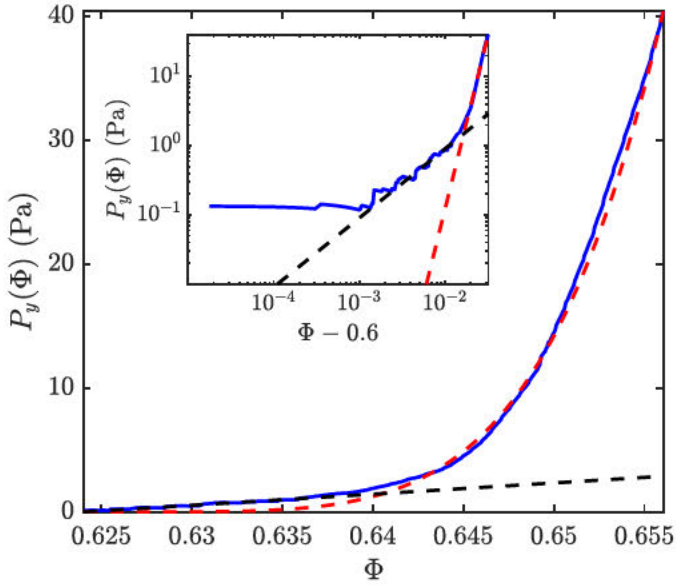


Figure 14: The load plot for quasi-steady loading at 0.1 mm/s (blue line). The black and red dashed lines are fits by eye of the form $0.9 \times 10^2(\Phi - 0.6)$ Pa and $1.2 \times 10^9(\Phi - 0.6)^5$ Pa, respectively. The inset shows a log-log plot with the horizontal axis shifted.

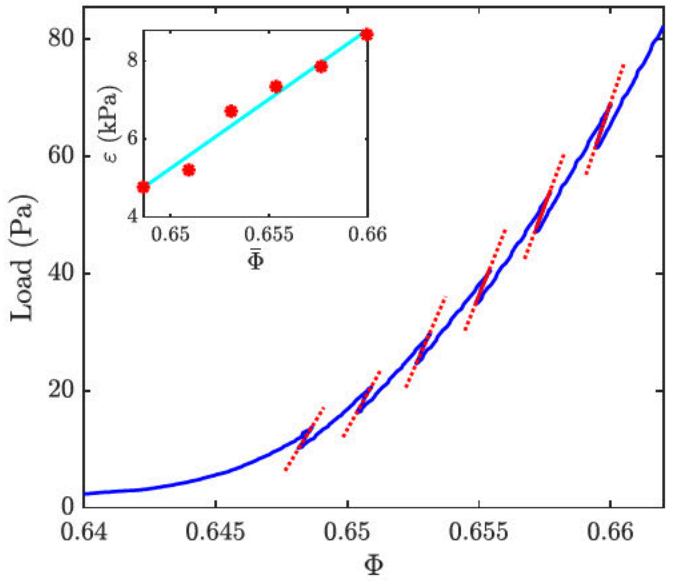


Figure 15: The load plot of quasi-steady loading and unloading carried out on hydrogel batch 6 at 0.1 mm/s to determine $\varepsilon(\Phi)$. The dotted red lines are straight line fits to the unloading sections. The inset shows the values of ε calculated from these fits (red stars), with a straight line fitted (cyan).

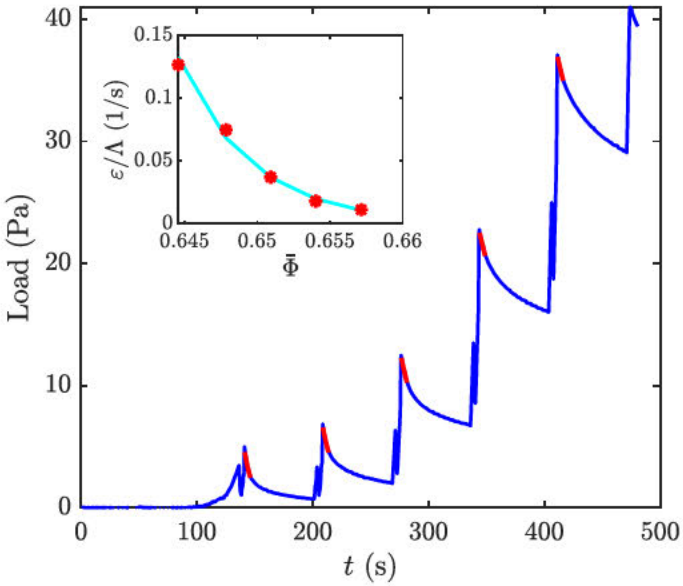


Figure 16: The load plot of a loading and unloading test with step changes carried out on hydrogel batch 6 to determine $\varepsilon(\Phi)/\Lambda(\Phi)$. The red lines are exponential fits to relaxations sections after the step changes. The inset shows the values of ε/Λ calculated from these fits (red stars), with a straight line fitted (cyan).

plunger in place and observe how the load responds over time. By Taylor expanding about the state before the step change, the small change in load ΔP in response to the small change in height Δh behaves like

$$\Delta P \sim \exp\left(-\frac{\varepsilon(\Phi_*)}{\Lambda(\Phi_*)}t\right), \tag{63}$$

where Φ_* is the solid fraction before the small step change. Hence, provided the bulk elastic modulus $\varepsilon(\Phi)$ has already be found, the bulk viscosity $\Lambda(\Phi)$ can be calculated from the exponential decay rate in response to step changes at different values of solid fraction.

Figure 16 shows an example experiment, where a small quasi-steady loading and unloading was made before each step change. The step change was a 0.4 mm/s and all other motion was at 0.1 mm/s. In this example, fitting the exponential decay rate of the relaxation gives, roughly, $\epsilon/\Lambda = e^{-59.4}\Phi^{-130.7}$ s^{-1}. Note that we choose not to use speeds greater than 0.4 mm/s because simple speed tests using only water (no hydrogels) showed that large overshoots occurred in the load readings after moving the plunger too quickly.

It is worth noting that all our loading and unloading tests are done over a very small range of solid fractions (close to the gel point) and with few data points, and only one repeat of each tests was considered here, so the functional forms of ε and Λ we get from our analysis are quite approximate. To test how good our estimates are, we could solve the constitutive relation (56) for a given loading and unloading protocol with the forms of $P_y(\Phi)$, $\varepsilon(\Phi)$ and $\Lambda(\Phi)$ that we have found, to see if the solution agrees with the experimental load. This is future work.

4 Conclusion

This work has been centered around the use of a pack of hydrogels saturated in water as a model for a two-phase deformable porous media, and how this can help form a link between observations, experiments and theory.

Firstly, we studied the observed finite-time decay of water waves under a layer of floating particles. We carried out experiments using hydrogels as the floating particles (based on experiments by [16]) and reproduced the finite-time decay rate found by [16]. Informed by these experimental results, we developed a theoretical framework to try and capture the finite time wave damping, modelling the floating layer as a two-phase deformable porous material and taking the membrane limit. It is hoped that this framework will enable us to write down an energy equation that will demonstrate the finite-time decay found in experiments and allow us to attribute this behaviour to certain physical dissipative processes.

In the second part of this work, we carried out one-dimensional compression tests on a saturated pack of hydrogels, motivated by assessing the applicability of the one-dimensional elastoviscoplastic constitutive relation proposed by [14]. Our aim was to use appropriate loading and unloading protocols to fit the parametric functions in the elastoviscoplastic model. However, when carrying out simple compression tests to assess reproducibility, we found that the load seems to decrease across repeated compressions, suggesting that the hydrogels are being irreversibly damaged by the load. It was unclear whether or not lower loads led to the same affect. Either way, it makes it difficult to say anything concrete about the properties of the hydrogel pack, given that the tests used to determine the properties seem to cause the properties themselves to change.

5 Acknowledgements

I would like to say a big thank you to Neil Balmforth for teaching me such a lot over the summer, giving me so much of his time, and for making working on this project so enjoyable. I would also like to thank Jim McElwaine for painstakingly writing the software for the experiments, Tom Eaves for helping with the theory, and Anders Jensen for being able to help with anything and everything in the lab. Finally, thank you to the directors Stefan and Colm for organising the GFD program, to the principal lecturers Laure and Peter for sharing their knowledge with us, and to the other fellows for providing great company.

References

[1] B. Applications and D. Seliktar, *Designing Cell-Compatible Hydrogels for*, Tech. Rep. 6085, 2012.

[2] T. Bertrand, J. Peixinho, S. Mukhopadhyay, and C. W. MacMinn, *Dynamics of Swelling and Drying in a Spherical Gel*, Physical Review Applied, 6 (2016).

[3] M. A. Biot, *General theory of three-dimensional consolidation*, Journal of Applied Physics, 12 (1941), pp. 155–164.

[4] D. A. Drew and S. L. Passman, *Theory of Multicomponent Fluids*, Springer-Verlag Berlin Heidelberg GmbH, 1999.

[5] A. K. Gaharwar, N. A. Peppas, and A. Khademhosseini, *Nanocomposite Hydrogels for Biomedical Applications*, Biotechnol. Bioeng, 111 (2014), pp. 441–453.

[6] D. R. Hewitt, J. S. Nijjer, M. Grae Worster, and J. A. Neufeld, *Flow-induced compaction of a deformable porous medium*, tech. rep., 2016.

[7] A. L. Kohout, M. J. Williams, S. M. Dean, and M. H. Meylan, *Storm-induced sea-ice breakup and the implications for ice extent*, Nature, 509 (2014), pp. 604–607.

[8] H. J. Kong, E. Alsberg, D. Kaigler, K. Y. Lee, and D. J. Mooney, *Controlling degradation of hydrogels via the size of cross-linked junctions*, Advanced Materials, 16 (2004), pp. 1917–1921.

[9] C. W. MacMinn, E. R. Dufresne, and J. S. Wettlaufer, *Fluid-driven deformation of a soft granular material*, Physical Review X, 5 (2015).

[10] C. C. Mei, M. Krotov, Z. Huang, and A. Huhe, *Short and long waves over a muddy seabed*, Journal of Fluid Mechanics, 643 (2010), pp. 33–58.

[11] T. K. Meyvis, S. C. De Smedt, J. Demeester, and W. E. Hennink, *Influence of the degradation mechanism of hydrogels on their elastic and swelling properties during degradation*, Macromolecules, 33 (2000), pp. 4717–4725.

[12] K. H. Parker, R. V. Mehta, and C. G. Caro, *Steady Flow in Porous, Elastically Deformable Materials*, tech. rep., 1987.

[13] D. T. Paterson, T. S. Eaves, D. R. Hewitt, N. J. Balmforth, and D. M. Martinez, *Flow-driven compaction of a fibrous porous medium*, Physical Review Fluids, 4 (2019).

[14] ——, *One-dimensional compression of a saturated elastoviscoplastic medium*, Physical Review Fluids, 7 (2022).

[15] V. A. Squire, *A fresh look at how ocean waves and sea ice interact*, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 376 (2018).

[16] B. R. Sutherland and N. J. Balmforth, *Damping of surface waves by floating particles*, Physical Review Fluids, 4 (2019).

[17] T. Yamamoto, H. L. Koning, H. Sellmeijer, and E. V. Van Hijum, *On the response of a poro-elastic bed to water waves*, Journal of Fluid Mechanics, 87 (1978), pp. 193–206.

Understanding Weakly Nonlinear Wave Interaction Using Dynamic Mode Decomposition

Claire Valva

Koopman operator formalism allows the linear dissection of nonlinear dynamical systems into coherent structures with characteristic evolution frequencies via the translation from state variables to an embedding of the state variable into an infinite space of observables [4]. Dynamic mode decomposition (DMD) provides approximate information about the spectrum of the Koopman operator via a straightforward algorithm (A.1). Similar to the decomposition that Koopman operators provide, DMD is a modal decomposition, where a high dimensional spatiotemporal signal is decomposed into dynamic elements: corresponding spatial modes, scalar amplitudes, and temporal signals that can be recovered with linear superposition.

Uses of DMD and related algorithms [20, 10, 19, 21] have been very sucessful in analysing both experimental and observational data. The algorithm has been used to significant success in a variety of applications including experimental fluid dynamics and climate science [22, 12, 9]. DMD is particularly effective in the analysis of flows where the evolution of observables is controlled by a small number of dynamic processes.

To our knowledge, weakly nonlinearly interacting waves have not yet been studied with DMD. We anticipate that DMD will be an effective way to analyze nonlinear wave interaction, particularly when we are interested in the interaction between a small number of waves. Linear wave interaction can be completely explained by DMD due to the construction of the algorithm; by necessity, nonlinear wave interaction will be harder to understand, particularly if the interaction is between waves of incommensurate frequencies. It will be useful to detect and quantify nonlinear interaction from the results of dynamical mode decomposition, but translating these qualities from linear dynamic components is nontrivial.

In this study, we will discuss a framework to analyze weakly nonlinear waves with DMD using the Korteweg-de Vries (KdV) equation as a simple test case. We will give an overview on the relevant equations and algorithms (section 1); discuss theoretical expectations and difficulties for interpreting DMD results (section 2); test these ideas on numerical simulations of KdV with varying amounts of expected nonlinearity (section 3); and discuss avenues to expand and improve upon these ideas (section 4).

1 Background

1.1 Dynamic Mode Decomposition (DMD)

Given some dynamical system $\{\mathbf{v}_j\}_j$, DMD seeks to find a linear function A such that $\mathbf{v}_{j+1} = A\mathbf{v}_j$. In practice, DMD finds an eigendecomposition of A in which each mode of the decomposition can be converted into scalar amplitudes a_k, temporal patterns $\mathbf{b}_k$ and spatial patterns $\mathbf{u}_k$. The temporal patterns $\mathbf{b}_j$ of DMD are of the form $\mathbf{b}_j = e^{t(i\omega_j + b_j)}$ from which we can extract some growth rate b_j and an eigenfrequency ω_j. In the following we will write λ_j to be equal to the log of $\mathbf{b}_j$ divided by t, i.e. $\lambda_j = b_j + i\omega_j$. Additionally, we can determine a spatial wavenumber ℓ_k associated with each mode $\mathbf{u}_k$. Then if $\mathbf{v}_j = \mathbf{v}(j\tau)$ for time resolution τ, we can reconstruct the evolution of a given dynamical system $\{\mathbf{v}_j\}_j$ as:

$$\mathbf{v}_m = \sum_k e^{(i\omega_k + b_j)m\tau}\mathbf{u}_k$$

In the algorithm's most basic form (and usual use) the number of DMD modes that come from the spectral decomposition will be equal to the spatial dimension. We can increase both the robustness and number of discovered of DMD modes found with delay embedding, a higher-order extension in which temporal resolution is substituted for spatial resolution [21]. (For a more in-depth discussion on computation and algorithms see A.1 and [1, 4].)

Eigenvalue lattices As DMD is a method which approximates the spectral properties of the Koopman operator, we expect the resolved eigenfrequencies of DMD to quasi-inherit the generation of a lattice of eigenvalues by the Koopman operator. This will ultimately result in some of the primary challenges of understanding nonlinear dynamical systems in DMD, which will be discussed more in 2.1.

The Koopman operator is an infinite-dimensional linear operator K^t which acts linearly on functions such that:

$$K^t f(\mathbf{v}_s) = f(\mathbf{v}_{s+t}) \tag{1}$$

Suppose that K^t has eigenfrequencies λ_1, λ_2 eigenfrequencies associated with eigenfunctions g_1, g_2 of the Koopman operator, then:

$$K^t g_1 g_2 = e^{i(\lambda_1+\lambda_2)t} g_1 g_2 \tag{2}$$

So $\lambda_1+\lambda_2$ will also be an eigenfrequency of K^t, as will the lattice of every other integer linear combinations of λ_1 and λ_2. (An additional difficulty is that if λ_1 and λ_2 are incommensurate, this lattice will be dense in $\mathbb{R}$.)

1.2 Korteweg-de Vries equation

The KdV equation describes the weakly nonlinear evolution of long, unidirectional surface waves in shallow water:

$$\eta_t + c\eta_x + \frac{3c}{2h}\eta\eta_x + \frac{3h^2}{6}\eta_{xxx} = 0,\ c = \sqrt{gh} \tag{3}$$

This equation (and variations thereof) have been of significant interest in geophysical fluid dynamics. KdV has been used to study surface water waves, internal waves, and equatorial Rossby waves among others [13, 15, 3].

Aside from GFD, KdV has been of broad mathematical interest since vibrating string experiments in 1955 by Fermi and colleagues [8]. Solutions of KdV decompose at large times into collections of solitons (well-separated solitary waves) and has N-torus families of solutions that are exactly periodic in space but "almost periodic" in time [23, 11, 15, 6].

The equation can also be rescaled to its more canonical form, where the nonlinear and dispersive terms more directly balance. From this formulation it is also easy to see in the low amplitude case (linear limit), the KdV equations have dispersion relation $\omega = -k^3$.

$$u_t + u_{xxx} + 6uu_x = 0 \tag{4}$$

It is notable that KdV is exactly integrable using the inverse scattering transform (IST), and that this transform can be used to find the analytical Koopman eigenfunctions for KdV (at least in the unbounded domain case) [14, 17]. We can use IST to write down solutions $u(x,t)$ of KdV (called cnoidal waves) in terms of a Riemann theta function $\theta(x,t)$:

$$u(x,t) = 2\partial_{xx} \ln(\theta(x,t)) \tag{5}$$

Riemann Theta Functions Riemann theta functions are a reduction of generalized Fourier series that are often used in the construction of various equations in mathematical physics and fundamental to IST [14].

In this work, we will primarily use θ function to provide a useful skeleton in which to understand wave interaction. For a system with N degrees of freedom (i.e. wave components), we can write the θ function as follows:

$$\theta(x,t) = \sum_{m \in \mathbb{Z}^N} e^{i(\mathbf{m}\cdot\mathbf{u})} e^{\frac{1}{2}\mathbf{m}\cdot B\mathbf{m}}, \ \mathbf{u}_j(x,t) = k_j - \omega_j t + \phi_j \tag{6}$$

Following the usual notation, k_j is the jth spatial wavenumber, ω_j is the frequency, and ϕ_j is the phase of the wave component. B is an $M \times M$ symmetric matrix that quantifies the interaction of each wave component. The diagonal entries $B_{j,j}$ parameterizes the amplitude of each individual wave component, while $B_{i,j}$ parameterizes the amplitude between each component i and j. We can then view the diagonal coefficients $B_{i,i}$ as modulating the importance of linear wave interaction, while the off-diagonal coefficients denote the relative importance of nonlinear wave interaction between the two components.

2 Expected DMD Interpretation and Anticipated Challenges

From given nonlinear wave data, we would like to be able to use DMD to decompose the system into a relatively small number of fundamental interactions. For the following discussion, we assume that the dynamical system that we are analyzing is due to the (not necessarily linear) interaction of a small number of waves. We will also assume that the DMD modes will have purely imaginary eigenfrequencies (λ_k is purely imaginary), i.e. each mode will not grow or decay, which is an assumption that would be consistent with a

conserved quantity of some kind. We would like to be able to write down the characteristic frequencies and spatial wavenumbers of the system as well as detect the relative impacts of linear and nonlinear interaction. We will discuss methods to achieve two primary goals: characteristic frequency estimation, then the use of these frequencies combined with DMD results to write down the terms of the θ function for KdV.

2.1 Frequency Estimation

We will first consider an extremely simple dynamical system (7), which characteristic frequencies ω_s and ω_f with M_s and M_f respective harmonics.

$$\begin{aligned} x &= \sum_{j=1}^{M_s}(\frac{a}{j!}\cos(j\cdot\omega_s t))(1+\sum_{k=1}^{M_f}(\frac{A}{k!}(\cos(k\cdot\omega_f t))) \\ y &= \sum_{j=1}^{M_s}(\frac{a}{j!}\cos(j\cdot\omega_s t))(1+\sum_{k=1}^{M_f}(\frac{A}{k!}(\sin(k\cdot\omega_f t))) \\ z &= \sum_{j=1}^{M_s}(\frac{a}{j!}\sin(j\cdot\omega_s t)) \end{aligned} \tag{7}$$

If $M_s = M_f = 1$, we can easily rewrite the above (with the use of a few trig) to get a linear system with frequencies $\omega_s + \omega_f$, $\omega_s - \omega_f$, and ω_s. Give sufficient data, DMD should also capture all 3 frequencies. Similarly, for larger M_s and M_f we get many more integer linear combinations of ω_s and ω_f that we expect to be captured by DMD[1]. Despite the many eigenvalues of the system, it is parameterized by two fundamental frequencies ω_s and ω_f that generate a lattice of eigenvalues. *We will seek ways to identify the two (or more) frequencies that generate an eigenvalue lattice given by DMD results.*

2.1.1 Differentiating eigenfrequencies when $N = 2$

We will discuss two methods to identify the generating frequencies from DMD results when we think there are two generating frequencies: the first of which uses the "eigenfrequency island" pattern of the eigenvalue lattice (which occurs when the number of resolved eigenfrequencies is relatively small) and the second of which poses an iterative optimization problem that is partially with an integer minimization algorithm called PSLQ. Ideally, not only do we want to identify ω_s and ω_f, but for every sufficiently well-resolved λ_k which is an eigenfrequency output of DMD, we would like to find M_f and M_s such that $\lambda_k = M_f\omega_f + M_s\omega_s$.

(It is significantly more trivial to identify the wanted frequency when $N = 1$. One possibility to estimate the wanted frequency is to take the difference of eigenvalues from one another. This idea and uses of it are discussed in depth for finding periodic orbits in turbulent flows in [16]).

[1]This discussion should be reminiscent of the eigenvalue lattice that is generated by the Koopman operator discussed in 1.1.

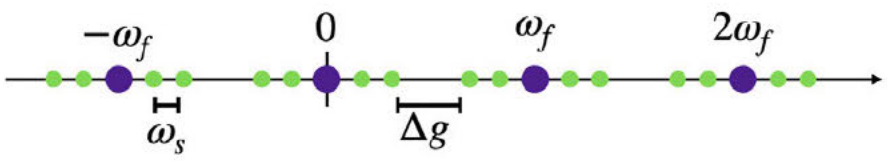


Figure 1: Schematic of "eigenvalue islands" that visually form given DMD results from a system with two generating eigenfrequencies: ω_s and ω_f where ω_s is sufficiently smaller than ω_f. The gap between islands is given by $\Delta g = \omega_f - 2(M+1)\omega_s$ where $M = 2$ is the number of resolved harmonics of ω_s.

Eigenfrequency islands Suppose — as in the sample dynamical system above — that the system we intend to analyze is parameterized by two frequencies ω_s and ω_f, where ω_s is sufficiently less than ω_f. Then the results from DMD will form visual islands (see Figure 1) of eigenfrequencies that are separated by a gap Δg that can be relatively easily identified by eye[2]. The mean distance between the center of each island will be approximately ω_f, while the mean difference between eigenvalues within each island will be ω_s. One can then similarly find the values M_f and M_s by counting within islands.

While this is a relatively simple idea to implement, there is a restriction both upon the relative value of ω_f and ω_s as well as the need for DMD eigenfrequencies to be fairly well resolved so that the mean distance between eigenfrequencies can be relied upon as an accurate estimate.

PSLQ iteration procedure The second method that we propose to identify eigenvalues is a 2-step minimization procedure that iterates between (1)finding the integers M_s and M_f for every λ_k such that $\lambda_k \approx M_s\omega_s + M_f\omega_f$. and (2) a minimization problem to guess the frequencies ω_s, ω_f.

The PSLQ algorithm — see [7, 2] — is an integer relation algorithm that given a vector of complex numbers $x = (x_1, x_2, \ldots, x_n)$ can find integers a_i which are not identically zero such that:

$$a_1x_2 + a_2x_2 + \ldots a_nx_n = 0 \tag{8}$$

We write λ_k to denote some eigenfrequency computed from DMD of which total L, ω_s and ω_f to denote current guesses for frequencies, and $M_{s,k}$ and $M_{f,k}$ to denote integer multiples. We propose the following two-step algorithm:

1: Guess ω_s, ω_f.
2: Choose error tolerance ϵL and define loss as $\mathcal{L} = \sum_k^L \|M_{s,k}\omega_s + M_{f,k}\omega_f - \lambda_k\|$.
3: **while** $\mathcal{L} \geq \epsilon L$ **do**
4: Use PSLQ with $x_k = (\omega_s, \omega_f, \lambda_k)$, to find best $M_{f,k}, M_{s,k}, A$ so that $M_{s,k}\omega_s + M_{f,k}\omega_f + A\lambda_k = 0$. (By design, should get $A = -1$.)
5: Pick new ω_s, ω_f by minimizing $\mathcal{L}$.
6: **end while**

[2]The formula for the gap between "islands" is given by $\Delta g = \omega_f - 2(M+1)\omega_s$ where M is the number of resolved harmonics of ω_s. For visual discernment, we must have that $\Delta g > \omega_s$, i.e. that $\omega_f > (2M+3)\omega_s$.

Unlike eigenfrequency identification via visual islands, this method should prove to work in general. However, it requires the implementation of a few optimization algorithms which can prove somewhat tricky to work with. As such, in 3 we will use the "island" method for ease.

2.2 Measuring relative amplitude and interaction via the θ function

The θ function formulation for solutions to the KdV equations allows us to nicely quanitfy the relative importance of linear and nonlinear interaction by comparing the relative magnitude of the coefficients of the interaction matrix B.

θ function formulation when $N = 2$ Given somewhat well-resolved DMD for given KdV data (suitably transformed as in 3) one can write down the parameters of the θ function. We will first consider data that we expect to be well-represented by a θ function with 2 degrees of freedom.

$$\theta(x,t) = \sum_{m_1,m_2\in\mathbb{Z}} e^{i(k_1m_1+k_2m_2)x-(\omega_1m_1+\omega_2m_2)t} e^{\frac{1}{2}m_1^2B_{11}+2m_1m_2B_{22}+m_2^2B_{22}} \tag{9}$$

In the following, we would like to recover the 2 expected wavenumber frequency pairs (i.e. k_j and ω_j pair) as well as the coefficients of interaction matrix: B_{11}, $B_{12} = B_{21}$, and B_{22} from given DMD data.

Recall that for each DMD mode, there is an associated frequency λ_j and spatial pattern $\mathbf{u}_j$ from which we can extract a spatial wavenumber ℓ_j. We expect that each λ_j is an integer linear combination of the frequencies ω_1 and ω_2, and that similarly, each corresponding spatial pattern should have a wavenumber ℓ_j that corresponds to an integer linear combination of the two characteristic wavenumbers k_1, k_2, i.e., we have:

$$\lambda_j = n_{1,j}\omega_1 + n_{2,j}\omega_2,\ \ell_j = n_{1,j}k_2 + n_{2,j}k_2$$

To solve for the matrix B (as well as the other wanted parameters) we will need 3 dynamical modes that come from DMD[3], which will include the eigenfrequencies λ_j, spatial wavenumbers ℓ_j, and amplitudes a_j. To solve for B, we can construct an invertible matrix M where each row M_j has entries $(n_{1,j}^2, 2n_{1,j}n_{2,j}, n_{2,j}^2)$. Then if we let $\mathbf{c}$ be a vector such that $c_j = 2\ln(a_j)$, we can solve the following system for B:

$$M\mathbf{b} = \mathbf{c},\ \mathbf{b} = (B_{1,1}, B_{2,1}, B_{2,2}) \tag{10}$$

θ function formulation for $N > 2$ The previous procedure to find the coefficients of B is not unique to $N = 2$ degrees of freedom. For $N > 2$, we can repeat the same process but must instead have $(N^2-N)/2+N$ well-resolved DMD modes from which we can construct a linearly independent set S to construct M of the following form:

$$S = \{(n_{1,j}^2, \ldots, n_{N,j}^2, 2n_{1,j}n_{2,j}, \cdots n_{N-1,j}n_{N,j})|\ j \le (N^2-N)/2+N\} \tag{11}$$

[3] If we know $n_{1,j}, n_{2,j}$, we can solve for both k_i and ω_i from the overdetermined system from the three ℓ_j and λ_j respectively. Alternatively, these can also be estimated using ideas discussed in 2.1.

3 In Practice: Numerical Experiments with the KdV equations

In the following, we analyze numerical simulations of KdV with a variety of initial conditions: we vary the number of initial wave modes as well as the amplitude of the initial condition —a proxy for the expected nonlinearity of the system.

Method We numerically integrate the KdV equations with psuedospectral methods using the `Dedalus v2` framework [5]. We use the canonical form of KdV (eq 4) on the periodic domain $[0, 2\pi]$ represented in a 128 point Fourier basis.

In all of the following numerical results, we perform DMD upon the θ function computed from the numerical results. To convert $u(x,t)$ to the theta function form, we take the scalar

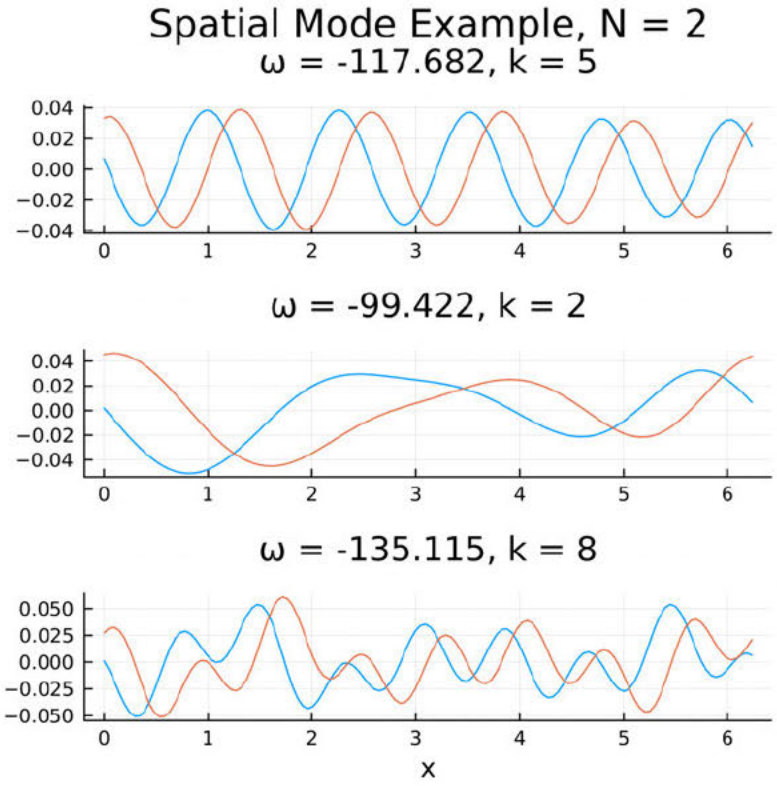


Figure 2: Example of chosen modes to determine the coefficients of the interaction matrix B for initial condition $u(x,0) = -4\cos(3x) - 5\cos(5x)$. We plot the real and complex parts of the spatial mode $\mathbf{u}_k$, where k is the dominant spatial wavenumber chosen by a Fourier transform, and ω is the corresponding temporal frequency. (top) Mode corresponding the $-117.682 = 0\omega_1 + 1\omega_2$, where ω_2 is the frequency induced by the $\cos(5x)$ wave of the initial condition. (middle) Mode corresponding the $-99.422 = -1\omega_1 + 1\omega_2$, which has wavenumber $2 = -3 + 5$. (bottom) Mode corresponding the $-135.115 = 1\omega_1 + 1\omega_2$, which has wavenumber $8 = 3 + 5$.

exponential of u and then apply the inverse Laplacian operator in Fourier space. We process 50 time units of data with the DMD algorithm and the number of required delays vary with the initial conditions (generally more delays in the case where initial conditions have larger amplitudes).

In order to choose the well-resolved DMD modes needed to compute the coefficients of DMD, we use the dispersion relation for linearized KdV ($\omega = -k^3$) and the "eigenfrequency island" method to inform initial guesses as to the generating frequencies and the integer coefficients between each mode. We found that the computation of eigenfrequencies, corre-sponding spatial wavenumbers, and coefficients of B are fairly robust to the choice of modes (i.e. the λ_j, k_j, and a_j for each mode) required to compute B and as such, we will only discuss results from one selection of modes for each initial condition (1 mode when $N = 1$, 3 modes when $N = 2$). A sample selection of 3 well-resolved modes is shown in figure 2, where both the wavenumbers and frequencies are reasonable given the linear dispersion relation. Also, the computed frequencies and wavenumbers correspond the the same integer linear combinations of the generating frequencies and wavenumbers.

We compare the theta function reconstruction and original KdV time series by com-puting various statistics (mean, variance, maximum, and minimum) using a series length corresponding to 50 time units with the same temporal spacing. The θ function is repre-sented as a partial series where the summation (the range of $\mathbf{m}$ in 6) goes up to 8 terms, in which the summation of more terms will have a magnitude less than machine precision.

3.1 Data reconstruction from DMD when $N = 1$

We first attempt to reconstruct numerical solutions of KdV using a θ function representation with $N = 1$ degrees of freedom for initial conditions with varying amplitudes. In all of these experiments, the initial conditions are of the form $u(x, 0) = A\cos(3x)$ for some amplitude A. When A is very small, we expect solutions to behave as a traveling wave that satisfies $u_t - u_{xxx} = 0$, and as such, we expect the dispersion relation to behave as in the linear limit with a frequency of $3^3 = 27$ and a spatial wavenumber of 3. When A becomes larger the linear theory will no longer be accurate. As A grows significantly ($\sim \mathcal{O}(1)$), we would expect that that the theta function for one degree of freedom will no longer lead to a good approximation of the system.

As expected, when the amplitude of the initial condition is very small ($A = -4e - 5$), the linear limit appears to apply quite well. The estimated frequency of the system was $\omega = 26.9999 \approx 27$ and the spatial mode was wavenumber 3. Additionally, the measured statistics of the original and reconstructed data are near identical (see table 1).

When the amplitude is increased, the linear limit becomes a less and less accurate guess. The increasing amplitude corresponded with a slowing of the flow, where the frequency that was $\omega = 27$ when $A = -4e - 5$ slowing to $\omega < 20$ for A of order 1. Figure 3 shows both a slowing down as well as DMD resolving additional harmonics of the generating frequency. When $A = -4e - 5$, the solution can be nearly completely explained by a single traveling wave, leading to one dominant DMD eigenfrequency in contrast to the larger amplitude cases. Here, the solution can no longer be explained by a single traveling wave and the interaction weakly nonlinear interaction between waves requires more DMD modes to be accurately explained.

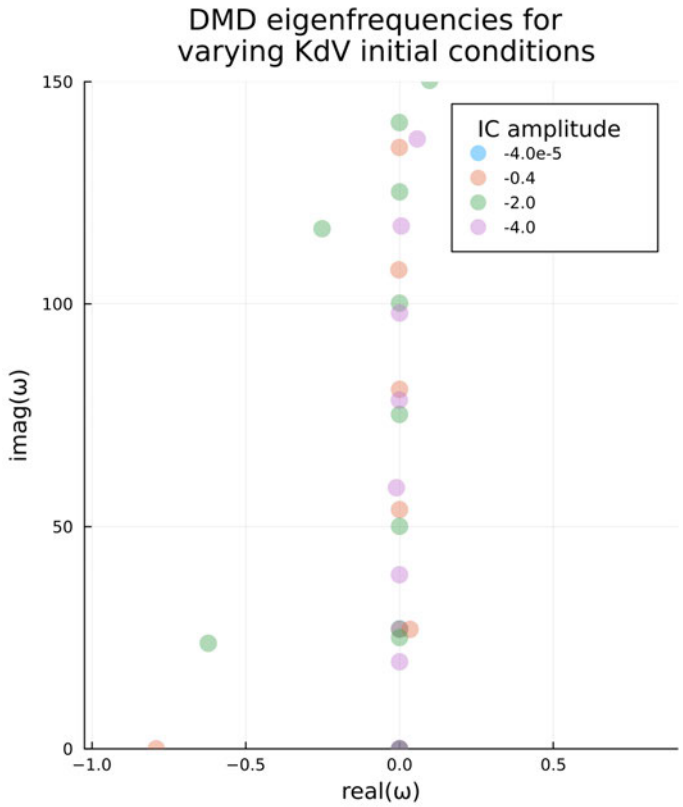


Figure 3: Comparison of eigenfrequencies obtained from DMD for KdV data with initial condition of the form $u(x,0) = A\cos(3x)$ where the amplitude A varies. eigenfrequencies less than 150 and real part with magnitude less than 1 are plotted. All eigenfrequencies obtained from DMD analysis for 50 time units and 5 delays. With increasing amplitude, the generating frequency slows from the linear limit of 27 oscillations per time unit to approximately 20 oscillations per time unit.

We find that the one degree of freedom approximation performs fairly well for an amplitude $A = -4e-1$, where the reconstructed and original data have nearly identical variances and extrema that barely differ (table 1) and are visually indistinguishable (figure 4). However, when $A \sim \mathcal{O}(1)$, the approximation has both less variance than the original data and does not achieve the same maxima. Visually this difference is also evident as some of the fine scale details of the original solution are missing (figure 5), however the reconstruction maintains the large scale attributes of the solution in both the dominant spatial wavenumber and frequency.

Ultimately, in this case, using DMD to reconstruct solutions of KdV for an initial condition with one wave mode is fairly successful method. The reconstructed solutions of the theta function form are near perfect for flows of small amplitudes. For larger amplitudes, the theta function captures the dominant features but misses small scale details. It is likely that a theta function approximation with $N > 1$ could improve the accuracy of reconstructions with larger amplitudes.

3.2 Data reconstruction and wave interaction from DMD when $N = 2$

We can repeat the above analysis for solutions of KdV with initial conditions of the form $u(x,0) = A_1\cos(3x) + A_2\cos(5x)$. In this case, we will reconstruct these solutions with a theta function with $N = 2$ degrees of freedom. We anticipate some of the same patterns as in the previous discussion: small A_1, A_2 will correspond with the propagation of two traveling waves and the dispersion relation $\omega = -k^3$ (we will see $\omega = 27, 125$ and $k = 3, 5$); the flow slowing as amplitudes increase; as well as the reconstructed decaying in accuracy as the amplitudes of the initial conditions increase.

One can reference table 2 to see that the same trends from the $N = 1$ case have carried to the $N = 2$ case. When A_1 and A_2 are very small, the expected frequencies from the linearized KdV dispersion relation match the frequencies identified by DMD and the reconstructed solution captures the original solution well (figure 6). As the amplitude A_1 and A_2 of each wave grows, the generating frequency corresponding to that wave decreases.

Additionally, as in the $N = 1$ case, when the amplitudes grow to about order 1, the reconstructed data series no longer is able to completely capture the variance or the extrema of the solution. The statistics as well as the visual comparison of these solutions appears more evident than in the previous discussion. Figures 7 and 8 both show results where either A_1 or A_2 are large. For both cases, we are unable to fully resolve all of the large scale details, even if the dominant two wavenumber and frequency pairs appear to be consistent between the original and constructed data.

initial condition		estimated ω		mean	variance	max	min
-4.00e+00cos(3x)	original data		θ	1.01e+00	2.28e-02	1.25e+00	7.97e-01
			u	4.15e-15	7.98e+00	5.60e+00	-3.96e+00
	reconstructed	19.5873	θ	1.00e+00	2.27e-02	1.21e+00	7.87e-01
			u	-3.33e-19	7.85e+00	4.84e+00	-3.17e+00
-2.00e+00cos(3x)	original data		θ	1.00e+00	6.03e-03	1.12e+00	8.92e-01
			u	3.57e-17	2.00e+00	2.43e+00	-1.99e+00
	reconstructed	25.0412	θ	1.00e+00	6.03e-03	1.11e+00	8.90e-01
			u	-5.00e-20	1.99e+00	2.22e+00	-1.78e+00
-4.00e-01cos(3x)	original data		θ	1.00e+00	2.47e-04	1.02e+00	9.78e-01
			u	1.54e-16	8.00e-02	4.18e-01	-4.00e-01
	reconstructed	26.9200	θ	1.00e+00	2.47e-04	1.02e+00	9.78e-01
			u	-2.53e-20	8.00e-02	4.09e-01	-3.91e-01
-4.00e-05cos(3x)	original data		θ	1.00e+00	2.47e-12	1.00e+00	1.00e+00
			u	-5.73e-20	8.00e-10	4.00e-05	-4.00e-05
	reconstructed	26.9999	θ	1.00e+00	2.47e-12	1.00e+00	1.00e+00
			u	3.90e-24	8.00e-10	4.00e-05	-4.00e-05

Table 1: Comparison of numerical simulation of KdV and reconstruction via θ function with initial conditions of the form $u(x,0) = A\cos(3x)$. As the amplitude A of the initial condition increases, the generating frequency decreases (column labeled estimated ω) and the reconstructed data performs worse, indicating that as the nonlinearity of the solution increases that $N = 1$ degrees of freedom is no longer a good assumption.

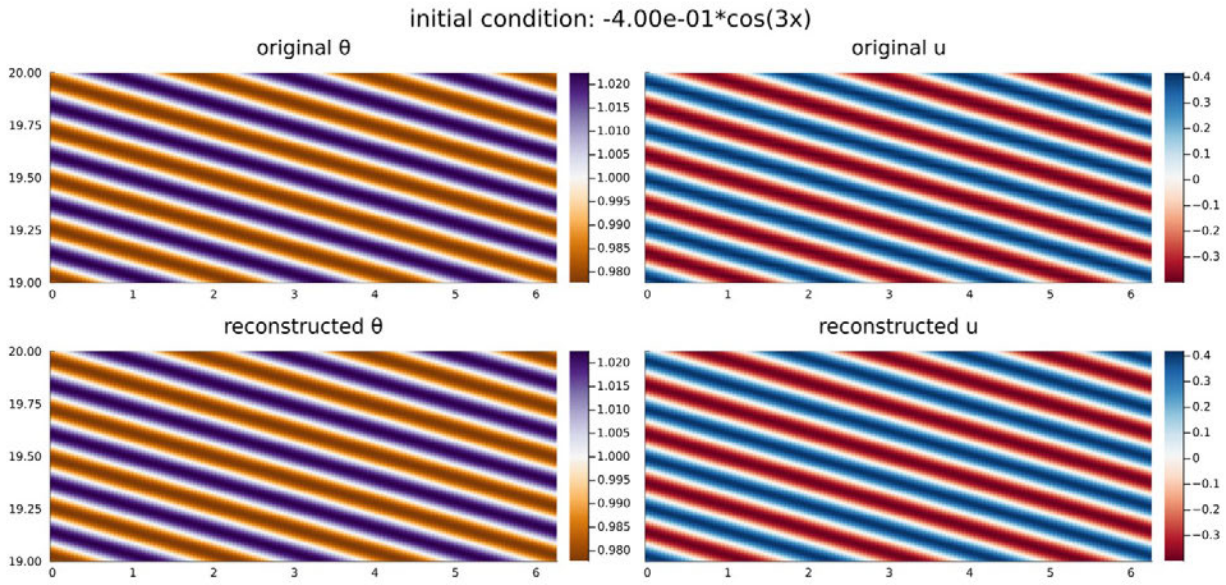


Figure 4: Comparison of original and reconstructed KdV using the θ function reconstruction for initial condition $u(x,0) = -0.4\cos(3x)$.

initial condition		est. ω_1	est. ω_2	$B_{1,1}$	$B_{1,2}$	$B_{2,2}$		mean	var	max	min
-4.00e+00*cos(3x)-5.00e+00*cos(5x)	original data						θ	1.15e+00	4.59e-01	5.00e+00	3.13e-01
							u	3.60e-15	1.97e+01	1.23e+01	-8.96e+00
	reconstructed	17.846	117.406	-5.07161	-2.02409	-5.57401	θ	1.00e+00	2.29e-02	1.36e+00	7.93e-01
							u	4.16e-19	2.28e+01	1.04e+01	-7.21e+00
-4.00e+00*cos(3x)-5.00e-01*cos(5x)	original data						θ	1.01e+00	2.57e-02	1.44e+00	7.41e-01
							u	-5.43e-15	8.10e+00	6.21e+00	-4.58e+00
	reconstructed	19.562	124.323	-4.47107	-2.51939	-10.5794	θ	1.00e+00	2.30e-02	1.24e+00	7.83e-01
							u	2.78e-20	8.04e+00	5.34e+00	-3.63e+00
-4.00e-01*cos(3x)-5.00e-01*cos(5x)	original data						θ	1.00e+00	3.00e-04	1.04e+00	9.68e-01
							u	1.14e-16	2.05e-01	9.41e-01	-9.14e-01
	reconstructed	26.920	124.919	-8.9999	-2.77186	-10.5993	θ	1.00e+00	2.98e-04	1.03e+00	9.70e-01
							u	-1.39e-20	2.04e-01	9.12e-01	-8.84e-01
-4.00e-01*cos(3x)-5.00e-05*cos(5x)	original data						θ	1.00e+00	2.47e-04	1.02e+00	9.78e-01
							u	-1.02e-16	8.00e-02	4.18e-01	-4.00e-01
	reconstructed	26.920	124.993	-9.00062	-2.77019	-29.0206	θ	1.00e+00	2.47e-04	1.02e+00	9.78e-01
							u	3.78e-20	8.00e-02	4.09e-01	-3.91e-01
-4.00e-05*cos(3x)-5.00e-05*cos(5x)	original data						θ	1.00e+00	2.97e-12	1.00e+00	1.00e+00
							u	-1.05e-19	2.05e-09	8.99e-05	-8.99e-05
	reconstructed	27.000	124.994	-27.4203	-2.77051	-29.0194	θ	1.00e+00	2.97e-12	1.00e+00	1.00e+00
							u	-1.02e-24	2.05e-09	8.99e-05	-8.99e-05

Table 2: Comparison of numerical simulation of KdV and reconstruction via θ function with an initial condition of the form $u(x,0) = A_1\cos(3x) + A_2\cos(5x)$. As the amplitude of the initial condition increases, the generating frequency decreases (column labeled estimated ω_i) and the reconstructed data performs worse. This indicates that with increasing amounts of nonlinear interaction the theta function approximation with 2 degrees of freedom is no longer valid. Note also that as the magnitude of both initial conditions increases, the relative influence of nonlinear interaction increases, which we can see as the relative value of $B_{1,2}$ increases in comparison to $B_{1,1}$ and $B_{2,2}$.

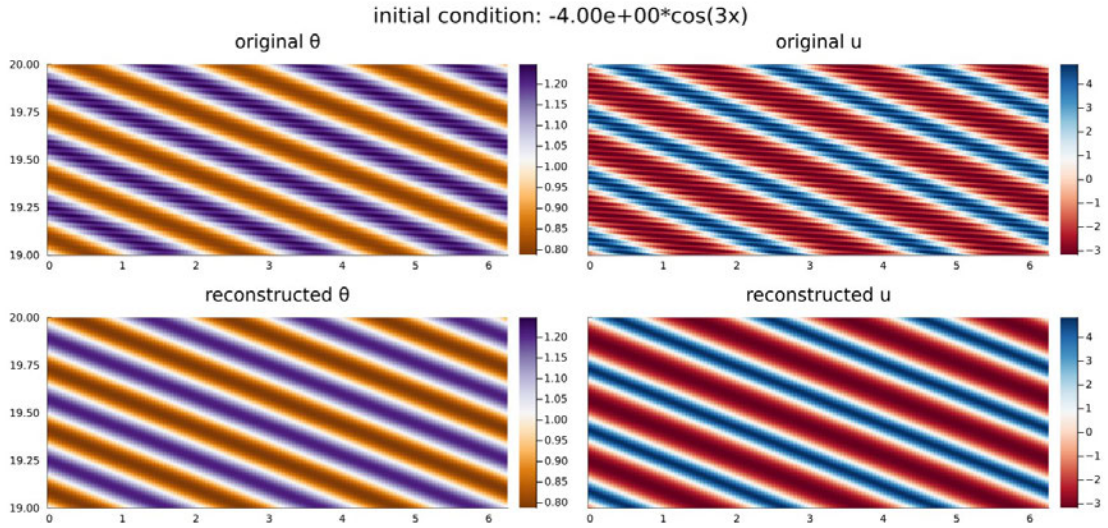


Figure 5: Comparison of original and reconstructed KdV using the θ function reconstruction for initial condition $u(x,0) = -4\cos(3x)$.

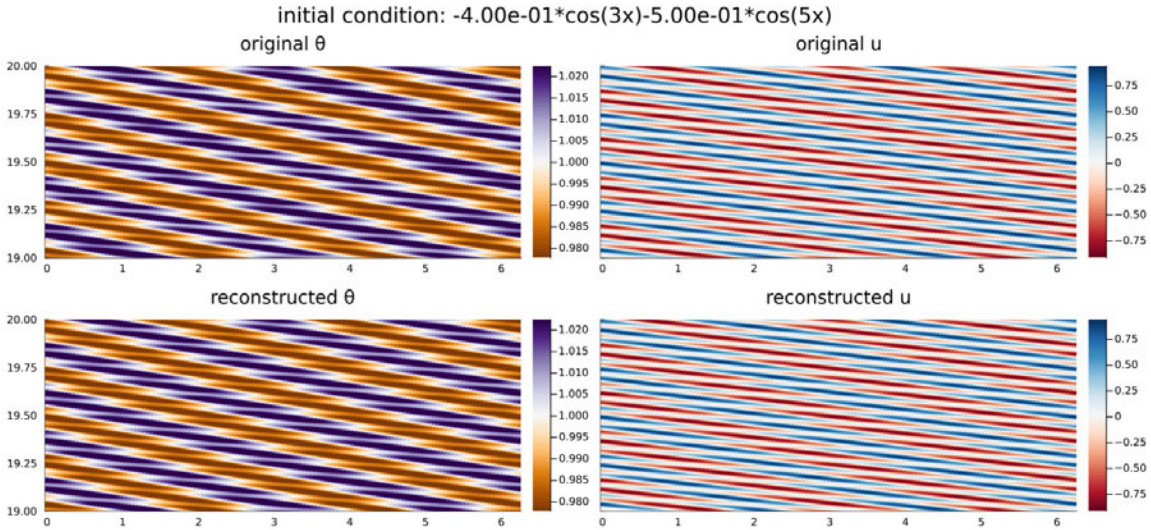


Figure 6: Comparison of original and reconstructed KdV using the θ function reconstruction for initial condition $u(x,0) = -0.4\cos(3x) - 0.5\cos(5x)$.

Despite the ability to capture all of the large scale details of the system, the theta function formulation with more than 1 degree of freedom allows us to quantify the amount of nonlinear interaction between different wave modes. We were able to compute the coefficients of the interaction matrix B for all of the experiments described in this section (table 2). The magnitude of the linear interaction coefficients $B_{1,1}$ and $B_{2,2}$ decrease as A_1 and

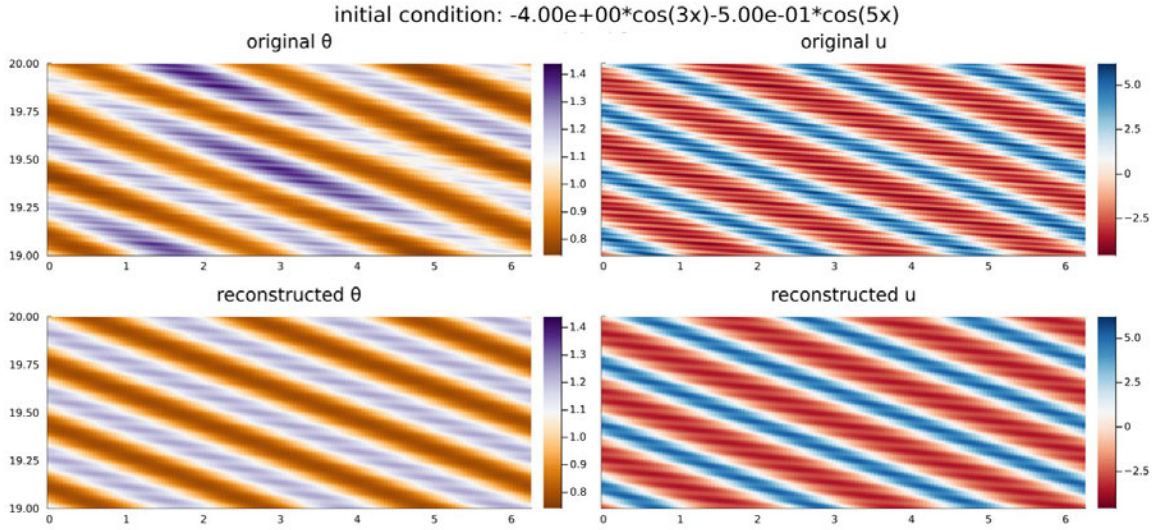


Figure 7: Comparison of original and reconstructed KdV using the θ function reconstruction for initial condition $u(x,0) = -4\cos(3x) - 0.5\cos(5x)$.

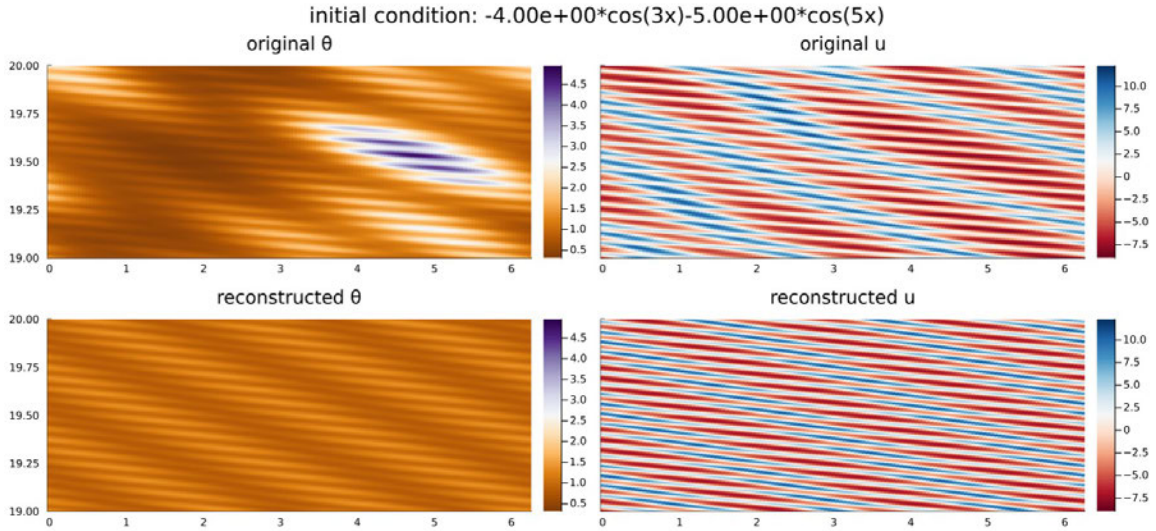


Figure 8: Comparison of original and reconstructed KdV using the θ function reconstruction for initial condition $u(x,0) = -4\cos(3x) - 5\cos(5x)$.

A_2 increase, which matches expectations as $B_{1,1}$ and $B_{2,2}$ are inversely correlated to the magnitude of the solution as these coefficients (eq 6). However, $B_{1,2}$ is positively correlated to the magnitude of the nonlinear interacts to the solution as the $B_{1,2}$ can be multiplied by a negative term in the exponential. As expected from theory, we found that as the amplitude of the initial condition increased, the magnitude of the nonlinear interaction $B_{1,2}$

increased relative to linear interaction terms $B_{1,1}$ and $B_{2,2}$. We note also that the linear interaction terms $B_{1,1}$ and $B_{2,2}$ change with A_1 and A_2, respectively. Further investigation will be required to see if the relative value of $B_{1,2}$ to $B_{1,1}$ and $B_{2,2}$ could be a heuristic as to validate whether the theta function representation will produce an accurate reconstruction of the system.

4 Conclusions and Future Work

We can analyze various aspects of the KdV equation using DMD. We are able to estimate the characteristic frequencies and wavenumbers of solutions as well as compute the parameters of the theta function for 1 or 2 degrees of freedom. It is likely that these same methods for frequency and wavenumber estimation would be effective for other systems of weakly nonlinearly interacting waves, as well as the theta function parameter estimation where a theta function can be related nicely to the system of interest.

Primary areas of interest to advance this work include:

- Computing parameters of the theta function for $N > 2$. This would require a robust implementation of the PSLQ iteration procedure described in section 2.1 to be able to choose the eigenvalue lattice without visual or ad hoc methods. This would also allow automatic propagation of the matrix M that is constructed when computing the coefficients of B (section 2.2).
- Analyze the robustness of frequency and wavenumber estimation methods to noise so that these ideas could be well applied to experimental or observational data. The most practically useful portion of this avenue would be to find heuristics to determine whether or not the estimates and reconstructions are reliable.
- Find a similar way to quantify nonlinear wave interaction for systems where the theta function is not known to directly relate to the solution of an equation. Ideally, this would also allow these methods to apply to observational data such as oceanographic surface waves.

5 Acknowledgements

My thanks to Jeremy Parker and Peter Schmid for their time and guidance this summer advising my project. (An extra thank you goes to Jeremy for a supreme amount of patience both within the cottage and while climbing rocks.) Thanks go also to Stefan and Colm for directing GFD this summer, Laure and Peter for their lectures, and, of course, the other fellows for an entertaining summer of swimming and losing at sorts of activities (trivia and softball included).

A Appendix

A.1 Full DMD algorithm

Given some dynamical system $\{\mathbf{v}_j\}_j$, DMD seeks to find a linear function A such that $\mathbf{v}_{j+1} = A\mathbf{v}_j$. Let V be the $N \times M$ data matrix where $\mathbf{v}_j$ are the columns of V. The base DMD algorithm approximates A such that:

$$V_2 \approx AV_1$$

where V_1, V_2 is are $(N-1) \times M$ matrices such that $V_1 = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N]$ and $V_2 = [\mathbf{v}_2, \mathbf{v}_3, \ldots, \mathbf{v}_N]$.

Base algorithm The base DMD algorithm computes A as follows [4]:

1. Take the singular value decomposition (SVD) of V, i.e.

$$V_1 = Y\Sigma X^* \tag{12}$$

2. Compute A with the pseudoinverse of V_1:

$$A = V_2 X \Sigma^{-1} Y^* \tag{13}$$

One can also (often more efficiently) compute $\tilde{A}$ which is an $r \times r$ projection of the matrix A onto principal orthogonal decomposition modes, where r is the rank of the reduced SVD approximation of V_2.

$$\tilde{A} = Y^* aY = Y^* V_2 X \Sigma - 1 \tag{14}$$

3. Compute the eigendecomposition of $\tilde{A}$ to get:

$$\tilde{A}W = W\Lambda \tag{15}$$

where Λ is a diagonal matrix containing the eigenvalues of A which are λ_k and the columns of W are the eigenvectors. We can find the matrix U of eigenvectors of A (which are the spatial modes $\mathbf{u}_k$ of DMD) with

$$U = V_2 X \Sigma^{-1} W \tag{16}$$

We can recompute the approximate solution $\mathbf{v}(t)$ as:

$$\mathbf{v}(t) \approx \sum_k^r \mathbf{u}_k \exp(\omega_k t)\mathbf{b}_k,\ \omega_k = \frac{\ln \lambda_k}{\Delta t} \tag{17}$$

where $\mathbf{b}_k$ is the initial amplitude of each mode. (The vector $\mathbf{b}$ which contains all $\mathbf{b}_k$ is often computed via psuedoinverse as $\mathbf{b} = \Phi^\dagger \mathbf{v}_1$). In all of the numerical experiments of this work, we choose r to be M, the size of the spatial dimension.

Delay Embedding We add delay embedding to previous procedure by writing down a new data matrix from the dynamical system $\{\mathbf{v}_j\}_j$. For d delays, the new data matrix $V^{'}$ is of size $(N-d) \times (d+1)M$ and is written so that the jth column $\mathbf{v}_j^{'}$ of $V^{'}$ is $\mathbf{v}_j^{'} = (\mathbf{v}_j, \mathbf{v}_{j+1}, \ldots, \mathbf{v}_{j+d}$. We then proceed as in the base algorithm substituting $V^{'}$ for V when required.

A.2 Connection to invariant 2-tori

The initial motivation for this project was to be able to use DMD to generate good guesses for invariant 2-tori in systems with chaotic dynamics. Studying unstable fixed points and periodic orbits is a known approach to understand chaotic dynamics. As with unstable periodic orbits, we expect that unstable 2-tori are generic in nonlinear dissipative systems [18].

Previously, Page and Kerswell [16] found that DMD applied to turbulent flows can be used to make guesses as to periodic orbits using a time series of length less than one 3rd of the full period. Ideally, we would like to be able to extend this result to invariant 2-tori in turbulence which requires accurate guesses as to the two generating frequencies of a given tori. Given further work on estimating the relative amplitudes of each mode associated with the generating frequencies, we anticipate that these results (particularly in subsection 2.1) could be useful in identifying and characterizing invariant 2 tori in chaotic systems.

References

[1] H. Arbabi and I. Mezic, *Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the koopman operator*, SIAM Journal on Applied Dynamical Systems, 16 (2017), pp. 2096–2126.

[2] D. H. Bailey and J. M. Borwein, *PSLQ: An Algorithm to Discover Integer Relations*, Tech. Rep. LBNL-2144E, Lawrence Berkeley National Lab. (LBNL), Berkeley, CA (United States), Apr. 2009.

[3] J. P. Boyd, *Long Wave/Short Wave Resonance in Equatorial Waves*, Journal of Physical Oceanography, 13 (1983), pp. 450–458. Publisher: American Meteorological Society Section: Journal of Physical Oceanography.

[4] S. L. Brunton and J. N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*, Cambridge University Press, 2 ed., 2022.

[5] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, *Dedalus: A flexible framework for numerical simulations with spectral methods*, Physical Review Research, 2 (2020), p. 023068. ADS Bibcode: 2020PhRvR...2b3068B.

[6] I. C. Christov, *Hidden solitons in the Zabusky-Kruskal experiment: Analysis using the periodic, inverse scattering transform*, Mathematics and Computers in Simulation, 82 (2012), pp. 1069–1078. arXiv:0910.3345 [nlin].

[7] H. Ferguson, D. Bailey, and S. Arno, *Analysis of PSLQ, an integer relation finding algorithm*, Mathematics of Computation, 68 (1999), pp. 351–369.

[8] E. Fermi, P. Pasta, S. Ulam, and M. Tsingou, *STUDIES OF THE NONLINEAR PROBLEMS*, Tech. Rep. LA-1940, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), May 1955.

[9] G. Froyland, D. Giannakis, B. R. Lintner, M. Pike, and J. Slawinska, *Spectral analysis of climate dynamics with operator-theoretic approaches*, Nature Communications, 12 (2021), pp. 1–21. Cc_license_type: cc_by Number: 1 Primary_atype: Research Publisher: Nature Publishing Group Subject_term: Applied mathematics;Climate sciences Subject_term_id: applied-mathematics;climate-sciences.

[10] M. R. Jovanović, P. J. Schmid, and J. W. Nichols, *Sparsity-promoting dynamic mode decomposition*, Physics of Fluids, 26 (2014), p. 024103. Publisher: American Institute of Physics.

[11] P. D. Lax, *Almost Periodic Solutions of the KdV Equation*, SIAM Review, 18 (1976), pp. 351–375.

[12] I. Mezić, *Analysis of fluid flows via spectral properties of the Koopman operator*, Annual Review of Fluid Mechanics, 45 (2013), pp. 357–378. Publisher: Annual Reviews.

[13] W. H. Munk, *The Solitary Wave Theory and Its Application to Surf Problems*, Annals of the New York Academy of Sciences, 51 (1949), pp. 376–424. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1749-6632.1949.tb27281.x.

[14] A. Osborne, *Nonlinear Ocean Waves and the Inverse Scattering Transform*, Academic Press, Apr. 2010. Google-Books-ID: wdmsn9icd7YC.

[15] A. R. Osborne, M. Serio, L. Bergamasco, and L. Cavaleri, *Solitons, cnoidal waves and nonlinear interactions in shallow-water ocean surface waves*, Physica D: Nonlinear Phenomena, 123 (1998), pp. 64–81.

[16] J. Page and R. R. Kerswell, *Searching turbulence for periodic orbits with dynamic mode decomposition*, Journal of Fluid Mechanics, 886 (2020), p. A28. Publisher: Cambridge University Press.

[17] J. P. Parker and J. Page, *Koopman Analysis of Isolated Fronts and Solitons*, SIAM Journal on Applied Dynamical Systems, 19 (2020), pp. 2803–2828. Publisher: Society for Industrial and Applied Mathematics.

[18] J. P. Parker and T. M. Schneider, *Invariant tori in dissipative hyperchaos*, July 2022. arXiv:2207.05163 [nlin].

[19] J. L. Proctor, S. L. Brunton, and J. N. Kutz, *Dynamic Mode Decomposition with Control*, SIAM Journal on Applied Dynamical Systems, 15 (2016), pp. 142–161. Publisher: Society for Industrial and Applied Mathematics.

[20] P. J. Schmid, *Dynamic mode decomposition of numerical and experimental data*, Journal of Fluid Mechanics, 656 (2010), pp. 5–28. Publisher: Cambridge University Press.

[21] ——, *Dynamic Mode Decomposition and Its Variants*, Annual Review of Fluid Mechanics, 54 (2022), pp. 225–254.

[22] P. J. Schmid, L. Li, M. P. Juniper, and O. Pust, *Applications of the dynamic mode decomposition*, Theoretical and Computational Fluid Dynamics, 25 (2011), pp. 249–259.

[23] N. J. Zabusky and M. D. Kruskal, *Interaction of "Solitons" in a Collisionless Plasma and the Recurrence of Initial States*, Physical Review Letters, 15 (1965), pp. 240–243. Publisher: American Physical Society.

Equatorial Ocean Dynamics on Enceladus Driven by Ice Topography

Rui Yang

1 Introduction

Enceladus is a small moon of Saturn, with a radius of around 250 km. Despite its small size, libration motion [45] and the jet sprays over the south pole [35, 12, 43, 16] indicates that Enceladus still retains a global surface ocean that is 40 km deep on average. Particles and gases sampled from these jets indicate the presence of hydrogen [47], organic matter [37], silica nanoparticles [13] and a modestly alkaline environment [10], all suggestive of potential to host life [47, 9, 44, 30].

The energy source to sustain the ocean is most likely to be related to tidal dissipation generated in the ice shell [2, 42, 32, 34], ocean [25] and inner core [4, 27]. Provided the global heat budget is in balance, ocean circulations and eddies can still alter the shape of the ice shell by redistributing heat across different latitudes and longitudes. Since the freezing water under thick ice (high pressure) tends to be colder than that under thin ice (low pressure), ocean mixing driven by the ice thickness variation will tend to converge heat toward the thick ice regions, causing ice to melt. In addition, ice flow driven by pressure gradient [1, 21] will further flatten the ice shell; the conductive heat loss is more efficient over the thin ice regions due to the weakened insulation effect. All these processes tend to remove the ice thickness variation, yet, outstanding ice topography is found on Enceladus [11]. Inhomogeneous tidal dissipation and the ice rheology effect have been proposed as mechanism to sustain the observed ice geometry [11, 2, 21, 23]. However, even with that, the ocean heat transport (OHT) cannot be arbitrarily strong: if the convergence of OHT toward the thick ice regions exceeds the heat conduction through the ice shell, heat will inevitably accumulate and ice will inevitably melt, because tidal dissipation cannot be negative [23].

Another reason to study the overturning circulation and eddies in the ocean is because they govern the tracer transport, which may in turn affect the composition and properties of the ejecta. In fact, the size of silica nanoparticles collected from the ejecta has been used to estimate the transport timescale and the ocean salinity [14]. Soaked in the seawater, the silica nanoparticles will grow as being transported from the seafloor to the surface. If the transport takes too long or the ocean is too salty, the size of the particles would be greater than what is measured.

The ice thickness gradient may play a key role in driving the overturning circulations and eddies in Enceladus' ocean, because the meridional temperature gradient under the ice

due to the freezing point suppression by pressure is likely to be one order of magnitude larger than the vertical temperature gradient induced by a 40 mW/m^2 bottom heating [23], according to the inviscous scaling [6]. With equatorial water colder than the polar water due to the different pressure under the ice, sinking motion occurs in low latitudes. However, the ocean would circulate in the opposite direction if the ocean is salty enough [23]. This peculiar behavior stems from the anomalous expansion of fresh water near freezing point being suppressed by salinity. Vertical diffusion and buoyancy fluxes across the top and bottom boundaries [1] provides the necessary energy to balance dissipation in the interior and the frictious layers [18]. As a result, the meridional ocean circulation and the heat/tracer transport are strongly affected by the ocean salinity [23], core-shell heat partition [23] and the ocean diffusivity and viscosity [22, 20] – none of these factors is well constrained [36, 17, 14, 46, 4].

Previous studies on Enceladus ocean circulation either assumes a flat ice shell [40, 41] or a zonally symmetric one [23, 22, 20], ignoring the zonal ice thickness variations on Enceladus [11]. The zonal thickness variation is particularly prominent near the equator dominated by a wavenumber-2 mode. Such ice thickness variation will induce temperature and salinity gradients along the zonal direction, driving circulation swirling in the equatorial plane. In this work, we will focus on the region outside the tangent cylinder (TC) as marked in Fig.1. This region is unique in a way that the planetary rotation doesn't directly interfere with such a circulation except keeping the flow 2D, i.e., uniform along the rotation axis which is perpendicular to the circulation plane. These features are not expected for meridional circulation because meridional flow is strongly inhibited by rotation away from rough boundaries. The atmospheric walker circulation on earth driven by zonal temperature gradients at the sea surface [26, 7] may be a better analogue. However, unlike Enceladus ocean, the earth atmosphere is extremely thin compared to the planetary size. Vertical motion needs to be orders of magnitude smaller than the horizontal motion if mass continuity is to be satisfied. As a result, the vertical motion does not even show up in the lowest-order momentum equation [7], excluding the dynamics of the swirl motion we would like to study. Given these differences, we are compelled to reconsider the dominant balance in momentum and buoyancy. This work will focus on the low salinity scenarios, in which ocean is stably stratifed near the equator as dense warm water sinks from the poles and slides equatorward in the lower part of the ocean [23]. Understanding high salinity scenarios will require a different framework involving convective instability; this is out of our scope here.

2 Methods

2.1 Theoretical model

Here, a simplified model is used to understand how ice topography drives ocean flows. We start from the linear solution and then discuss the higher order corrections.

[1] The buoyancy flux from the ice produces (consumes) energy at low (high) salinity. The bottom heat flux produces (consumes) energy at high (low) salinity.

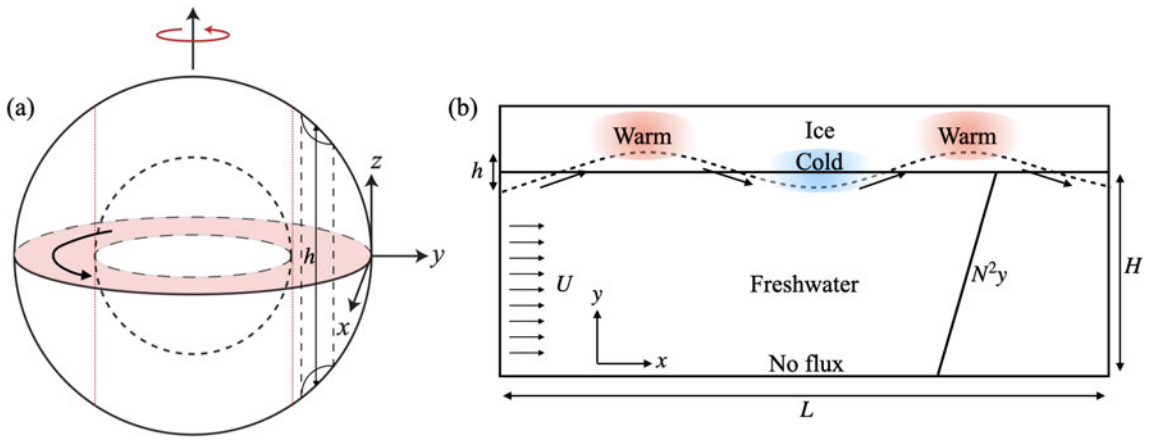


Figure 1: (a) Illustration of 3D spherical shell in Cartesian coordinate. The red shade region represents the equatorial region we are focusing on. (b) Illustration of the simplified rectangular setup of the equatorial region.

2.1.1 Simplified model equation

As sketched in figure 1(a), we consider the spherical shell geometry for Encedalus in the local Cartesian coordinate, where x is the longitude at the equator (positive towards west), y the radius (points outward radially), and z the rotation axis. The governing equations are

$$\frac{D}{Dt}u - fv = -\frac{\partial}{\partial x}p + \nu\nabla^2 u \tag{1}$$

$$\frac{D}{Dt}v + fu = -\frac{\partial}{\partial y}p + b + \nu\nabla^2 v \tag{2}$$

$$0 = -\frac{\partial}{\partial z}p \tag{3}$$

$$\frac{D}{Dt}b = \kappa\nabla^2 b \tag{4}$$

where u, v, w are the zonal, radial, and vertical (rotation axis) velocity, respectively, p the pressure, b the buoyancy, f the Coriolis force, κ the thermal diffusivity, and ν the kinematic viscosity.

The vorticity equation along z can be derived by taking x-derivative of the v-momentum equation minus the y-derivative of the u-momentum equation.

$$\frac{\partial}{\partial t}\zeta + \mathbf{u}\cdot\nabla\zeta = -(f+\zeta)\frac{\partial w}{\partial z} + \frac{\partial b}{\partial x} + \nu\nabla^2\zeta, \tag{5}$$

where $\zeta = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$.

According to the Taylor-Proudman theorem, rotation-dominant, inviscid, imcompressible fluid has u and v invariant along the rotating axis z. As the swirl motion approaches the top and bottom boundaries in z, the flow needs to be parallel to the tilted boundaries,

$$w(\pm h/2) = \pm v\frac{\partial}{\partial y}\frac{h}{2} \quad \Rightarrow \quad \frac{\partial}{\partial z}w = v\frac{1}{h}\frac{\partial}{\partial y}h \equiv \frac{1}{f}\beta v. \tag{6}$$

This boundary condition allows us to replace the w in Eq.5 with v. Assuming a small Rossby number, which is true when the resultant swirl is much smaller than the planetary rotation f, we finally obtain

$$\begin{aligned}\frac{\partial}{\partial t}\zeta + \mathbf{u}\cdot\nabla\zeta &= -(f+\zeta)\frac{\partial w}{\partial z} + \frac{\partial b}{\partial x} + \nu\nabla^2\zeta \\ &= (1+\zeta/f)\beta v + \frac{\partial b}{\partial x} + \nu\nabla^2\zeta \\ &\simeq \beta v + \frac{\partial b}{\partial x} + \nu\nabla^2\zeta.\end{aligned} \tag{7}$$

Again, because the equatorial swirl flow is mostly 2D, we can introduce a stream function ψ' to represent the flow field,

$$u = -\frac{\partial \psi}{\partial y} \quad , \quad v = \frac{\partial \psi}{\partial x} \quad \Rightarrow \quad \zeta = \nabla^2\psi \tag{8}$$

If we further ignore the curvature effects, the 2D annulus can be converted to a rectangular shape, as illustrated by figure 1(b). To account for the stratification induced by meridional circulation [23, 22, 20] and the thermal wind induced by the meridional density gradient, the total buoyancy and streamfunction contains the background state and perturbation,

$$= -Uy + \quad ' \tag{9}$$

$$b = N^2 y + b' \tag{10}$$

The background state is diagnosed from 3D MITgcm simulations, whose setup is summarized in section 3.1. The final governing equations for the equatorial flow are

$$\frac{\partial \zeta}{\partial t} + J(\psi' - Uy, \zeta) = \beta\frac{\partial \psi'}{\partial x} + \frac{\partial b'}{\partial x} + \nu\nabla^2\zeta \tag{11}$$

$$\zeta = \nabla^2\psi' \tag{12}$$

$$\frac{\partial b'}{\partial t} + J(\psi' - Uy, b' + N^2 y) = \kappa\nabla^2 b' \tag{13}$$

Without losing generality, we consider a sinusoidal ice topography at the ocean-ice interface. The impacts on the ocean underneath is two-fold. On one hand, the freezing point of ice T_f depends on pressure $p = \rho g h(x)$, where $h(x)$ is the thickness of ice along x, so the surface buoyancy anomaly should be relaxed toward $b_0 = \alpha T_f g$, which corresponds to the freezing point. On the other hand, flow needs to follow the topography. Assuming the topography is sufficiently small, the boundary conditions at $y = 0, H$ can be written as

$$\begin{aligned} ' &= Uh\cos(kx),\ y = 0 \\ ' &= 0,\ y = -H \\ \zeta &= 0,\ y = 0 \\ \zeta &= 0,\ y = -H \\ \kappa\partial_y b' &= -\gamma_T(b - \alpha g\cdot C_h\rho g h\cos(kx)),\ y = 0 \\ \partial_y b' &= 0,\ y = -H \end{aligned} \tag{14}$$

Here, we set the tracer mixing coefficient $\gamma_T = 10^{-5}$ m/s, and the ice topography amplitude $h = 1$ km. The changing rate of freezing point with pressure $C_h \approx 7.8 \times 10^{-8}$ K $\cdot$ m^{-1}s^{-2}, reference density $\rho = 10^3$ kg/m^3, surface gravity on Enceladus $g = 0.1$ m/s^2, ocean depth $H = 40$ km. Physical constants are summarized in Table 1. We set no flux boundary conditions at the bottom boundary and periodic in the horizontal direction.

To investigate the flow dynamics in more details. Next, we consider to do the order expansion to the full equations. We start from the lowest order equation, i.e. linearizaed equation.

2.1.2 Linearized equations

Given h is sufficiently small, the lowest order equation only contains linear terms. We pursue a steady state solution forced by the ice topography forcing (Eq.14) by dropping the time-derivatives,

$$U\frac{\partial \zeta}{\partial x} = \beta\frac{\partial \psi'}{\partial x} + \frac{\partial b'}{\partial x} + \nu\nabla^2\zeta \tag{15}$$

$$\nabla^2\psi' = \zeta \tag{16}$$

$$U\frac{\partial b'}{\partial x} + N^2\frac{\partial \psi'}{\partial x} = \kappa\nabla^2 b' \tag{17}$$

Substituting the following solution ansatz for b', ψ', and ζ,

$$b' = \sum_{n=1}^{6} A_n e^{s_n y} e^{ikx} \tag{18}$$

$$\psi' = \sum_{n=1}^{6} B_n e^{s_n y} e^{ikx} \tag{19}$$

$$\zeta = \sum_{n=1}^{6} C_n e^{s_n y} e^{ikx} \tag{20}$$

into equations 15-17, we have

$$ikUC_n - ik\beta B_n - ikA_n - \nu(s_n^2 - k^2)C_n = 0, \tag{21}$$

$$(s_n^2 - k^2)B_n - C_n = 0, \tag{22}$$

$$ikUA_n + ikN^2 B_n - \kappa(s_n^2 - k^2)A_n = 0. \tag{23}$$

After replacing B_n and C_n with A_n, we obtain a 6-order equation for the modes s_n

$$[i\nu\kappa p^3 + kU(\nu+\kappa)p^2 - (k\kappa\beta + ik^2U^2)p + ik^2(U\beta - N^2)]A_n = 0 \tag{24}$$

with $p = s_n^2 - k^2$, and also the relation between A_n, B_n and A_n, C_n

$$B_n = -\frac{kU + i\kappa p}{kN^2}A_n \tag{25}$$

$$C_n = -\frac{kU + i\kappa p}{kN^2}pA_n \tag{26}$$

Then substitute 18-20 into the boundary conditions 14, we can solve the solution of A_n, B_n, and C_n from a linear equation set.

2.1.3 Higher order equations

After keeping the lowest order equations, now we pursue the second order correction. The dropped nonlinear advection terms in the lowest order equation (Eq.15-17) will lead to corrections in the next order. Here we expand ψ', ζ, b based on ϵ as

$$\psi' = \psi'_0 + \epsilon\psi'_1 + \epsilon^2\psi'_2 + ... \tag{27}$$

$$\zeta = \zeta_0 + \epsilon\zeta_1 + \epsilon^2\zeta_2 + .. \tag{28}$$

$$b = b_0 + \epsilon b_1 + \epsilon^2 b_2 + ... \tag{29}$$

Solving the governing full equations with the solution above, one can obtain the solutions for higher orders. The magnitudes of the higher order results are found to be two orders smaller than the 0th order result so we neglect them. More details of the derivations of high-order equations are shown in Appendix.

2.2 Heat budget, flow penetration depth, and tracer transport timescale

From the linear solution, we can estimate the quantities we are interested in, i.e. the heat fluxes $\mathcal{H}$, which measures the heat fluxes from the ocean into ice crust, flow penetration depth h_0, which measures how deep the topography-driven flow penetrates downward into the deeper ocean, and tracer transport time scale t_0, which measure how long does tracer transport from the bottom boundary (sea floor) to the top boundary (sea surface).

Since we impose a no-flux boundary condition at the bottom boundary, the main heat fluxes in the ocean include the horizontal heat flux in the interior $\mathcal{H}_i$ and the heat flux at the top boundary $\mathcal{H}_t$, which need to be balanced with each other to close the heat budget. The expressions for $\mathcal{H}_i$ and $\mathcal{H}_t$ are

$$\mathcal{H}_i(x) = \frac{\rho C_p}{\alpha g}\frac{d}{dx}\int_{-H}^{0}(u+U)(b'+N^2y)dy \tag{30}$$

$$\mathcal{H}_t(x) = \frac{\rho C_p}{\alpha g}\kappa b'_y\Big|_{y=0} \tag{31}$$

where C_p is the specific heat capacity of water.

The flow penetration depth h_0 measures how deep the flow can penetrate downward when the interior is stably stratified, which can indicate the transport efficiency of tracers. We define h_0 as the height where the velocity decays to 10% of the max velocity at the top boundary.

$$\overline{u}(y=h_0) = 0.1\overline{u}_{y=0} \quad \Rightarrow \quad h_0 = \frac{1}{\log(e^{(s_n)_{\min}})} \tag{32}$$

where $\overline{u}$ is the x-averaged profile of $|u|$.

The tracer transport timescale t_0 measures the time needed for passive tracers to move from the bottom to the top boundary, which can be compared to the estimation based on silica nanoparticles [14]. Here, t_0 is estimated by the vertical integral of inversed vertical velocity

$$t_0 = \int_{-H}^{0}\frac{dy}{v} \tag{33}$$

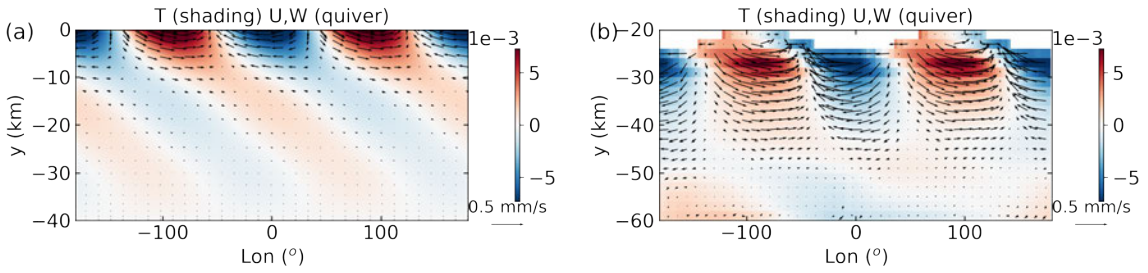


Figure 2: (a) Flow pattern of the linear solution from the model, with the temperature field in shading and velocity u, v in arrows. (b) The flow pattern of the 3D simulation from MITgcm, with the temperature field in shading and velocity u, v in arrows. The white region represents the ice topography.

3 Results

In this part, we will consider the ocean pattern and our interested quantities in fresh/salty water and with/without ice topography effect.

3.1 Configuration of 3D simulation.

To ensure that the simplified theoretical model captures the key dynamics and to obtain the background stratification N^2 and zonal flow U, we set up a 3D global ocean circulation model using the Massachusetts Institute of Technology OGCM (MITgcm [31, 28]). The model configuration is almost the same as the one used in *Kang et al. 2022* [23], except that the zonal dimension is considered, for it is necessary to this work. Here we will briefly review how the boundary conditions are set up and the reasoning behind them. For more detailed model description, readers are referred to the "Materials & Methods" section in *Kang et al. 2022* [23]. The parameters used in the model is summarized in Table.1.

In our model, the ocean is covered by an ice topography that resembles that of the present-day Enceladus [11] (see Fig.XY), which is assumed to be sustained against the ice flow by a prescribed freezing/melting q (see Fig.XY), regardless of the ice shell's heat budget. Unsurprisingly, q is generally in phase with the ice thickness H, because ice topography can be sustained only if the thick ice is freezing and the thin ice is melting. By prescribing q, we guarantee the ice shell to be in mass balance and, furthermore, cut off the positive feedback loop between the ocean heat transport and the ice freezing/melting rates, thus preventing the simulated circulation from seeking a completely new state.

At the water-ice interface, a downward salinity flux $S_0 q$ is imposed to represent the brine rejection and freshwater production associated with freezing/melting, where $S_0 = 4$ psu is the assumed ocean salinity. Meanwhile, the ocean temperature there is restored toward the local freezing point. Thus, the ocean will deposit heat to the ice when its temperature is slightly higher than the freezing point, and vice versa. Since the water tends to be colder under thicker ice, heat tends to be converged and deposited to the thick-ice regions. In order for the heat budget of the ice to close, this ocean-ice heat exchange $\mathcal{H}_{\text{ocn}}$, together with the tidal heat produced in the ice $\mathcal{H}_{\text{ice}}$ and the latent heat released $\mathcal{H}_{\text{latent}}$ ($\mathcal{H}_{\text{latent}} = \rho L_f q$,

where ρ and L_f are the density and fusion energy of ice) should balance the conductive heat loss through the ice shell $\mathcal{H}_{\rm cond}$

$$\mathcal{H}_{\rm ocn} = \mathcal{H}_{\rm cond} - \mathcal{H}_{\rm ice} - \mathcal{H}_{\rm latent}. \tag{34}$$

Among these terms, $\mathcal{H}_{\rm ice}$ is always non-negative and $\mathcal{H}_{\rm latent}$ is positive under thick ice, which requires the magnitude of $\mathcal{H}_{\rm ocn}$ to be no greater than $\mathcal{H}_{\rm cond}$. The conductive heat loss rate through a 30 km ice shell is around 30 mW/m^2 [23], setting an upper limit of $\mathcal{H}_{\rm ocn}$ if the ice shell is in equilibrium state.

The ocean-ice interaction takes tens of thousands of simulated years to fully equilibrate. To keep the computational cost manageable, we are forced to choose a rather coarse resolution: the globe is divided into 6 faces, each of which has 32×32 grids to resolve. There are totally 70 layers in the vertical direction (the y direction in theoretical model setup), unevenly distributed to better resolve the dynamics near the surface. The layer thickness increases from less than 500 m near the surface to 1600 m toward the bottom. At this resolution, we cannot resolve convection or baroclinic eddies, because the convective cone scaling is less than 1 km [19] and the Rhines' scale is only a few kilometers [20]. The cross-isopycnal transport by convection is parameterized by setting vertical diffusivity to a much larger value in regions with unstable stratification. The baroclinic eddies are parameterized by the Gent-McWilliams (GM) scheme [39, 8], a widely-used approach to parameterize the associated eddy-induced circulation and mixing of tracers along isopycnal surfaces in modeling Earth's ocean. Parameters used in these schemes are in Table.1.

3.2 Configuration of 2D simulations.

We solve the 2D simplified model equations (Eq. 11-13) with the boundary conditions (Eq. 14) by the spectral PDE solver Dedalus [3]. We impose the background stratification N^2, zonal flow speed U, and topography amplitude h, obtained from the 3D simulation. The basic values for the parameters used is summarized in Table 1. Note that the actual ν and κ are much smaller ($\sim 10^{-6}$) in Enceladus' ocean. Due to the limitation of computation power, we choose larger ν and κ and fix the Prandtl number $Pr = \nu/\kappa = 1$. Different situations are considered, including with/without topography effect and fresh/salty water. The dependence of density ρ on temperature T and salinity S (equation of state) is expressed as follows

$$\rho(\theta, S) = \rho_0 \left(1 - \alpha_{\rm T} \left(\theta - \theta_0\right) + \beta_{\rm S} \left(S - S_0\right)\right) \tag{35}$$

3.3 Flow pattern with fresh water

Assume the water is fresh and neglect the ice topography effect, i.e. $h(x) = 0$, the top boundary condition becomes $b' = 0,\ \psi' = 0$. In this situation, the flow is completely stably stratified with a background zonal flow. It means that the tracers cannot be transported by vertical velocity but only by pure diffusion ($t_0 \to \infty$). Thus this is not a possible condition for the Enceladus ocean.

Then we consider the ice topography effect with $h(x) = 1$ km, and other main parameters are $\nu = 60\text{m}^2/\text{s},\ \kappa = 10^{-4}$ m^2/s, $U = 10^{-3}$ m/s, $N^2 = 10^{-11}$ s^{-2}, same as those in the 3D simulation. The result from the linear solution is shown in figure 2(a). One can see that

Symbol	Name	Definition/Value
	Enceladus parameters	
a	radius	252 km
D	global mean ocean depth	40 km: ref [11]
Ω	rotation rate	5.307×10^{-5} s^{-1}
g_0	surface gravity	0.113 m/s^2
$\bar{T}_s$	mean surface temperature	59K
	Physical constants	
L_f	fusion energy of ice	334000 J/kg
C_p	heat capacity of water	4000 J/kg/K
C_h	melting point suppression rate with pressure	7.61×10^{-4} K/dbar
ρ_i	density of ice	917 kg/m^3
ρ_w	density of the ocean	Eq. 35
α	thermal expansion coeff.	$10^{-5}/K$
β	haline contraction coeff.	7.82×10^{-4}/psu [29]
κ_0	conductivity coeff. of ice	651 W/m: ref [33]
η_m	ice viscosity at freezing point	10^{14} Ps·s
	Default parameters in the ocean model	
$\nu_h,\ \nu_v$	horizontal/vertical viscosity	20 m^2/s
$\tilde{\nu}_h,\ \tilde{\nu}_v$	bi-harmonic hyperviscosity	10^7 m^4/s
$\kappa_h,\ \kappa_v$	horizontal/vertical diffusivity	0.001 m^2/s
$\kappa_{\rm conv}$	convective mixing rate	3 m^2/s
$\alpha_{\rm GM}$	the universal constant used in GM scheme	0.015
$l_{\rm GM}$	the mixing length scale used in GM scheme	3 km
$S_{\rm GM}$	the maximum slope before clipping	0.2
$(\gamma_T,\ \gamma_S,\ \gamma_M)$	water-ice exchange coeff. for T, S & momentum	(10^{-5}, 10^{-5}, 10^{-4}) m/s
	Default parameters in the 2D numerical simulations	
h	amplitude of ice topography	1 km
ν	ocean viscosity	0.1 m^2/s
κ	ocean thermal diffusivity	0.1 m^2/s
U	equatorial zonal flow speed	0.001 m/s
N^2	stable stratification	10^{-11} s^{-2}

Table 1: Parameters used in our study.

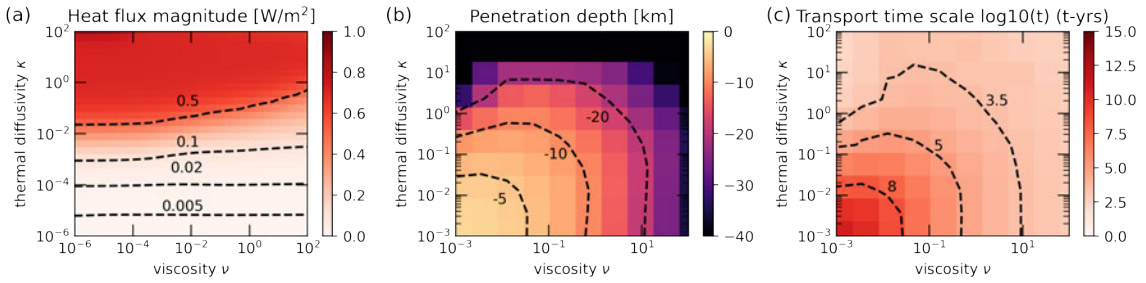


Figure 3: (a) Heat map of heat flux magnitude (b) Flow penetration depth (c) Tracer transport timescale from linear solutions as a function of ν and κ. The dashed lines are contour lines with the value as black numbers.

the flow pattern is mainly driven by the top topography: a wavy flow pattern along the longitude direction near the top boundary and decays as depth increases. The temperature near the top boundary is also governed by the pressure-dependent freezing point, which follows the topography. In figure 2(b), we plot the result from the MITgcm with the same parameters. One can see that the linear solution from our model shows a good agreement with that from the MITgcm.

3.3.1 Effect of parameters

Then we explore the effect of parameters on the penetration depth h_0, transport timescale t_0, and heat flux $\mathcal{H}$.

In figure 3, we show how ν and κ affect these quantities. In figure 3(a), we plot the heat flux magnitude as a function of ν and κ. The magnitude is calculated as the real part of the magnitude of $\mathcal{H}_i(x)$ from equation 30. One can see that the heat flux magnitude increases as κ increases, and decreases as ν increases. It is because that large κ means stronger heat diffusion and flow, which enhances the heat transport, while ν hardly affects the heat flux at small κ, since velocity is following the temperature contours. In figure 3(b), h_0 increases as κ and ν increases, because of the thicker boundary layer at large κ and ν help flow penetrate downward. This can be seen from the flow pattern in figure 4(a) (for $\nu = \kappa = 1$ m^2/s) and 4(c) (for $\nu = \kappa = 0.1$ m^2/s). In figure 3(c), t_0 decreases as κ and ν increases, opposite to the trend of h_0, since larger h_0 means stronger vertical mixing, so that smaller transport timescale t_0. In figure 4(b) and 4(d), we plot the linear solution with the same parameters as the full solution in figure 4(a) and 4(c). From these plots, we show that the linear solutions can estimate the full solution well.

The same analysis is also conducted for varying zonal speed U and stratification N. In figure 5(a), the heat flux increases as N increases and decreases as U increases. In figure 5(b), h_0 increases as U increases or N decreases, since stronger zonal flow or weaker stratification both help flow penetrate downward. In figure 5(c), t_0 shows an opposite trend as h_0. Then we compared the flow pattern at $U = 10^{-3}$ m/s, N $= 10^{-4}$ s^{-2} (figure 6(a)) and $U = 10^{-2}$ m/s, N $= 10^{-5.5}$ s^{-2} (figure 6(c)). In the former case, topography-driven flow is confined only near the top boundary due to strong stratification, while in the latter case with weaker stratification and stronger zonal speed, flow penetrates directly to the bottom

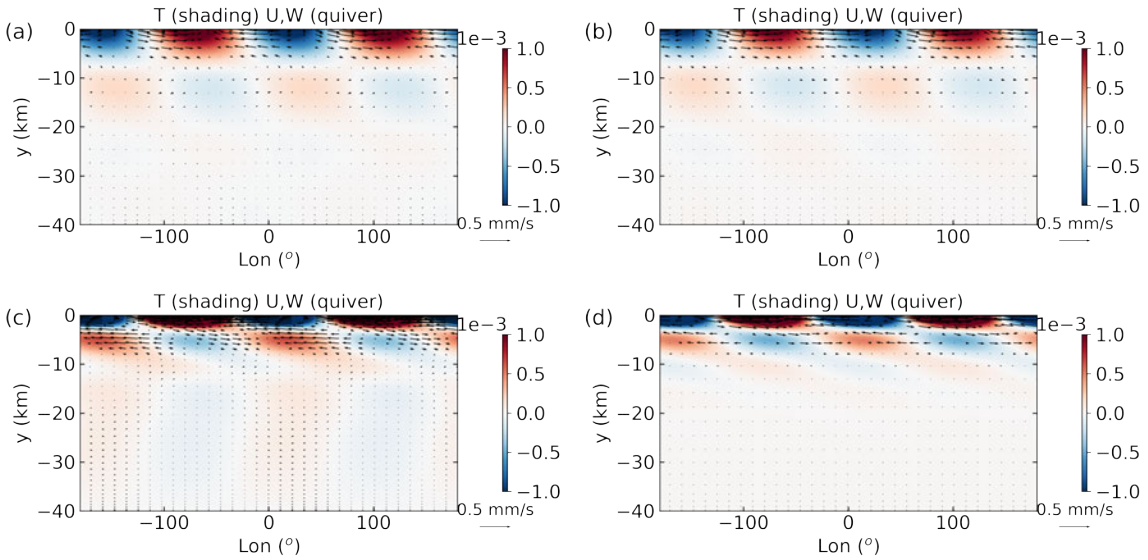


Figure 4: (a) Full solution result and (b) linear solution result with $\nu = \kappa = 1$ m^2/s, with buoyancy b' in shading and velocity u, w in arrows. (c) Full solution result and (d) linear solution result with $\nu = \kappa = 0.1$ m^2/s.

boundary. The corresponding linear solutions are shown in figure 6(b) and 6(d), which are basically same as the full solution. Only when U is large, the nonlinear terms affect the buoyancy field (asymmetric between positive and negative buoyancy pattern) near the top boundary.

3.4 Ocean dynamics with salty water

Now we consider salty water. The realistic effect of salinity on ocean dynamics is complicated: salinity results in double-diffusive convection [15, 38, 5] because salinity has 100 smaller diffusivity than temperature. Also, salinity affects the anomalous density maximum of water and the freezing point of ice. Here, we simplify the effect of salinity by only

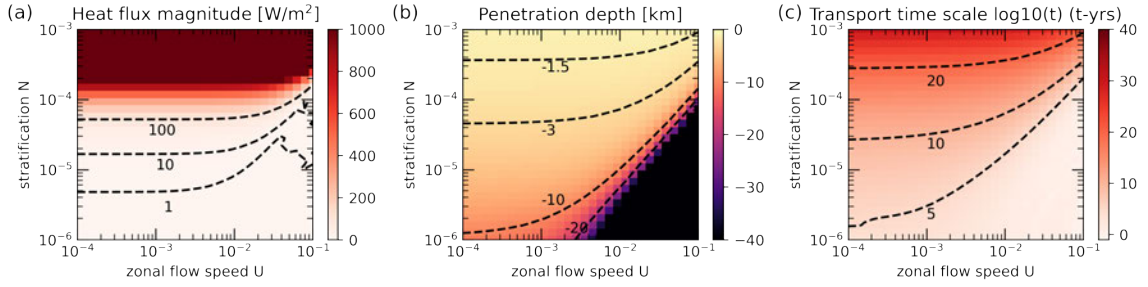


Figure 5: (a) Heat map of heat flux magnitude (b) Flow penetration depth (c) Tracer transport timescale from linear solutions as a function of U and N. The dashed lines are contour lines with the value as black numbers.

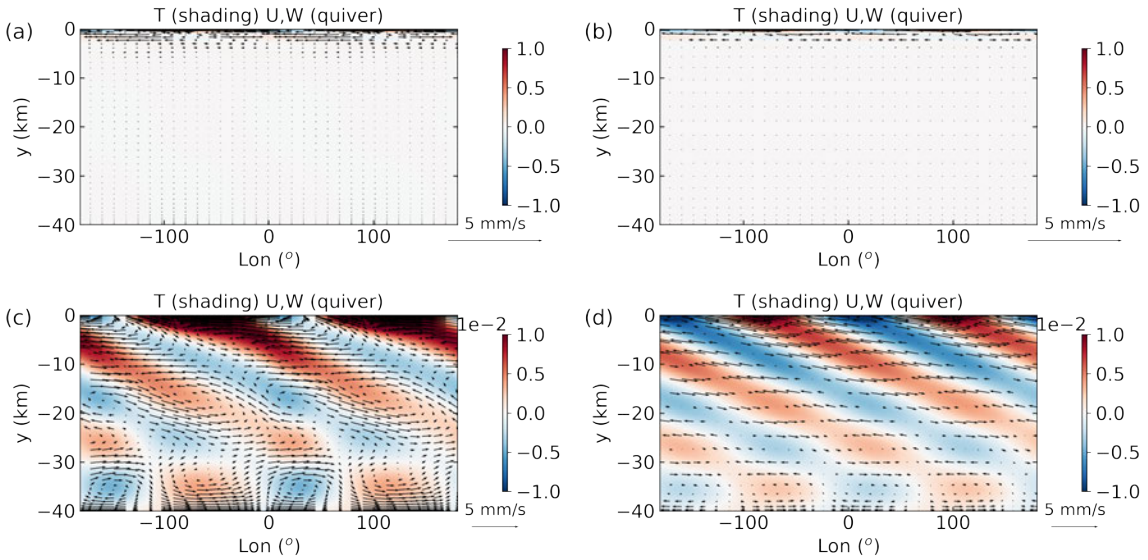


Figure 6: (a) Full solution result and (b) linear solution result with $U = 10^{-3}$ m/s, N = 10^{-4} s^{-2}, with buoyancy b' in shading and velocity u, w in arrows. (c) Full solution result and (d) linear solution result with $U = 10^{-2}$ m/s, N = $10^{-5.5}$ s^{-2}.

considering the effective thermal expansion coefficient α as a function of salinity S. We set $\alpha = 10^{-4}$ K^{-1} for salty water, which means the flow becomes unstably stratified and convection dominates the flow. We conduct simulation by spectral PDE solver - Dedalus [3] with a resolution of 2048×256. The control parameters are kept the same as in table 1.

First, we neglect the topography effect ($h(x) = 0$). The simulation results are shown in figure 7. As expected, the ocean stratification becomes unstable and convective flow dominates. Small-scale plumes emerge near the boundary and gradually dissipated towards the interior. The missing of large-scale circulation and plumes is due to the topographic β effect from equation 11-13. The magnitude of vertical velocity is $\sim 10^{-3}$ m/s, which is three orders larger than the results with fresh water. Therefore, in salty water, the convective flow will significantly enhance the ocean mixing and heat and mass transport.

Then, we consider the topography effect ($h(x) = 1$ km). The simulation results are shown in figure 8. With topography, the flow becomes spatially non-uniform due to the buoyancy variation at the top boundary. The flow is unstably stratified in the cold water (thick ice) region (middle part) while stably stratified in the warm water (thin ice) region (left and right part). The drifting of the convective plumes towards the west is due to the β term in equation 11-13. The maximum velocity magnitude shown in figure 8 is 5×10^{-3} m/s, even larger than the result without topography, which means the vertical transport can be more efficient but space-dependent.

Here we give a summary of this section. We investigated the effect of topography and salinity on the Enceladus ocean flow pattern. In fresh water, the flow is mainly driven by the top topography while stably stratified in the interior. We derived the linear solution and studied how the parameters (ν, κ, U, N) affect the heat flux, flow penetration depth, and vertical transport time scale. In salty water, the flow becomes convectively unstable and

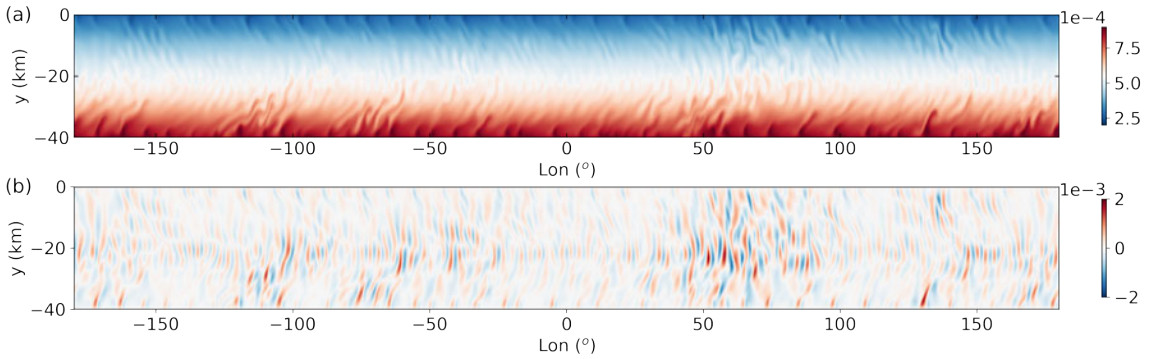


Figure 7: (a) Instantaneous temperature field with salty water after running for ~ 10 year, and no topography at top boundary. (b) Corresponding instantaneous vertical velocity field.

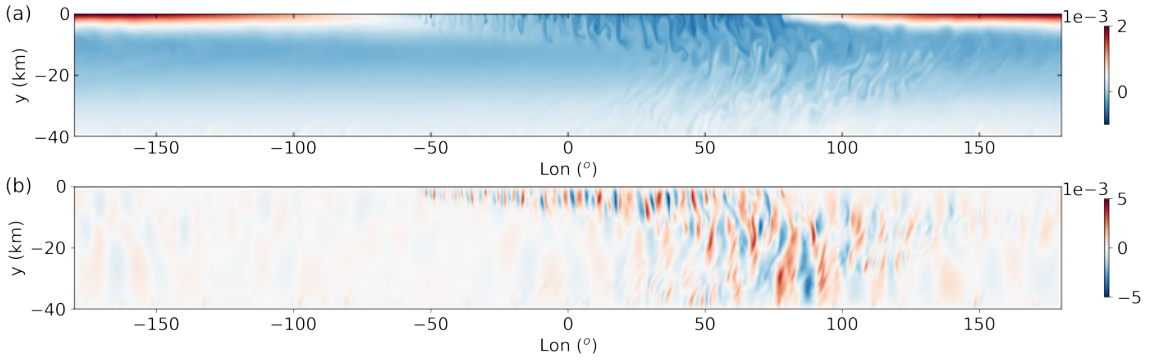


Figure 8: (a) Instantaneous temperature field with salty water after running for ~ 10 year, and with topography at top boundary. (b) Corresponding instantaneous vertical velocity field.

turbulent. Small-scale plumes emerge, which enhance ocean mixing and tracer transport efficiency. In salty water, topography results in a spatial non-uniform flow pattern - unstably stratified in thicker ice and stably stratified in thinner ice, which could be important for the distribution of tracers.

4 Discussion

In this section, we will study the heat budget and tracer transport with fresh water and salty water, which can be used as two constraints on the salinity in the Enceladus ocean.

4.1 Heat budget constraint of salinity

To ensure the ice topography can be sustained, the total heat budget for ice needs to be closed [21, 23]:

$$\mathcal{H}_{tidal} + \mathcal{H}_{ocean} + \mathcal{H}_{latent} - \mathcal{H}_{cond} = 0, \tag{36}$$

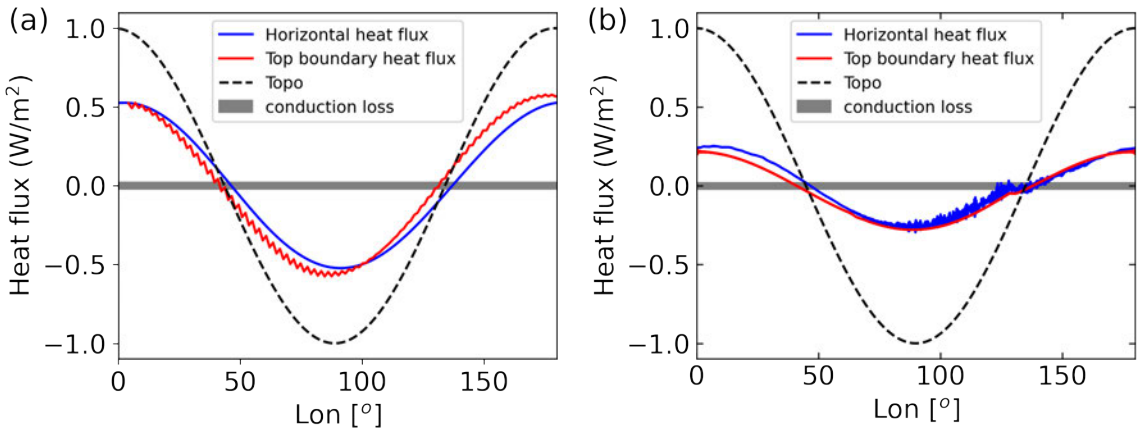


Figure 9: Time averaged heat fluxes along zonal direction for (a) fresh water and (b) salty water. The blue and red solid lines represent the interior heat flux and top boundary heat flux, respectively. The dashed line represents the shape of the topography. The gray area represents the value of heat loss from the conduction of ice.

where $\mathcal{H}_{tidal}$ is the tidal dissipation heating, $\mathcal{H}_{ocean}$ is the ocean heat flux into ice, $\mathcal{H}_{latent}$ is the heat flux due to latent heat considering ice flow, and $\mathcal{H}_{cond}$ is the conductive heat loss from the outer surface of the ice. This relation means that $\mathcal{H}_{ocean} < \mathcal{H}_{cond}$, otherwise the ice will keep melting due to a larger heat influx than outflux. $\mathcal{H}_{cond}$ is known as around 20 mW/m^2 from previous studies [23]. By comparing $\mathcal{H}_{ocean}$ from our simulations with $\mathcal{H}_{cond}$, we can see if the heat budget can be closed.

There are two ways to compute $\mathcal{H}_{ocean}$. One is to directly diagnose the heat flux at the top (denoted as $\mathcal{H}_t$, equation 31), and the other is to calculate the heat convergence in the interior plus the bottom heat flux (denoted as $\mathcal{H}_i$, equation 30). After the ocean reaches equilibrium, the two estimates should give the same answer.

$$\mathcal{H}_{ocean}(x) = \mathcal{H}_i(x) = \mathcal{H}_t(x) \tag{37}$$

We calculated time-averaged $\mathcal{H}_t(x)$ and $\mathcal{H}_i(x)$, and plot them for the case with fresh water in figure 9(a) and with salty water in figure 9(b). From the results, we see that the simulations have reached an equilibrium state since $\mathcal{H}_t(x)$ approximately matches with $\mathcal{H}_i(x)$ for both fresh water and salty water. However, the ocean heat flux for both freshwater ($\sim$ 500 mW/m^2) and salty water ($\sim$ 200 mW/m^2) are much larger than the conductive heat loss ($\sim$ 20 mW/m^2), which means in both cases the ice topography cannot be sustained but will keep melting due to larger heat influx from the ocean. Therefore, from the constraint of heat budget, we think that the salinity of the Enceladus ocean should be in an intermediate range, which means that the effective thermal expansion coefficient α would be close to 0, thus the ocean flow will be less convective and less heat flux will be generated from the ocean.

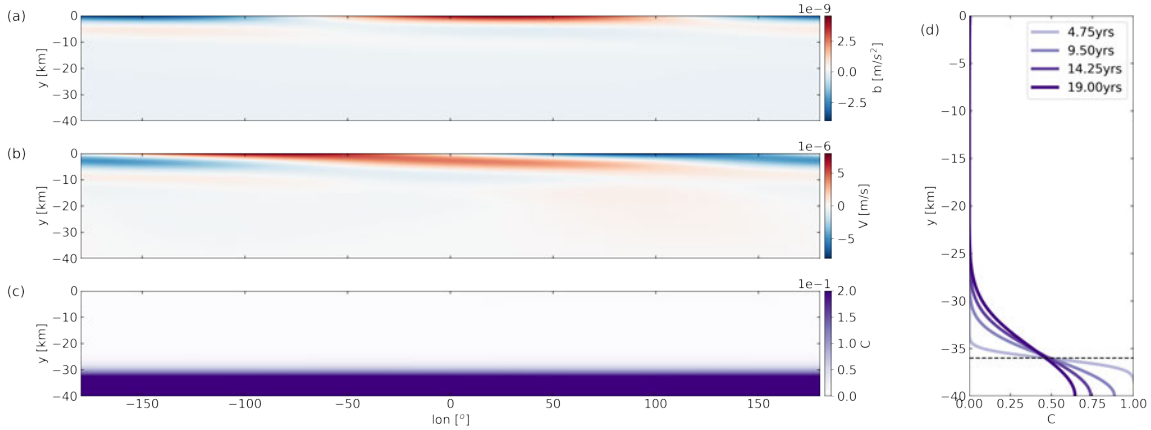


Figure 10: The instantaneous (a) buoyancy field, (b) velocity field, and (c) tracer concentration field at ~ 30 years for fresh water. (d) The vertical profile of tracer concentration at different times for freshwater. The dashed line shows the initial height of the tracer.

4.2 Tracer transport constraint of salinity

Another constraint of salinity is the tracer transport time scale. From the detected size of silica nanoparticles from the spray and the growth rate expected from Ostwald ripening, the vertical transport time scale has been estimated to be at most several years [14]. Here we estimate the vertical transport time scale t_0 by adding passive tracers into the bottom layer after the flow reaches the equilibrium state. The initial concentration of the tracer is restored to 1 at the bottom of the model. After 20-40 years, the tracer distributions and concentration profiles are shown in figure 10 and 11 for both fresh water and salty water. In freshwater, due to the stable stratification in the interior, the tracers are hardly transported by the flow but only by pure diffusion, see the flow and concentration snapshots in figure 10(a) and the concentration field in 10(b). Based on the velocity magnitude $\sim 10^{-6}$ m/s, the time needed to transport the tracers to the top is $\sim 10^3$ years. which is much larger than the expectation from [14].

While in salty water, stronger convection flow ($\sim 10^{-3}$ m/s) more efficiently transports tracers from the bottom to the top compared to in fresh water, estimated as ~ 1 year from the velocity magnitude. The enhancement of tracer transport can be observed from the concentration field in figure 11(a), where turbulent flow carries tracers upward. From the concentration profile shown in figure 11(d), the tracers can reach the top boundary in 20 years. The measured time from the profile is larger than the estimated ~ 1 year from the velocity magnitude since that tracers also move horizontally from the thick ice region to the thin ice region, which increases the total transport time scale. Therefore, from the constraint of the tracer transport time scale, the salinity of the Enceladus ocean should be even higher than the value we applied in our simulation so that the time scale can be even smaller. Higher salinity (larger $\alpha(S)$) results in more unstably stratification, stronger convective flow can transport tracers more efficiently.

Based on the constraint from heat budget and tracer transport, we get two different ranges of reasonable salinity for the Enceladus ocean without overlapping. To close the

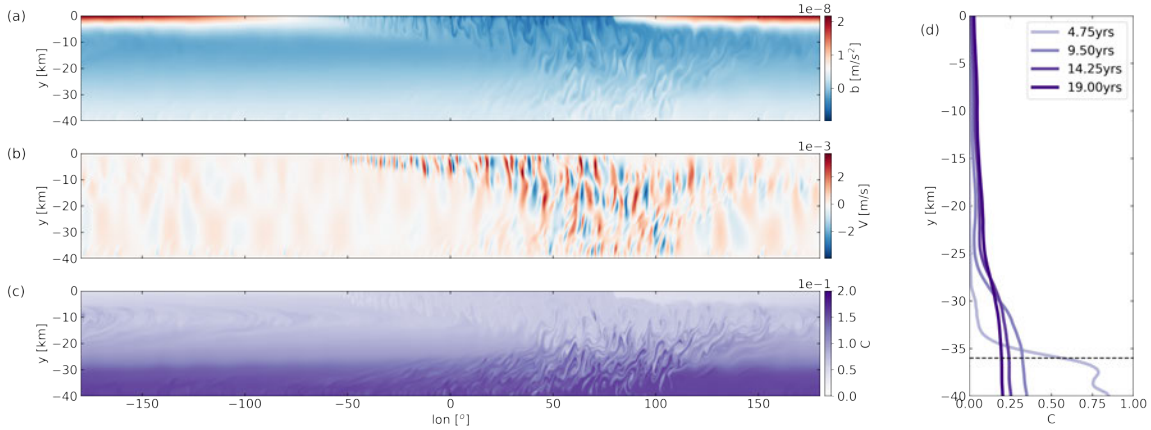


Figure 11: The instantaneous (a) buoyancy field, (b) velocity field, and (c) tracer concentration field at ~ 40 years for salty water. (d) The vertical profile of tracer concentration at different times. The dashed line shows the initial height of the tracer.

heat budget, the salinity should be intermediate so that heat flux from the ocean is small enough to balance with the conductive heat loss. While to satisfy the tracer transport time scale, the salinity should be high enough to efficiently transport the tracers in years. There are many possible reasons to explain this inconsistency:

First, in our model, we assumed two-dimensional flow due to fast rotation, and only equatorial flow is considered. However, the meridional circulation might also play an important role in the equatorial flow. [21]

Second, our choice of $\nu = \kappa = 0.1$ is limited by the computation power. The realistic values of ν and κ are $\sim 10^{-6}$. From our results of linear solution in figure 3(a), decreasing ν and κ will decrease the ocean heat flux. Thus the realistic ocean heat flux could be able to balance with the conductive heat loss.

Third, our consideration of salinity is simplified, while the actual salinity effect could be more complicated [38, 5].

Last but not least, we ignored the non-uniform boundary conditions at the seafloor of Enceladus, such as rough topography, localized heating, and vent or volcanic eruptions [24]. These factors could enhance the mixing at the bottom and drive tracers towards the surface in a short time.

5 Conclusion

In conclusion, we theoretically and numerically investigated the equatorial flow of the Enceladus ocean with the ice topography. The key question motivating our study is what are the conditions most likely on Enceladus?

To answer it, we build a simplified 2D model to describe the equatorial flow. Based on the linear solution, we investigated the effect of system parameters (ν, κ, U, N) on the heat budget, flow penetration depth, and tracer transport time scale. We further run simulations for salty water and also add passive tracers to study the transport time scale in salty water.

Different ocean flow patterns are observed for fresh water and salty water. In fresh water, the flow is mainly driven by the ice topography at the top boundary and decays downward due to the stable stratification. In salty water, the flow is globally convective due to unstable stratification. Topography-induced buoyancy variation at the top boundary also results in spatially non-uniform flows.

We investigated how heat budget and tracer transport time scale constraint the salinity of the Enceladus ocean, which is inconsistent with each other from our results. Many processes are absent from our study, and we give some possible reasons to explain this discrepancy. However, more work is needed to fill the gap between these two constraints.

Acknowledgments

I would sincerely like to thank Wanying Kang, Glenn Flierl for their time and guidance through this project. I would also like to thank Keaton Burns for the help in Dedalus code. I also sincerely thank Colm-cille Caulfield and Stefan Llewellyn Smith for organizing the summer school, and my fellow fellows for their companionship.

Appendix - Higher order expansion

Following the linear solution, we continue to expand to the first order. We have first-order equations

$$\begin{aligned} U\frac{\partial \zeta_1}{\partial x} &= \beta\frac{\partial \psi_1'}{\partial x} + \frac{\partial b_1'}{\partial x} + \nu\nabla^2\zeta_1 + J(\psi_0', \zeta_0) \\ \nabla^2\psi_1' &= \zeta_1 \\ U\frac{\partial b_1'}{\partial x} + N^2\frac{\partial \psi_1'}{\partial x} &= \kappa\nabla^2 b_1' + J(\psi_0', b_0') \end{aligned} \tag{38}$$

with the boundary conditions

$$\begin{aligned} b_1' &= 0,\ y = 0 \\ b_{1y}' &= 0,\ y = -H \\ \psi_1' &= 0,\ y = -H, 0 \\ {}'_{1yy} &= 0,\ y = -H, 0 \end{aligned} \tag{39}$$

The Jacobian matrixes give

$$
\begin{aligned}
J(\psi_0', b_0') &= \frac{\partial \psi_0'}{\partial x}\frac{\partial b_0'}{\partial y} - \frac{\partial \psi_0'}{\partial y}\frac{\partial b_0'}{\partial x} \\
&= \left[ik\sum_{n=1}^{6} B_n^0 e^{s_n y} e^{ikx} - ik\sum_{n=1}^{6} \widetilde{B}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \left[\sum_{n=1}^{6} s_n A_n^0 e^{s_n y} e^{ikx} + \sum_{n=1}^{6} \widetilde{s}_n \widetilde{A}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \\
&- \left[\sum_{n=1}^{6} s_n B_n^0 e^{s_n y} e^{ikx} + \sum_{n=1}^{6} \widetilde{s}_n \widetilde{B}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \left[\sum_{n=1}^{6} ik A_n^0 e^{s_n y} e^{ikx} - \sum_{n=1}^{6} ik \widetilde{A}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \\
&= \sum_{n=1}^{6}\sum_{m=1}^{6} D_{+,n,m}^1 e^{(s_n+s_m)y} e^{2ikx} + \sum_{n=1}^{6}\sum_{m=1}^{6} D_{o,n,m}^1 e^{(s_n+\widetilde{s}_m)y} + \sum_{n=1}^{6}\sum_{m=1}^{6} D_{-,n,m}^1 e^{(\widetilde{s}_n+\widetilde{s}_m)y} e^{-2ikx}
\end{aligned}
\tag{40}
$$

$$
\begin{aligned}
D_{+,n,m}^1 &= ik(s_m A_m^0 B_n^0 + s_n A_n^0 B_m^0 - s_n A_m^0 B_n^0 - s_m A_n^0 B_m^0) \ (m \geq n) \\
D_{o,n,m}^1 &= ik(\widetilde{s}_m \widetilde{A}_m^0 B_n^0 - s_n A_n^0 \widetilde{B}_m^0 + s_n \widetilde{A}_m^0 B_n^0 - \widetilde{s}_m A_n \widetilde{B}_m^0) \\
D_{-,n,m}^1 &= ik(-\widetilde{s}_m \widetilde{A}_m^0 \widetilde{B}_n^0 - \widetilde{s}_n \widetilde{A}_n^0 \widetilde{B}_m^0 + \widetilde{s}_n \widetilde{A}_m^0 \widetilde{B}_n^0 + \widetilde{s}_m \widetilde{A}_n^0 \widetilde{B}_m^0) \ (m \geq n)
\end{aligned}
\tag{41}
$$

$$
\begin{aligned}
J(\psi_0', \zeta_0) &= \frac{\partial \psi_0'}{\partial x}\frac{\partial \zeta_0}{\partial y} - \frac{\partial \psi_0'}{\partial y}\frac{\partial \zeta_0}{\partial x} \\
&= \left[ik\sum_{n=1}^{6} B_n^0 e^{s_n y} e^{ikx} - ik\sum_{n=1}^{6} \widetilde{B}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \left[\sum_{n=1}^{6} s_n C_n^0 e^{s_n y} e^{ikx} + \sum_{n=1}^{6} \widetilde{s}_n \widetilde{C}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \\
&- \left[\sum_{n=1}^{6} s_n B_n^0 e^{s_n y} e^{ikx} + \sum_{n=1}^{6} \widetilde{s}_n \widetilde{B}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \left[\sum_{n=1}^{6} ik C_n^0 e^{s_n y} e^{ikx} - \sum_{n=1}^{6} ik \widetilde{C}_n^0 e^{\widetilde{s}_n y} e^{-ikx} \right] \\
&= \sum_{n=1}^{6}\sum_{m=1}^{6} E_{+,n,m}^1 e^{(s_n+s_m)y} e^{2ikx} + \sum_{n=1}^{6}\sum_{m=1}^{6} E_{o,n,m}^1 e^{(s_n+\widetilde{s}_m)y} + \sum_{n=1}^{6}\sum_{m=1}^{6} E_{-,n,m}^1 e^{(\widetilde{s}_n+\widetilde{s}_m)y} e^{-2ikx}
\end{aligned}
\tag{42}
$$

$$
\begin{aligned}
E_{+,n,m}^1 &= ik(s_m C_m^0 B_n^0 + s_n C_n^0 B_m^0 - s_n C_m^0 B_n^0 - s_m C_n^0 B_m^0) \ (m \geq n) \\
E_{o,n,m}^1 &= ik(\widetilde{s}_m \widetilde{C}_m^0 B_n^0 - s_n C_n^0 \widetilde{B}_m^0 + s_n \widetilde{C}_m^0 B_n^0 - \widetilde{s}_m C_n^0 \widetilde{B}_m^0) \\
E_{-,n,m}^1 &= ik(-\widetilde{s}_m \widetilde{C}_m^0 \widetilde{B}_n^0 - \widetilde{s}_n \widetilde{C}_n^0 \widetilde{B}_m^0 + \widetilde{s}_n \widetilde{C}_m^0 \widetilde{B}_n^0 + \widetilde{s}_m \widetilde{C}_n^0 \widetilde{B}_m^0) \ (m \geq n)
\end{aligned}
\tag{43}
$$

We set

$$
\begin{aligned}
b_1' =& \sum_{n=1}^{6} A^1_{+,n} e^{s_{+,n} y} e^{2ikx} + \sum_{n=1}^{6} F_{An}(y) + \sum_{n=1}^{6} A^1_{-,n} e^{s_{-,n} y} e^{-2ikx} \\
&+ \sum_{n=1}^{6}\sum_{m=1}^{6} A^1_{+,n,m} e^{(s_n+s_m)y} e^{2ikx} + \sum_{n=1}^{6}\sum_{m=1}^{6} A^1_{o,n,m} e^{(s_n+\widetilde{s}_m)y} + \sum_{n=1}^{6}\sum_{m=1}^{6} A^1_{-,n,m} e^{(\widetilde{s}_n+\widetilde{s}_m)y} e^{-2ikx} \\
{}'_1 =& \sum_{n=1}^{6} B^1_{+,n} e^{s_{+,n} y} e^{2ikx} + \sum_{n=1}^{6} F_{Bn}(y) + \sum_{n=1}^{6} B^1_{-,n} e^{s_{-,n} y} e^{-2ikx} \\
&+ \sum_{n=1}^{6}\sum_{m=1}^{6} B^1_{+,n,m} e^{(s_n+s_m)y} e^{2ikx} + \sum_{n=1}^{6}\sum_{m=1}^{6} B^1_{o,n,m} e^{(s_n+\widetilde{s}_m)y} + \sum_{n=1}^{6}\sum_{m=1}^{6} B^1_{-,n,m} e^{(\widetilde{s}_n+\widetilde{s}_m)y} e^{-2ikx} \\
\zeta_1 =& \sum_{n=1}^{6} C^1_{+,n} e^{s_{+,n} y} e^{2ikx} + \sum_{n=1}^{6} F_{Cn}(y) + \sum_{n=1}^{6} C^1_{-,n} e^{s_{-,n} y} e^{-2ikx} \\
&+ \sum_{n=1}^{6}\sum_{m=1}^{6} C^1_{+,n,m} e^{(s_n+s_m)y} e^{2ikx} + \sum_{n=1}^{6}\sum_{m=1}^{6} C^1_{o,n,m} e^{(s_n+\widetilde{s}_m)y} + \sum_{n=1}^{6}\sum_{m=1}^{6} C^1_{-,n,m} e^{(\widetilde{s}_n+\widetilde{s}_m)y} e^{-2ikx}
\end{aligned}
\tag{44}
$$

Substitute to 38, we obtain

$$
\begin{aligned}
2ikUC^1_{+,n,m} - 2ik\beta B^1_{+,n,m} - 2ikA^1_{+,n,m} - \nu((s_n+s_m)^2 - 4k^2)C^1_{+,n,m} &= E^1_{+,n,m} \\
-\nu(s_n+\widetilde{s}_m)^2 C^1_{o,n,m} &= E^1_{o,n,m} \\
-2ikUC^1_{-,n,m} + 2ik\beta B^1_{-,n,m} + 2ikA^1_{-,n,m} - \nu((\widetilde{s}_n+\widetilde{s}_m)^2 - 4k^2)C^1_{-,n,m} &= E^1_{-,n,m} \\
((s_n+s_m)^2 - 4k^2)B^1_{+,n,m} &= C^1_{+,n,m} \\
(s_n+\widetilde{s}_m)^2 B^1_{o,n,m} &= C^1_{o,n,m} \\
((\widetilde{s}_n+\widetilde{s}_m)^2 - 4k^2)B^1_{-,n,m} &= C^1_{-,n,m} \\
2ikUA^1_{+,n,m} + 2ikN^2 B^1_{+,n,m} - \kappa((s_n+s_m)^2 - 4k^2)A^1_{+,n,m} &= D^1_{+,n,m} \\
-\kappa(s_n+\widetilde{s}_m)^2 A^1_{o,n,m} &= D^1_{o,n,m} \\
-2ikUA^1_{-,n,m} - 2ikN^2 B^1_{-,n,m} - \kappa((\widetilde{s}_n+\widetilde{s}_m)^2 - 4k^2)A^1_{-,n,m} &= D^1_{-,n,m}
\end{aligned}
\tag{45}
$$

From these relations, we can solve out $A^1_{*,n,m}$, $B^1_{*,n,m}$, and $C^1_{*,n,m}$.

$s_{+,n}$, $s_{-,n}$, $F_{An}(y)$, $F_{Bn}(y)$, and $F_{Cn}(y)$ can be obtained by substituting the solution into governing equation.

Finally by satisfying the solution 44 with the boundary conditions in 39, we can obtain $A^1_{*,n}$, $B^1_{*,n}$ and $C^1_{*,n}$. So that we obtain b_1', ψ_1', and ζ_1.

Following the same way, one can derive the second order equation and solve out b_2', ψ_2', and ζ_2, and so on. In figure 5, we present the result of each order solution and summation of them. One can see that the second-order solution is already two orders smaller than the linear solution.

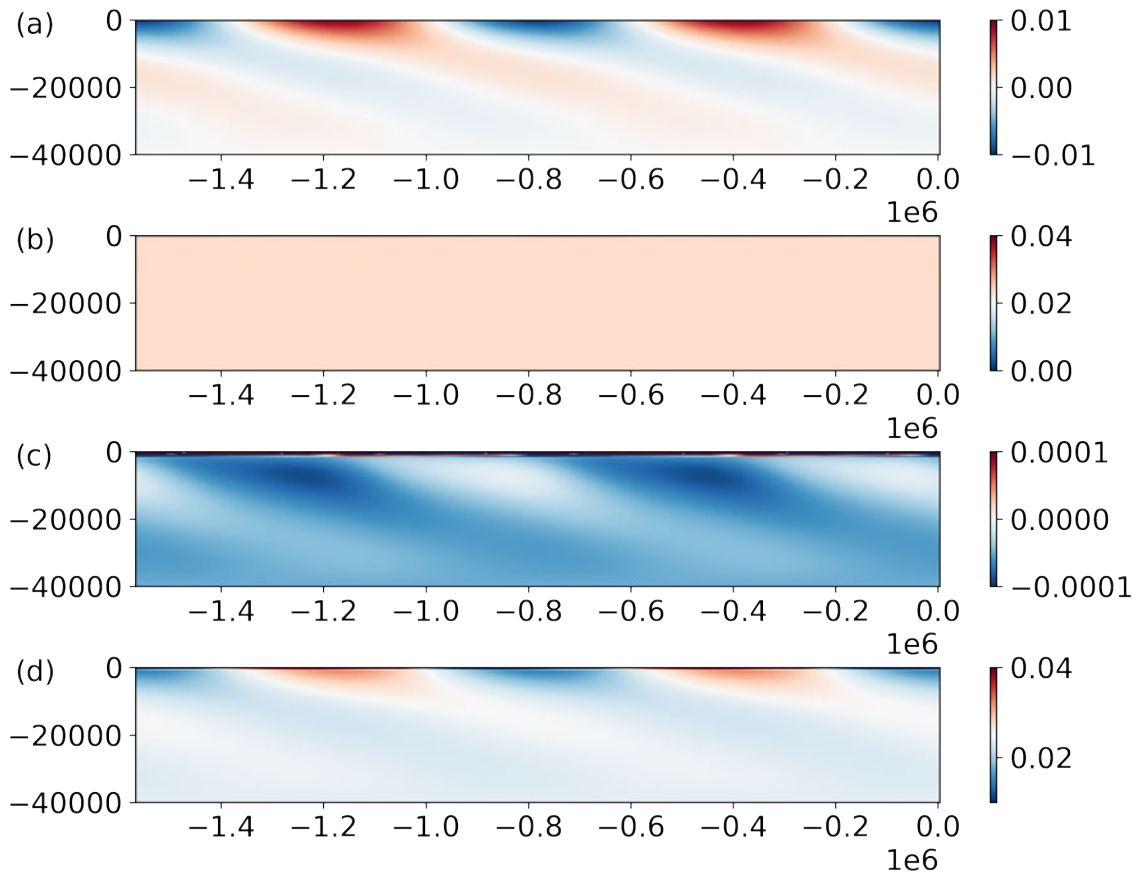


Figure 12: Temperature field from the solution of different orders. (a) 0th order (linear) solution. (b) 1st order solution. (c) 2nd order solution. (d) The summation of 0th to 2nd order solutions.

References

[1] Y. Ashkenazy, R. Sayag, and E. Tziperman, *Dynamics of the global meridional ice flow of Europa's icy shell*, Nature Astronomy, 2 (2018), pp. 43–49.

[2] M. Beuthe, *Enceladus's crust as a non-uniform thin shell: Ii tidal dissipation*, Icarus, 332 (2019), pp. 66 – 91.

[3] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, *Dedalus: A flexible framework for numerical simulations with spectral methods*, Phys. Rev. Res, 2 (2020), p. 023068.

[4] G. Choblet, G. Tobie, C. Sotin, M. Běhounková, O. Čadek, F. Postberg, and O. Souček, *Powering prolonged hydrothermal activity inside Enceladus*, Nature Astronomy, 1 (2017), pp. 841–847.

[5] P. Garaud, *Double-diffusive convection at low prandtl number*, Annu. Rev. Fluid Mech., 50 (2018), pp. 275–298.

[6] T. Gastine, J. Wicht, and J. Aubert, *Scaling regimes in spherical shell rotating convection*, Journal of Fluid Mechanics, 808 (2016), pp. 690–732.

[7] J. Geisler, *A linear model of the walker circulation*, Journal of Atmospheric Sciences, 38 (1981), pp. 1390–1400.

[8] P. R. Gent and J. C. Mcwilliams, *Isopycnal mixing in ocean circulation models*, Journal of Physical Oceanography, 20 (1990), pp. 150–155.

[9] C. Glein, F. Postberg, and S. Vance, *The geochemistry of enceladus: Composition and controls*, Enceladus and the icy moons of Saturn, 39 (2018).

[10] C. R. Glein, J. A. Baross, and J. H. Waite Jr, *The ph of enceladus' ocean*, Geochimica et Cosmochimica Acta, 162 (2015), pp. 202–219.

[11] D. J. Hemingway and T. Mittal, *Enceladus's ice shell structure as a window on internal heat production*, Icarus, 332 (2019), pp. 111–131.

[12] C. J. A. Howett, J. R. Spencer, J. Pearl, and M. Segura, *High heat flow from Enceladus' south polar region measured using 10-600 cm-1 Cassini/CIRS data*, Journal of Geophysical Research-Atmospheres, 116 (2011), p. 189.

[13] H.-W. Hsu, F. Postberg, Y. Sekine, T. Shibuya, S. Kempf, M. Horányi, A. Juhász, N. Altobelli, K. Suzuki, Y. Masaki, et al., *Ongoing hydrothermal activities within enceladus*, Nature, 519 (2015), pp. 207–210.

[14] ——, *Ongoing hydrothermal activities within enceladus*, Nature, 519 (2015), pp. 207–210.

[15] H. E. Huppert and J. S. Turner, *Double-diffusive convection*, J. Fluid Mech., 106 (1981), pp. 299–329.

[16] L. Iess, D. J. Stevenson, M. Parisi, D. Hemingway, R. A. Jacobson, J. I. Lunine, F. Nimmo, J. W. Armstrong, S. W. Asmar, M. Ducci, and P. Tortora, *The Gravity Field and Interior Structure of Enceladus*, Science, 344 (2014), pp. 78–80.

[17] A. P. Ingersoll and M. Nakajima, *Controlled boiling on enceladus. 2. model of the liquid-filled cracks*, Icarus, 272 (2016), pp. 319–326.

[18] M. F. Jansen, W. Kang, and E. Kite, *Energetics govern ocean circulation on icy ocean worlds*, 2022.

[19] H. Jones and J. Marshall, *Convection with Rotation in a Neutral Ocean: A Study of Open-Ocean Deep Convection*, J. Phys. Oceanogr., 23 (1993), pp. 1009–1039.

[20] W. Kang, *Different ice-shell geometries on europa and enceladus due to their different sizes: Impacts of ocean heat transport*, The Astrophysical Journal, 934 (2022), p. 116.

[21] W. Kang and G. Flierl, *Spontaneous formation of geysers at only one pole on enceladus's ice shell*, Proc. Natl. Acad. Sci. U.S.A., 117 (2020), pp. 14764–14768.

[22] W. Kang and M. Jansen, *On icy ocean worlds, size controls ice shell geometry*, ApJ, accepted, (2022).

[23] W. Kang, T. Mittal, S. Bire, J.-M. Campin, and J. Marshall, *How does salinity shape ocean circulation and ice geometry on enceladus and other icy satellites?*, Sci. Adv., 8 (2022), p. eabm4665.

[24] J. S. Kargel and S. Pozio, *The volcanic and tectonic history of enceladus*, Icarus, 119 (1996), pp. 385–404.

[25] E. S. Kite and A. M. Rubin, *Sustained eruptions on enceladus explained by turbulent dissipation in tiger stripes*, Proceedings of the National Academy of Sciences, 113 (2016), pp. 3972–3975.

[26] K. Lau and S. Yang, *Walker circulation*, Encyclopedia of atmospheric sciences, 2505 (2003), p. 2510.

[27] Y. Liao, F. Nimmo, and J. A. Neufeld, *Heat production and tidally driven fluid flow in the permeable core of enceladus*, Journal of Geophysical Research: Planets, 125 (2020), p. e2019JE006209.

[28] J. Marshall, A. Adcroft, C. Hill, L. Perelman, and C. Heisey, *A finite-volume, incompressible Navier Stokes model for studies of the ocean on parallel computers*, J. Geophys. Res., 102 (1997), pp. 5,753–5,766.

[29] T. J. McDougall and P. M. Barker, *Getting started with teos-10 and the gibbs seawater (gsw) oceanographic toolbox*, SCOR/IAPSO WG, 127 (2011), pp. 1–28.

[30] C. McKay, A. Davila, C. Glein, K. Hand, and A. Stockton, *Enceladus astrobiology, habitability, and the origin of life*, Enceladus and the Icy Moons of Saturn; Schenk, PM, Clark, RN, Howett, CJA, Verbiscer, AJ, Waite, JH, Eds, (2018), pp. 437–452.

[31] MITgcm-group, *MITgcm User Manual*, online documentation, MIT/EAPS, Cambridge, MA 02139, USA, 2010. http://mitgcm.org/public/r2_manual/latest/online_documents/manual.html.

[32] F. Nimmo, J. Spencer, R. Pappalardo, and M. Mullen, *Shear heating as the origin of the plumes and heat flux on enceladus*, Nature, 447 (2007), pp. 289–291.

[33] V. Petrenko and R. Whitworth, *Physics of Ice*, OUP Oxford, 1999.

[34] K. Pleiner Sládková, O. Souček, and M. Běhounková, *Enceladus' tiger stripes as frictional faults: Effect on stress and heat production*, Geophysical Research Letters, 48 (2021), p. e2021GL094849.

[35] C. C. Porco, P. Helfenstein, P. Thomas, A. Ingersoll, J. Wisdom, R. West, G. Neukum, T. Denk, R. Wagner, T. Roatsch, et al., *Cassini observes the active south pole of enceladus*, science, 311 (2006), pp. 1393–1401.

[36] F. Postberg, S. Kempf, J. Schmidt, N. Brilliantov, A. Beinsen, B. Abel, U. Buck, and R. Srama, *Sodium salts in e-ring ice grains from an ocean below the surface of enceladus*, Nature, 459 (2009), pp. 1098–1101.

[37] F. Postberg, N. Khawaja, B. Abel, G. Choblet, C. R. Glein, M. S. Gudipati, B. L. Henderson, H.-W. Hsu, S. Kempf, F. Klenner, et al., *Macromolecular organic compounds from the depths of enceladus*, Nature, 558 (2018), pp. 564–568.

[38] T. Radko, *Double-diffusive convection*, Cambridge University Press, 2013.

[39] M. H. Redi, *Oceanic isopycnal mixing by coordinate rotation*, J. Phys. Oceanogr., 12 (1982), pp. 1154–1158.

[40] K. Soderlund, B. Schmidt, J. Wicht, and D. Blankenship, *Ocean-driven heating of europa's icy shell at low latitudes*, Nat. Geosci., 7 (2014), pp. 16–19.

[41] K. M. Soderlund, *Ocean Dynamics of Outer Solar System Satellites*, Geophysical Research Letters, 46 (2019), pp. 8700–8710.

[42] O. Soucek, M. Behounkova, O. Cadek, J. Hron, G. Tobie, and G. Choblet, *Tidal dissipation in Enceladus' uneven, fractured ice shell*, Icarus, 328 (2019), pp. 218–231.

[43] J. R. Spencer, C. J. A. Howett, A. Verbiscer, T. A. Hurford, M. Segura, and D. C. Spencer, *Enceladus Heat Flow from High Spatial Resolution Thermal Emission Observations*, European Planetary Science Congress, 8 (2013), pp. EPSC2013–840.

[44] R.-S. Taubner, P. Pappenreiter, J. Zwicker, D. Smrzka, C. Pruckner, P. Kolar, S. Bernacchi, A. H. Seifert, A. Krajete, W. Bach, et al., *Biological methane production under putative enceladus-like conditions*, Nature communications, 9 (2018), pp. 1–11.

[45] P. Thomas, R. Tajeddine, M. Tiscareno, J. Burns, J. Joseph, T. Loredo, P. Helfenstein, and C. Porco, *Enceladus's measured physical libration requires a global subsurface ocean*, Icarus, 264 (2016), pp. 37–47.

[46] B. J. Travis and G. Schubert, *Keeping Enceladus warm*, Icarus, 250 (2015), pp. 32–42.

[47] J. H. Waite, C. R. Glein, R. S. Perryman, B. D. Teolis, B. A. Magee, G. Miller, J. Grimes, M. E. Perry, K. E. Miller, A. Bouquet, J. I. Lunine, T. Brockwell, and S. J. Bolton, *Cassini finds molecular hydrogen in the Enceladus plume: Evidence for hydrothermal processes*, Science, 356 (2017), pp. 155–159.

Stochasticity of Turbulence Closures

Iury Simoes-Sousa

March 27, 2023

1 Motivation

Turbulence is a fascinating natural phenomenon that is characterized by its chaotic behavior. It is a product of the nonlinear dynamics of fluid motion across scales, which causes the merging and splitting of fluid vortices and leads to a cascade of energy towards smaller scales until dissipation occurs. Visualizing the turbulence cascade helps to understand how different scales of motion interact with each other, which is a critical aspect of turbulence. However, simulating turbulence accurately is challenging because it requires solving the equations of motion across all scales, the so-called Direct Numerical Simulations (DNS). The expectation expressed in the early 70s about DNS was that it was a promising approach for studying the large-scale features of turbulence [12]. The idea was to use DNS to simulate the entire range of scales of the turbulence, without the need for a subgrid-scale model. Despite the wide application of DNS over the years, it still remains computationally expensive, and the computational cost grows rapidly with the size of the domain, which limits its applicability to larger-scale problems. Therefore, for most atmospheric and oceanography questions, DNS is not practical, and instead, models are run on large grid boxes with most of the turbulence parameterized by closures [9, 11, 4].

Parameterizations, or subgrid-scale models, are an essential tool for simulating turbulence in coarser grids. They allow us to account for the effects of the unresolved scales on the resolved scales by approximating the interactions that take place at those smaller scales. The closure models are usually based on physical principles and aim to reproduce the effects of the unresolved scales on the resolved scales accurately. These models have been developed and improved over the years, and they are an essential part of most numerical models used in atmospheric and oceanic simulations [4].

In atmospheric sciences and oceanography, turbulence parameterizations are especially important as they are an essential tool for representing the effect of subgrid scales on mixing. For instance, the mixing of water masses is a crucial process that affects the ocean's temperature, salinity, and density structure, which, in turn, can influence ocean circulation and climate [6]. Therefore, accurate representation of turbulent mixing in ocean models is essential for understanding and predicting the behavior of the ocean [4].

2 Introduction

The Navier-Stokes equations describe the motion of fluids and provide a mathematical framework for analyzing their behavior. One of the most challenging aspects of the Navier-Stokes equation is its non-linearity, which represents the interactions between fluid particles and can cause turbulence and chaotic behavior. For most of the ocean scales, for example, we can assume that the fluid is incompressible ($\nabla \cdot \mathbf{u} = 0$) and the momentum conservation from Navier-Stokes is expressed as:

$$\overbrace{\partial_t \mathbf{u}}^{\text{Tendency}} = -\underbrace{\mathbf{u}\cdot\nabla\mathbf{u}}_{\text{Advection}} - \overbrace{(1/\rho_0)\nabla p}^{\text{Pressure}} + \underbrace{\nu\nabla^2\mathbf{u}}_{\text{Friction}} + \overbrace{\rho'\mathbf{g}}^{\text{Gravity}} + \underbrace{\mathbf{F}_u}_{\text{Forcing}}, \tag{1}$$

Here, $\partial_t \mathbf{u}$ represents the time derivative of the fluid velocity vector $\mathbf{u}$, also known as the tendency term. $\mathbf{u}\cdot\nabla\mathbf{u}$, represents the effect of the fluid's motion on itself. The term $(1/\rho_0)\nabla p$ represents the gradient of pressure within the fluid and $\nu\nabla^2\mathbf{u}$, known as friction, represents the fluid's internal resistance to motion, where ν is the kinematic viscosity coefficient. The term $\rho'\mathbf{g}$ represents the effect of gravitational force on the fluid, where ρ is the fluid density, and $\mathbf{g}$ is the acceleration due to gravity. Lastly, $\mathbf{F}_u$ represents any external forcing acting on the fluid.

Assuming that the external forcing is linear, all terms in the equation are linear except for the advection term. This term represents the interaction between different scales of motion, which is responsible for turbulence. The nonlinear nature of this term makes it extremely difficult to properly resolve the discrete version of the equation in coarser grids, such as those used in Large Eddy Simulations (LES). As previously explained, parameterizations of the subgrid-stress are then needed to accurately account for the effects of the unresolved scales on the resolved scales. These parameterizations express the effect of the smaller and unresolved scales on the larger scales because the advection term cannot be accurately calculated at the LES resolution.

Stepping back to a more general mathematical description, we can simplify by starting with a DNS model in the form of

$$\partial_t \mathcal{X} = \mathcal{L}\cdot\mathcal{X} + \mathcal{N}(\mathcal{X}), \tag{2}$$

where all linear terms are expressed by $\mathcal{L}$ and the nonlinear term is in $\mathcal{N}$. The state variable is expressed by $\mathcal{X}$. If we filter and coarse-grain in space to some larger scale $\overline{(\cdot)}$, we obtain

$$\overline{\partial_t \mathcal{X}} = \overline{\mathcal{L}\cdot\mathcal{X}} + \overline{\mathcal{N}(\mathcal{X})}. \tag{3}$$

The commonly-used filters in LES are convolution linear operators (e.g. Gaussian filter) and thus, they commute with differentiation. Based on that, we can replace all averages over linear terms in Equation 3 by the linear operators applied to the coarse-grained (LES) fields:

$$\partial_t \overline{\mathcal{X}} = \mathcal{L}\cdot\overline{\mathcal{X}} + \overline{\mathcal{N}(\mathcal{X})}. \tag{4}$$

We recall that the main idea in this process is to find an equation that depends on $\overline{\mathcal{X}}$ only, which are the coarse-resolution fields. If we sum and subtract an arbitrary nonlinear function applied to the coarse-grained field, we obtain

$$\partial_t \overline{\mathcal{X}} = \mathcal{L}\cdot\overline{\mathcal{X}} + \mathcal{G}(\overline{\mathcal{X}}) + \overline{\mathcal{N}(\mathcal{X})} - \mathcal{G}(\overline{\mathcal{X}}), \tag{5}$$

which can be approximated to the closure form of

$$\partial_t \overline{\mathcal{X}} - \mathcal{L}\cdot\overline{\mathcal{X}} = \mathcal{G}(\overline{\mathcal{X}}) + \mathcal{C}_{\mathcal{G}}(\overline{\mathcal{X}}), \tag{6}$$

where $\mathcal{C}_{\mathcal{G}}(\overline{\mathcal{X}})$ is the closure that depends on $\mathcal{G}$ and parameterize $\overline{\mathcal{N}(\mathcal{X})} - \mathcal{G}(\overline{\mathcal{X}})$. There are two main choices for $\mathcal{G}(\overline{\mathcal{X}})$:

"Implicitly filtered closure"

$$\mathcal{G}(\overline{\mathcal{X}}) = \mathcal{N}(\overline{\mathcal{X}})$$

For this option, we apply the filter only to $\mathcal{X}$ and not to $\mathcal{N}(\mathcal{X})$. This is easier to do, but the problem is that the nonlinear operator may leak energy to larger wavenumbers ($e^{i\mathbf{k}x}e^{i\mathbf{k}x} = e^{i\mathbf{2k}x}$), that will simply cut off by the limited resolution of the LES.

"Explicitly filtered closure"

$$\mathcal{G}(\overline{\mathcal{X}}) = \overline{\mathcal{N}(\overline{\mathcal{X}})}$$

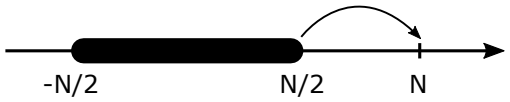
For this option we apply the filter twice. First to $\mathcal{X}$ and then to $\mathcal{N}(\overline{\mathcal{X}})$, that guarantees that we have the same number of wavenumbers for all terms in the LES equation (Equation 6)

The implicit representation assumes that the product of the filtered fields constrain the energy at the resolved scales, which is not necessarily true. So for the explicit representation, we filter not only the fields but also the nonlinear term derived from the filtered fields, which guarantees that the closure model would train with the solution and subgrid stress constrained at the resolved scales.

The argument around the explicit representation can be explained by using the pure spectral method. Let us suppose we have some nonlinear term as fg, for $k = N/2$ in Fourier basis

$$f\left(\tfrac{N}{2}\right)\, g\left(\tfrac{N}{2}\right) \Rightarrow \left(\hat{f}e^{i\frac{N}{2}x}\right)\, \left(\hat{g}e^{i\frac{N}{2}x}\right) = \hat{f}\hat{g}e^{iNx}, \tag{7}$$

which means that the nonlinear terms leak energy to $k = N$, which is not resolved by our system. We here chose to approximate functions between $-N/2$ and $N/2$.



Now, in the context of fluids, the low-pass filtered Navier-Stokes equation with turbulence closure is frequently called Large-Eddy Simulation (LES). In that case

$$\mathcal{N}(\mathcal{X}) = \nabla \cdot (\mathbf{u} \otimes \mathbf{u}), \tag{8}$$

which means that,

$$\mathcal{C}(\overline{\mathcal{X}}) \approx \overline{\mathcal{N}(\mathcal{X})} - \mathcal{N}(\overline{\mathcal{X}}) = \overline{\nabla \cdot (\mathbf{u} \otimes \mathbf{u})} - \nabla \cdot (\overline{\mathbf{u}} \otimes \overline{\mathbf{u}}), \tag{9}$$

that can be simplified to

$$\mathcal{C}(\overline{\mathcal{X}}) \approx \nabla \cdot \underbrace{(\overline{\mathbf{u} \otimes \mathbf{u}} - \overline{\mathbf{u}} \otimes \overline{\mathbf{u}})}_{\tau_s}, \text{ for } \textbf{implicitly} \text{ filtered} \tag{10a}$$

and

$$\mathcal{C}(\overline{\mathcal{X}}) \approx \nabla \cdot \underbrace{(\overline{\mathbf{u} \otimes \mathbf{u}} - \overline{\overline{\mathbf{u}} \otimes \overline{\mathbf{u}}})}_{\tau_s}, \text{ for } \textbf{explicitly} \text{ filtered.} \tag{10b}$$

τ_s is the subgrid-scale stress tensor, which is symmetric and defined by

$$\tau_s^{3D} = \begin{bmatrix} \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \end{bmatrix}. \tag{11}$$

Variables in scientific processes can be classified into two main categories: deterministic and stochastic. The fundamental difference between these two types of variables lies in the predictability of their outcomes. In deterministic processes, there is only one possible outcome for a given set of initial conditions. The outcome is entirely determined by the rules that govern the system. On the other hand, in stochastic processes, the outcome is not predetermined, and it can be influenced by random factors.

Stochastic processes are characterized by their probability distribution, which describes the likelihood of different outcomes occurring. In other words, the distribution tells us how often each outcome is expected to occur. The concept of stochasticity is related to the chaotic nature of some systems. A system is considered chaotic if nearby solutions to the system's equations diverge due to numerical noise. In other words, small changes in the initial conditions can lead to vastly different outcomes.

For LES, most of the available turbulence closures are deterministic and physics-based expressing the subgrid stresses by an extra diffusivity [6], with

$$\mathcal{C}(\overline{u}) \approx \nabla \cdot (\nu_t \nabla \overline{\mathbf{u}}), \tag{12}$$

which means that for the same coarse-resolution velocity field ($\overline{\mathbf{u}}$), the closure will always return the same extra diffusivity field (ν_t). Some of these models that follow this approach are the Smagorinsky-Lilly model [10, 14], the Dynamic Smagorinsky model [5] and Anisotropic Minimum Dissipation model [15, 16]. One of the drawbacks of using those models is that they do not represent the backscatter and tend to be too diffusive [7].

More recently, there are efforts in representing turbulence closures using machine learning methods. A recently published paper shows that deep learning methods can correctly represent the backscatter, and that the closure model can be adapted for different Reynolds numbers by using transfer learning [7]. For small LES cutoff (LES $\leq 8\,$DNS), another recent study has shown that we can recover $\mathcal{N}(\mathcal{X})$ from $\overline{\mathcal{X}}$ and the closure $\mathcal{C}_\mathcal{G}$ can directly estimate $\overline{\mathcal{N}(\mathcal{X})}$ [1], but for larger coarse-grainings (akin to ocean and atmospheric simulations) or larger Reynolds number simulations (which lead to longer enstrophy/direct cascade), we expect that the closure stops being deterministic and becomes stochastic, which means we can no longer recover $\mathcal{X}$ from $\overline{\mathcal{X}}$. In other words, there are many $\mathcal{X}$ that can generate the same $\overline{\mathcal{X}}$. The stochastic approach for the turbulent closure model has recently been employed to parameterize oceanic momentum forcing [8]. However, it is still open as to whether and how the transition between deterministic and stochastic closure is made.

Thus, this paper aims to answer the following scientific questions:

- How different are subgrid stresses from different nearby turbulent solutions?
- How fast do nearby coarse-grained solutions diverge?
- How quickly do the subgrid stresses decorrelate compared to the LES timestep?

3 Experimental Setup and Computer Resources

As Equation 11 shows, the subgrid stress is quite complex and multidimensional. Following the top-down modeling approach, we will simplify the problem by considering a 2D turbulent and incompressible flow $\mathbf{u}(\mathbf{x}, t) = [u(\mathbf{x}, t), v(\mathbf{x}, t)]$ defined by

$$\partial_t \mathbf{u} = -\mathbf{u} \cdot \nabla \mathbf{u} - (1/\rho_0)\nabla p + \nu \nabla^2 \mathbf{u} - \alpha \mathbf{u} + \mathbf{F}_u, \tag{13}$$

where $\mathbf{F}_u$ is a forcing term and α is a linear drag [2]. This linear drag comes from the 3D world that is not being solved and removes energy at large scales, which is being accumulated by the inverse energy cascade.

We can simplify by introducing a stream function $\mathbf{u} = [\partial_y \psi, -\partial_x \psi]$ and rewrite equation 13 for the vorticity field, that in 2D, will be a scalar $\omega = \nabla \times \mathbf{u}$:

$$\partial_t \omega = -J(\omega, \psi) + \nu \nabla^2 \omega - \alpha \omega + F \tag{14}$$

where $J(\omega, \psi) = \partial_x \omega \partial_y \psi - \partial_y \omega \partial_x \psi = \mathbf{u} \cdot \nabla \omega$ (advection of vorticity) and $F = \nabla \times \mathbf{F}_u$.

We use the Dedalus package [3] to solve the system of differential equations on a doubly periodic square domain with 4096 points on each side. Then we apply a stochastic Gaussian forcing centered at k=32, with width of 2 and random phase to the model. The viscosity and linear drag were chosen to fit the power spectrum within the wave numbers solved by the model. We use adaptive time stepping for the simulations, always keeping the CFL condition stable.

The computational resources used by this project were a separate challenge. We ran each pair of simulations on MIT's Supercloud machine [13] using 16 nodes and 768 cores. This added up to a total of 200,000 CPU hours, including simulations and the analyzes.

The primary challenge of this project is the huge amount of data generated by the DNS simulations. The simulations generated a massive amount of data due to the high temporal resolution and domain size. To handle the data efficiently, we used various Python packages, including xarray. These packages helped us work with labeled, multi-dimensional arrays and allowed us to handle large volumes of data through parallelized computing. We also used xhistogram to group wavenumber bins of computed correlations, which required significant computational resources in our case.

Despite having access to the vast resources from MIT Supercloud, we still had to spend a lot of time testing different methods of chunking the data to optimize the simulations and analyses in parallel. This involved experimenting with different ways of allocating memory resources and balancing the computational load. We explored various chunk sizes, overlapping chunks, and parallelization strategies to speed up the analyses. Additionally, we had to consider the significant memory usage and storage requirements of the data. Managing the large volume of data generated from the simulations was a crucial aspect of this study that required us to develop efficient workflows and data handling techniques.

4 Simulation Spin-up

We start the simulation at rest. Thus, we expect the direct and inverse energy cascades to distribute at different scales the energy fed at the forcing wavenumber. The steady state is reached when the energy spectrum no longer changes, i.e., when the energy introduced by the forcing is entirely dissipated at smaller scales through viscosity and at larger scales through linear drag (Figure 1).

The direct cascade stabilizes first and is commonly called the enstrophy ($\langle \omega^2 \rangle / 2$) cascade, because the enstrophy is concentrated at smaller scales. The reverse cascade transfers energy to

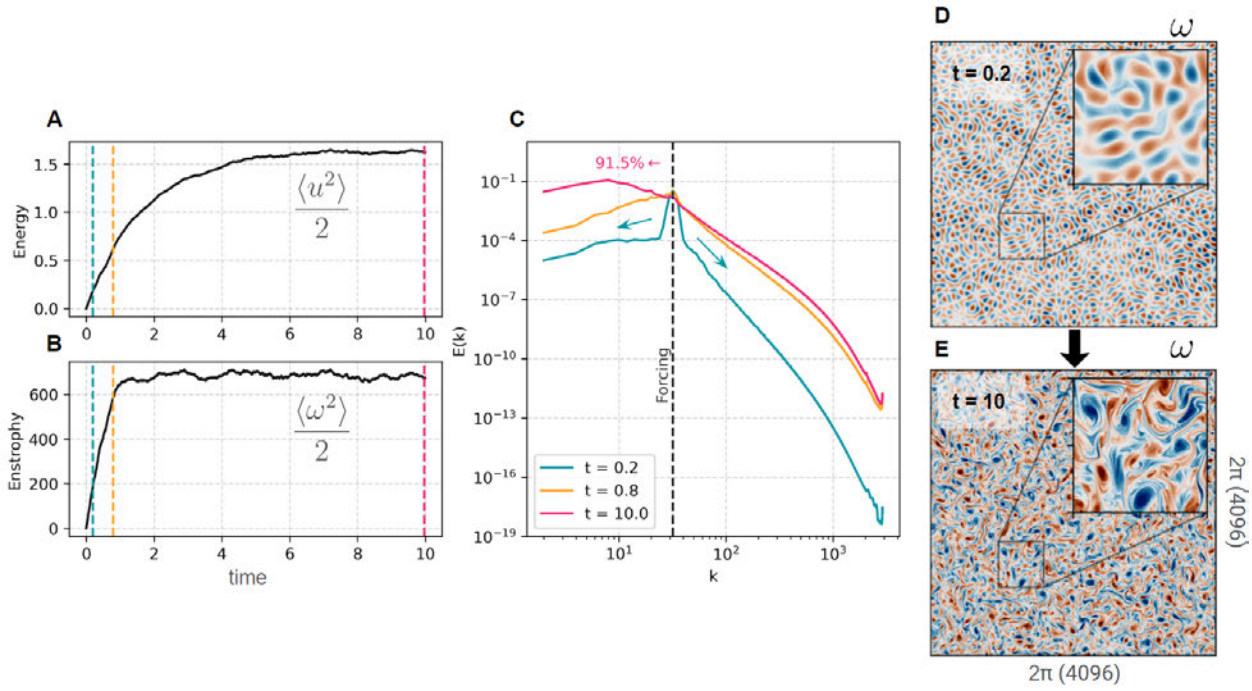


Figure 1: Spin-up time series of kinetic energy (A) and enstrophy (B). Colored vertical dashed lines mark the snapshots used to present the energy spectrum in C. D and E depict vorticity snapshots for t=0.2 and 10, respectively.

larger, more energetic scales that are finally drained by the linear drag added to the system. We verify the steady state of the inverse cascade through the time series of the kinetic energy ($\langle u^2 \rangle / 2$) of the domain. $\langle \cdot \rangle$ denotes a horizontal average of the quantity.

Based on these characteristics of the turbulent process, we observe that the direct cascade stabilizes around t=2, and the inverse one takes longer, having the entire spectrum steady after t=10. (The time here is non-dimensional and depends on the injection rate at the forcing scale.) The steady state and turbulence become evident when looking at two vorticity snapshots (Figure 1), one at t=0 with most of the energy clearly at the forcing scale and another at t=10 where a myriad of scales are observed in a more physically-structured flow. Starting at t=10 we then run the sets of simulations that are analyzed in later sections.

To investigate decorrelation at different spatial scales, we compute the autocorrelation (C) of the Fourier coefficients (f_k) of the solution and performed radial averages for Δk=2 bins.

$$C(k_x, k_y, \tau) = \mathcal{F}_\tau^{-1}\{\mathcal{F}_t\{f(k_x, k_y, t)\} \cdot \mathcal{F}_t\{f(k_x, k_y, t)\}^*\}, \tag{15a}$$

and

$$C(k, \tau) = \frac{1}{N} \sum_{i=1}^{N} C(k_x^i, k_y^i, \tau), \text{ for all } \left| \sqrt{(k_x^i)^2 + (k_y^i)^2} - k \right| < \Delta k. \tag{15b}$$

The autocorrelation was performed using Fast Fourier Transform (FFT, hereafter $\mathcal{F}$) to speed up the calculation and reduce computational bottlenecks. We observe that the decorrelation time scale is shorter than the time scale of the CFL condition at all wave numbers, which means that features at all spatial scales decorrelate more slowly than can be resolved in a low-resolution model

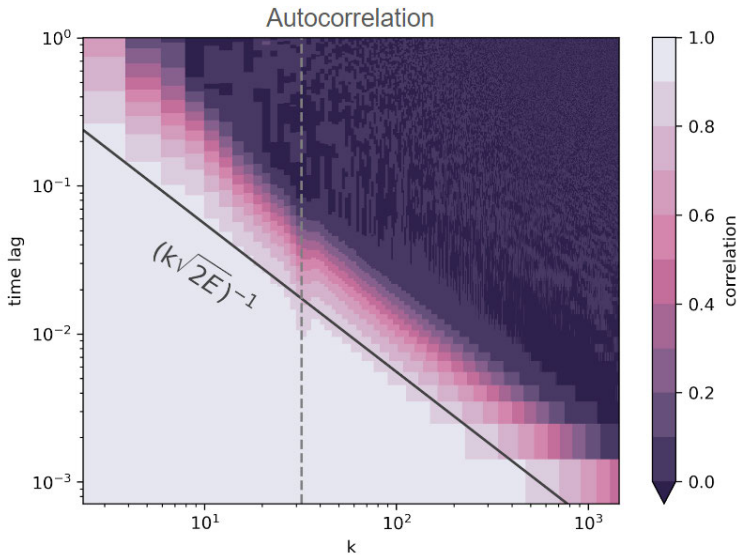


Figure 2: Autocorrelation of streamfunction as a function of wavenumber and time lag for the noise-free solution. Vertical dashed line marks the forcing wavenumber.

(Figure 2). The time scale of the CFL condition is $T = (k\sqrt{2E})^{-1}$, E being the average kinetic energy of the system (≈1.6) and k the wavenumber.

5 Nearby Solutions

After the simulation spin-up, we added noise so that we have nearby solutions that diverge over time. To do this, we run sets of simulations for another t=3 adding white noise to them. We test noise varying for each simulation between 10^{-10} and 10^{-4}, but keep 10^{-7} and 10^{-6} for the following analyses (Figure 3). The value of the added noise was chosen so that it would only affect scales smaller than the forcing scale. In other words, the noise is in the inertial sub-range.

To track the divergence of nearby simulations, we computed the normalized error in the noise-added simulations with respect to the noiseless simulation as follows:

$$\mathcal{E}_i = \frac{|\psi_0 - \psi_i|}{|\psi_0|}. \tag{16}$$

where ψ_0 and ψ_i are the spectral coefficients of the noise-free and noise-added simulations, respectively.

The evolution of the normalized error at different scales shows that there are two different time scales for the propagation of noise at different scales (Figure 4). The shortest scale lasts only t=0.2 and concerns the time it takes for the noise to change to smaller scales, that is, to travel down the direct cascade of turbulence. The second time scale is longer and is about the backscatter, causing the noise to follow the reverse turbulence cascade and alter larger scales. The error is calculated from the amplitude of the difference between the spectral coefficients of the solution without noise and the solution with noise, and is normalized by the amplitude of the coefficients of the solution without noise. After calculating the error, we perform box averaging to transform from the horizontal vector wave number to the isotropic wave number. In other words, we radially average the error in the wave number plane for dk=2.

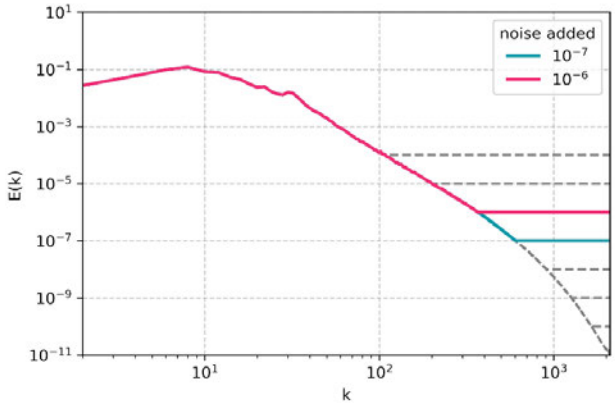


Figure 3: Initial energy spectrum for noise-added simulations. White noise is added at different levels, but we keep 10^{-7} and 10^{-6} for further analyses.

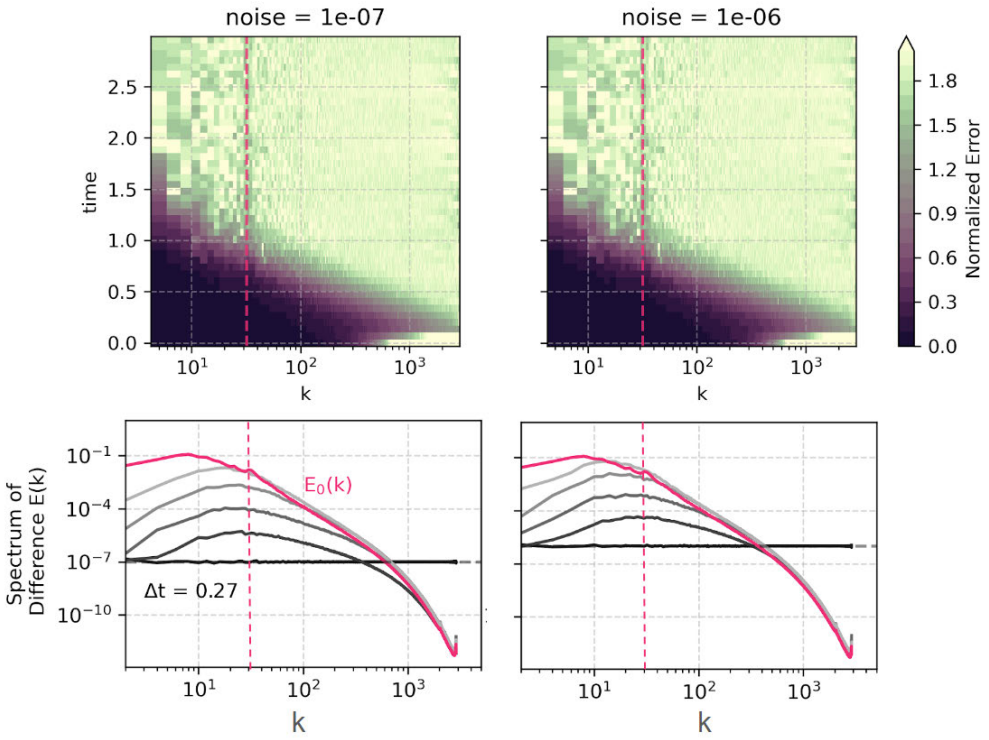


Figure 4: Spectral coefficients error between noise-added and noise-free simulations as a function of wavenumber and time (upper panels). The error is normalized by the amplitude of the coefficients in the noise-free simulation. Energy spectrum (pink line) and spectrum of difference evolution for each Δt=0.27 (gray scale). Left panels for noise=10^{-7} and right panels for noise=10^{-6}. Vertical dashed line marks the forcing wavenumber.

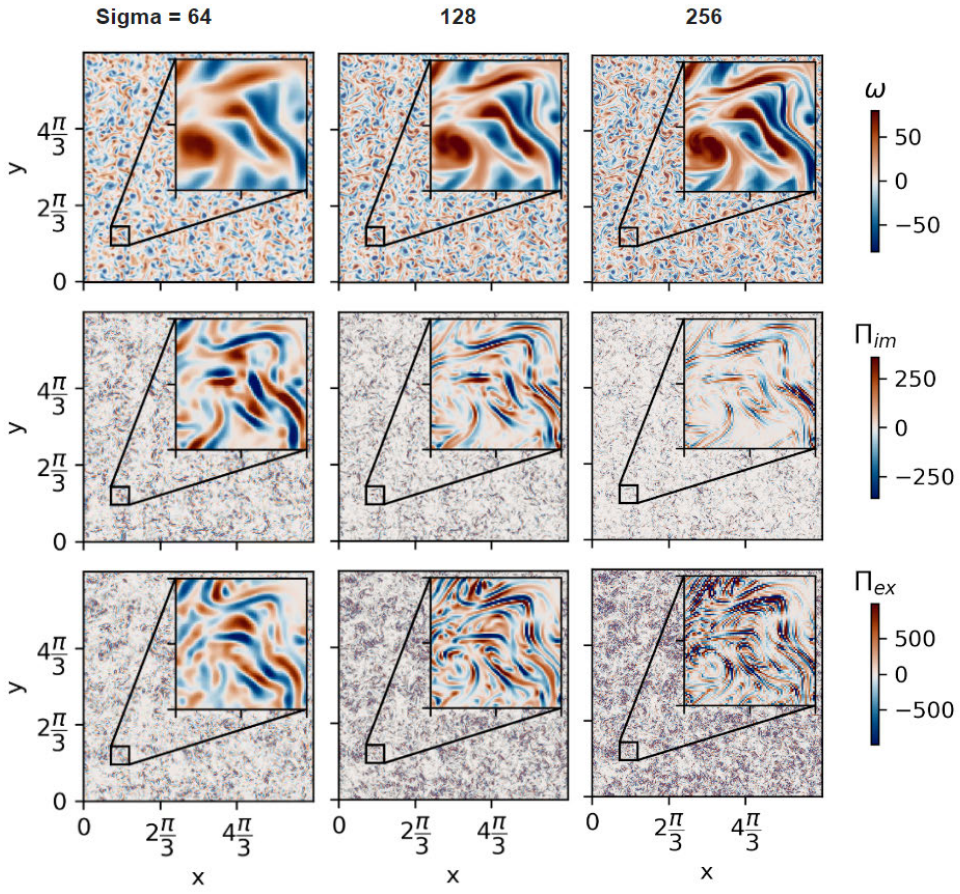


Figure 5: Filtered fields for vorticity (top), implicitly (middle) and explicitly (bottom) filtered subgrid stress. The standard deviation of the filter increases from left to right (σ = 64, 128 and 256)

The difference spectrum shows that, as expected, initially the noisy simulation differs only by white noise. Quickly the differences reach the level of the spectrum for features smaller than the scale of the noise. More slowly, the backscatter accounts for larger features, taking approximately t=2 for the differences to reach the spectrum level.

To analyze the sub-grid stress for LES, we filtered the data using a cutoff of k=512. The filter is defined by a Gaussian function with different standard deviations (σ = 64, 128 and 256):

$$\hat{f}_k = e^{-k^2/2\sigma^2}. \tag{17}$$

The larger the deviation, the less information will be filtered out. We then apply these filters to the solutions of the equations at different noise levels to check the differences with respect to subgrid stress.

Since the definition of subgrid stress depends on the filter applied (top bars in Equation 3) the stress fields differ for each sigma used. We observe that the field is smoother and larger in scale for filters with smaller deviations. The reverse is also true for larger deviations, with more complex fields with more variance at smaller scales.

The evolution of the vorticity and subgrid stress fields demonstrate an apparent separation of

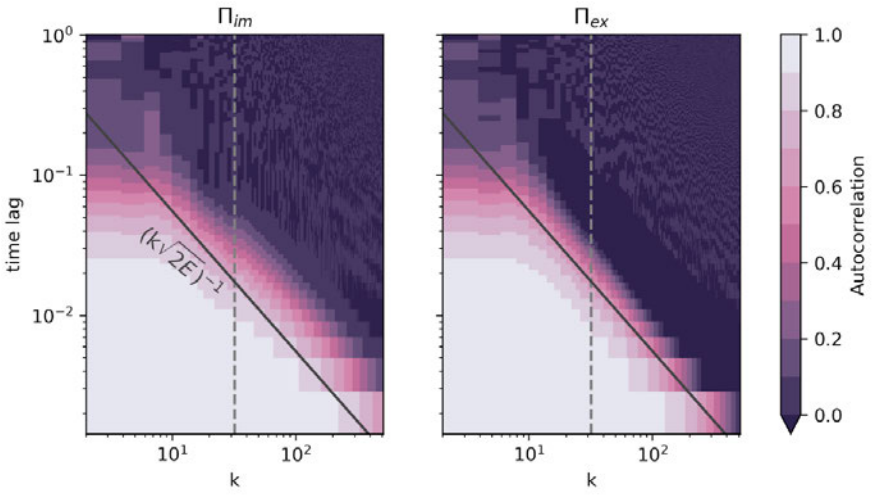


Figure 6: Autocorrelation of streamfunction as a function of wavenumber and time lag for for implicitly (left) and explicitly (right) filtered subgrid stress ($\sigma = 128$) of the noise-free simulation. Vertical dashed line marks the forcing wavenumber.

scales. In other words, we can observe from the video [1] that the time scale at which the subgrid stress diverges in close simulations is different from the time scale at which the solutions (vorticity field) diverge (pause video for simulation t=1 and check that vorticity fields are more similar between noise=0 and 10^{-7} than subgrid stresses.) This apparent separation of scales does not seem to depend on the type of filter used, that is, the apparent time scale at which the subgrid stress diverges between nearby simulations is indistinguishable between the implicit or explicit filter. Both exhibit decorrelation scaling larger than the CFL time scale for most spatial scales, with the exception of k<10, when the decorrelation scaling becomes constant and independent of k. This pattern is probably associated with very little energy at large scales for subgrid stress. Despite the similarities between implicit and explicit filter, the decorrelation scale of the subgrid stress is slightly smaller for the explicit filter.

The filters had a minimal effect on the current function fields, but as expected, the effect increased with the order of the derivatives, particularly for vorticity fields. When examining the subgrid stress spectrum, it was found that the patterns varied based on the filter width, with distinct differences observed for implicit and explicit filters. Specifically, larger standard deviations were associated with larger energy peaks in explicit filters, while smaller energy peaks were found in implicit filters. This intriguing relationship between peak energy and filter width in explicit and implicit filtering warrants further investigation in future studies. Overall, these findings highlight the importance of carefully considering filter choice and width, particularly when investigating the effects of subgrid stress on larger scales.

To quantitatively investigate the apparent scaling separation observed in the video, we calculate the evolution of the Root Mean Squared Error (RMSE) of different variables at noise levels and filter widths.The error is scaled by the maximum error from the end of the time series. The time scale it takes for RMSE to reach O(1) for the streamfunction is larger than for the subgrid stress, but in any case, by the time the subgrid stress diverges completely, the vorticity fields have reached 50% of the maximum error (Figure 8). This makes it clear that the scaling separation, although it exists, is not fully defined for this problem.

[1] video only viewable in oneline version of paper

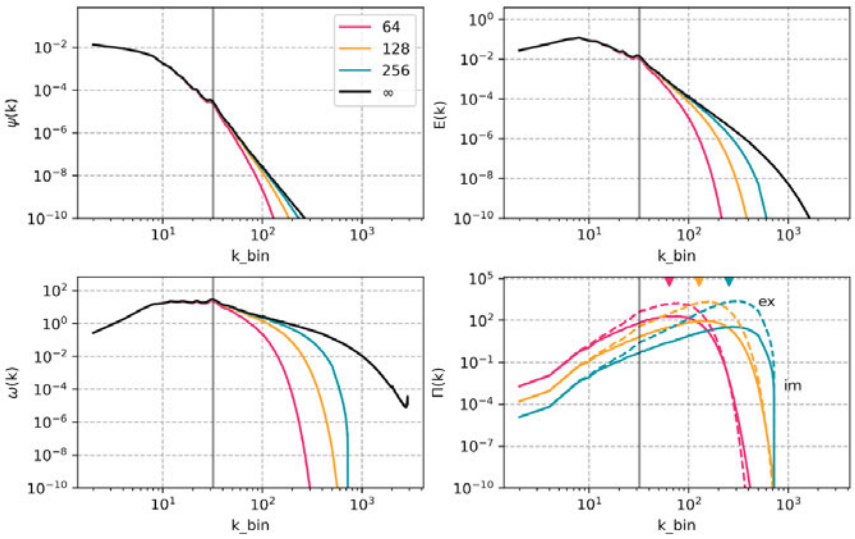


Figure 7: Unfiltered (black) and filtered (colors) spectra for streamfunction, kinetic energy, vorticity and subgrid stresses (implicitly filtered in solid and explicitly filtered in dashed lines). $\sigma = 64$ (pink), 128 (orange) and 256 (blue). Vertical gray line marks the forcing wavenumber.

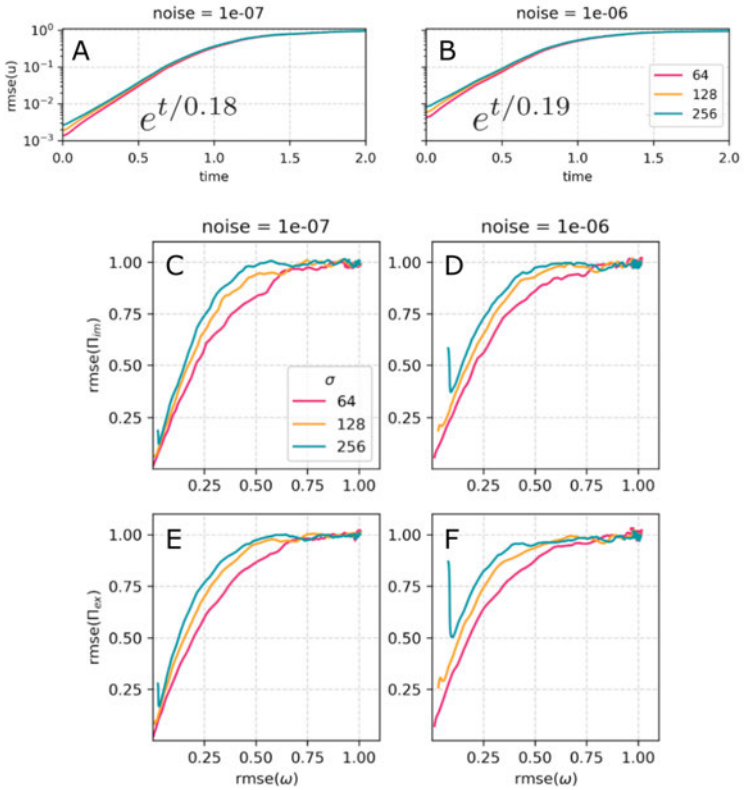


Figure 8: Timeseries and Lyapunov Exponent for the Root mean squared error (RMSE) of velocity (A,B). RMSE of implicitly (C,D) and explicitly (E,F) filtered subgrid stresses as a function of the RMSE of vorticity.

We also measured the rate of divergence between nearby solutions assuming an exponential character on the RMSE $\approx e^{t/\lambda}$, where λ is the Lyapunov Exponent. Lyapunov Exponents are a fundamental concept in chaos theory, used to measure the rate of divergence of nearby trajectories in a dynamical system, which provides a quantitative measure of the unpredictability and complexity of a system.

The exponential character of the RMSE of the velocity fields has a Lyapunov exponent of 0.18/0.19 (Figure 8), considerably larger than the decorrelation or CFL time scale for virtually any spatial scale, which means that not only is the separation of scales unclear, but a possible LES solution would have enough temporal resolution to capture the divergence.

6 Conclusions and Future Work

Our results suggest that DNS solutions for two-dimensional turbulence are chaotic, as numerically similar solutions diverge, but the LES perspective of this problem is not strongly stochastic. We now list the scientific questions presented in the introduction section with their respective answers:

- How different are subgrid stresses from different nearby turbulent solutions?
  We could not find a case in which numerically nearby solutions had completely different subgrid stress fields.
- How fast do nearby coarse-grained solutions diverge?
  The Lyapunov exponent of the RMSE of the velocity fields is about 0.18/0.19 time units.
- How quickly do the subgrid stresses decorrelate compared to the LES timestep?
  For most of the scales, both implicit and explicit subgrid stress fields decorrelate slower than the CFL time scale.

However, this conclusion was based on testing a limited range of flow problems, and it is not yet clear whether it holds true for larger domains with longer turbulence cascades (i.e., higher Reynolds numbers). As the Reynolds number increases, the range of scales involved in the turbulence cascade becomes larger, which could lead to more separation between the large and small scales. This, in turn, could potentially lead to increased stochasticity in the LES turbulence closure. Additionally, at higher Reynolds numbers, the larger scales might interact with a greater number of smaller scales, making the overall dynamics more complex and harder to predict.

We initially hypothesized that explicit filtering would result in smoother subgrid stress fields, as compared to the implicit method. However, our findings suggest otherwise, as the subgrid stress fields obtained using explicit filtering were actually more noisy than those obtained using implicit filtering (Figure 5). This unexpected result raises questions about the impact of filtering method choice on the accuracy of turbulence closures. Our future work will focus on exploring this issue in more detail. We aim to investigate the sources of noise in the explicit filtering approach and provide insight into the best practices for choosing a filtering method that optimizes the accuracy of turbulence closures.

In addition to these challenges, it is also important to consider the impact of three-dimensional effects on the stochasticity of LES models. Our study focused on two-dimensional turbulence, but many practical applications, such as small-scale atmospheric and oceanic modeling, involve three-dimensional flows. In these cases, the interactions between the large and small scales are even more complex, and the separation of scales could present different results to those discussed in this study. Understanding the behavior of LES models in 3D turbulence are crucial for accurately simulating many real-world problems.

Finally, it is essential to explore how sensitive the results presented in LES are to different types of dynamical problems. For example, the quasi-geostrophic approximation is a widely used simplification in some large-scale atmospheric and oceanic problems and it assumes the flow is nearly two-dimensional due to the effect of rotation. It is possible that the accuracy of subgrid-scale modeling could vary depending on the type of flow problem being studied and we believe that the tools developed in this project can be used for further investigations.

In summary, there are still many open questions and challenges related to this study, including the sensitivity of the results to different types of dynamical problems and geometry, the choice of filtering method, and the potential for increased stochasticity at higher Reynolds numbers. Addressing these challenges will be critical for developing more accurate and efficient LES models and understanding the stochastic nature of turbulence closures.

Data Availability Statement

We are committed to the principles of open science and believe that research should be accessible and transparent. As such, all of the code used for modeling, analysis, and plotting in this study has been made publicly available on Github (`https://github.com/iuryt/stochastic_closures`). We support open-source projects and believe that sharing code and data is an important step towards advancing scientific knowledge.

Acknowledgments

I would like to start by thanking all the mentors I have had throughout my career, including my current advisor **Amit Tandon**, my previous advisor **Ilson Silveira**, and who collaborated with my research and have given me letters of recommendation for this program (**Simon de Szoeke** and **Amala Mahadevan**). I came from a place with few opportunities and both my parents (**Sandra** and **Vicente**) and my academic mentors believed and invested a lot of time to get me to where I am now. Since I started my undergraduate degree in Oceanography at the Federal University of Ceará in northeastern Brazil, I have always greatly admired and wanted to participate in this program, despite finding it practically unattainable for someone with my background. I was in disbelief when I received the approval email and perhaps out of impostor syndrome, it took me a while to understand that the selection committee and the directors believed in my potential.

The program directors, **Colm** and **Stefan**, with such different personalities, were always very welcoming and were always very concerned about how I was handling the program.

I would like to thank **Keaton Burns** very much for being so patient and mentoring me so closely throughout the process of developing this project. I learned a lot from his mentorship and I am very grateful that he trusted and invested in my work. I felt lucky to have been mentored by someone so skillful, pedagogical and patient.

I would also like to thank **André Souza** for always being so open to my questions about this new world that the GFD introduced me to. As Supercloud computing resources were crucial to the development of this work, I could not leave **Chris Hill** out of this statement. He was essential in enabling us to run so many complex numerical experiments in such a short time.

I would especially like to thank **Peter Schimd** and **Laure Zanna** for being inspiring teachers, making such complex subjects seem simple, and for always being so open to discuss and answer our questions. Laure also mentored me on this project and I loved all the conversations we had about science and career. **Pedram Hassanzadeh** helped me a lot to understand the language and

application of different artificial intelligence algorithms and without him, I would not have been able to understand some of the applications this work may have in the future.

I would also like to thank **all the speakers** who took the time to present their work to us and then talk to us about various topics. Likewise, I would like to give a special thanks to those who were there all summer. **Glenn** for being our oracle, **Pascale** and **Carl** for asking the best questions, Neil for showing that science can also be fun and **Joe** for his dynamics insights.

I also dedicate this paper to all the Fellows this year. **Sam**, **Rui**, **Kasturi**, **Tilly**, **Ruth**, **Claire** and **Ludovido** each showed me their perspective on the world and science. I was very happy to have had such a diverse, collaborative, friendly and non-competitive group. You all made this whole experience shareable.

I would like to thank my friend **Nikiforos** very much. In between the comings and goings, he was my road companion and helped me keep my head while I was completely overwhelmed with so much daily information.

I would like to thank **Julie Hildebrandt** and **Janet Fields** for all their support in the admission and administration. As I would also like to thank the janitor **Rich** who came by late at night every day to clean the cottage, remind me that I needed to come home and sleep, and have friendly conversations about random subjects.

I also dedicate this work to my great friends who encouraged me and helped me from the beginning to the end: **Cesar Rocha**, **André Schmidt**, **Filipe Pereira**, **Igor Uchôa**, **Bruno Gonçalves**, **Jeane Rodrigues**, **Letícia Lima**, **Christian Buckingham**, **Elizabeth and the Ells family**, and **Siddhant Kerhalkar**.

Finally I would like to thank my beloved wife **Ágata** for supporting me so much and giving me so much comfort so that I could feel at home even if I was far away. You are the best companion.

References

[1] Y. Bar-Sinai, S. Hoyer, J. Hickey, and M. P. Brenner, *Learning data-driven discretizations for partial differential equations*, Proceedings of the National Academy of Sciences, 116 (2019), pp. 15344–15349.

[2] G. Boffetta and R. E. Ecke, *Two-dimensional turbulence*, Annual Review of Fluid Mechanics, 44 (2012), pp. 427–451.

[3] K. J. Burns, G. M. Vasil, J. S. Oishi, D. Lecoanet, and B. P. Brown, *Dedalus: A flexible framework for numerical simulations with spectral methods*, Physical Review Research, 2 (2020), p. 023068.

[4] B. Fox-Kemper, A. Adcroft, C. W. Böning, E. P. Chassignet, E. Curchitser, G. Danabasoglu, C. Eden, M. H. England, R. Gerdes, R. J. Greatbatch, S. M. Griffies, R. W. Hallberg, E. Hanert, P. Heimbach, H. T. Hewitt, C. N. Hill, Y. Komuro, S. Legg, J. Le Sommer, S. Masina, S. J. Marsland, S. G. Penny, F. Qiao, T. D. Ringler, A. M. Treguier, H. Tsujino, P. Uotila, and S. G. Yeager, *Challenges and Prospects in Ocean Circulation Models*, Frontiers in Marine Science, 6 (2019), p. 65.

[5] M. Germano, U. Piomelli, P. Moin, and W. H. Cabot, *A dynamic subgrid-scale eddy viscosity model*, Physics of Fluids A: Fluid Dynamics, 3 (1991), pp. 1760–1765.

[6] M. C. Gregg, *Mixing and its role in the ocean*, in Ocean Mixing, Cambridge University Press, Cambridge, 2021, pp. 1–24.

[7] Y. Guan, A. Chattopadhyay, A. Subel, and P. Hassanzadeh, *Stable a posteriori les of 2d turbulence using convolutional neural networks: Backscattering analysis and generalization to higher re via transfer learning*, Journal of Computational Physics, 458 (2022), p. 111090.

[8] A. P. Guillaumin and L. Zanna, *Stochastic-deep learning parameterization of ocean momentum forcing*, Journal of Advances in Modeling Earth Systems, 13 (2021), p. e2021MS002534.

[9] W. G. Large, J. C. McWilliams, and S. C. Doney, *Oceanic vertical mixing: A review and a model with a nonlocal boundary layer parameterization*, Reviews of Geophysics, 32 (1994), p. 363.

[10] D. K. Lilly, *On the numerical simulation of buoyant convection*, Tellus, 14 (1962), pp. 148–172.

[11] J. Marshall, A. Adcroft, C. Hill, L. Perelman, and C. Heisey, *A finite-volume, incompressible Navier Stokes model for studies of the ocean on parallel computers*, Journal of Geophysical Research: Oceans, 102 (1997), pp. 5753–5766.

[12] S. A. Orszag, *Analytical theories of turbulence*, 41, pp. 363–386.

[13] A. Reuther, J. Kepner, C. Byun, S. Samsi, W. Arcand, D. Bestor, B. Bergeron, V. Gadepally, M. Houle, M. Hubbell, M. Jones, A. Klein, L. Milechin, J. Mullen, A. Prout, A. Rosa, C. Yee, and P. Michaleas, *Interactive supercomputing on 40,000 cores for machine learning and data analysis*, in 2018 IEEE High Performance extreme Computing Conference (HPEC), IEEE, 2018, pp. 1–6.

[14] J. Smagorinsky, *General circulation experiments with the primitive equations: I. the basic experiment*, Monthly weather review, 91 (1963), pp. 99–164.

[15] R. Verstappen, *How much eddy dissipation is needed to counterbalance the nonlinear production of small, unresolved scales in a large-eddy simulation of turbulence?*, Computers & Fluids, 176 (2018), pp. 276–284.

[16] C. A. Vreugdenhil and J. R. Taylor, *Large-eddy simulations of stratified plane Couette flow using the anisotropic minimum-dissipation model*, Physics of Fluids, 30 (2018), p. 085104.

Statistical Analysis of Multidimensional Dynamical Systems

Ludovico Theo Giorgini

Introduction

In this work we will study the temporal evolution of an autonomous continuous-time dynamical system of the form

$$\dot{x}(t) = F(x(t)), \tag{1}$$

where $x : [0, T] \to \mathbb{R}^D$, $T > 0$ is the state of the system and $F : \mathbb{R}^D \to \mathbb{R}^D$ a deterministic force field. We point out that this representation can be extended to non-autonomous dynamical systems by increasing the dimension and it can take into account multi-scale characteristics by rescaling each variable according to the time-scale τ_i on which the dynamics are realised $x_i(t/\tau_i) \to y_i(t)\ \forall i \in [1, D]$.

In many cases the number of snapshots of the dynamical system and their size make its study impractical, hence the need for building a coarse-grained model able to maintain the statistical and dynamical features of the original one (see for an example of clusterization of a multidimensional dynamical system Fig. (1)).

This coarse-grained model is constructed by defining a distance d between snapshots and clusterizing them according to it. In this way, we map snapshots close in space to the same cluster and the original multi-dimensional time series reduces to a one-dimensional one. The choice of the distance and the number of clusters N is not defined *a priori* and strongly depends on the system under study and on the physical quantities of interest.

The dimensionality reduction makes our coarse-grained system stochastic to take into account the information loss of the precise trajectory followed by the system in the phase space, and a probablistic approach becomes necessary to study its dynamics.

The temporal evolution of our coarse-grained system is described by the following Markov process

$$X_{n+1} = S(X_n) = s(X_n) + W(X_n), \tag{2}$$

where $X : [0, T] \to \mathbb{R}^{N^m}$, $T > 0$ is the state of the system embedded in a delay embedding of size m and $s : \mathbb{R}^{N^m} \to \mathbb{R}^{N^m}$ and $W : \mathbb{R}^{N^m} \to \mathbb{R}^{N^m}$ are respectively deterministic and stochastic force fields. The amplitude of the stochastic force field can be reduced by increasing the values of N and m. From now on we will consider $m = 1$, the generalization of our results to arbitrary values of m is straightforward.

The forward evolution of the probability distribution function (PDF) ρ associated to the coarse-grained system's state is described by the following Fokker-Plank equation

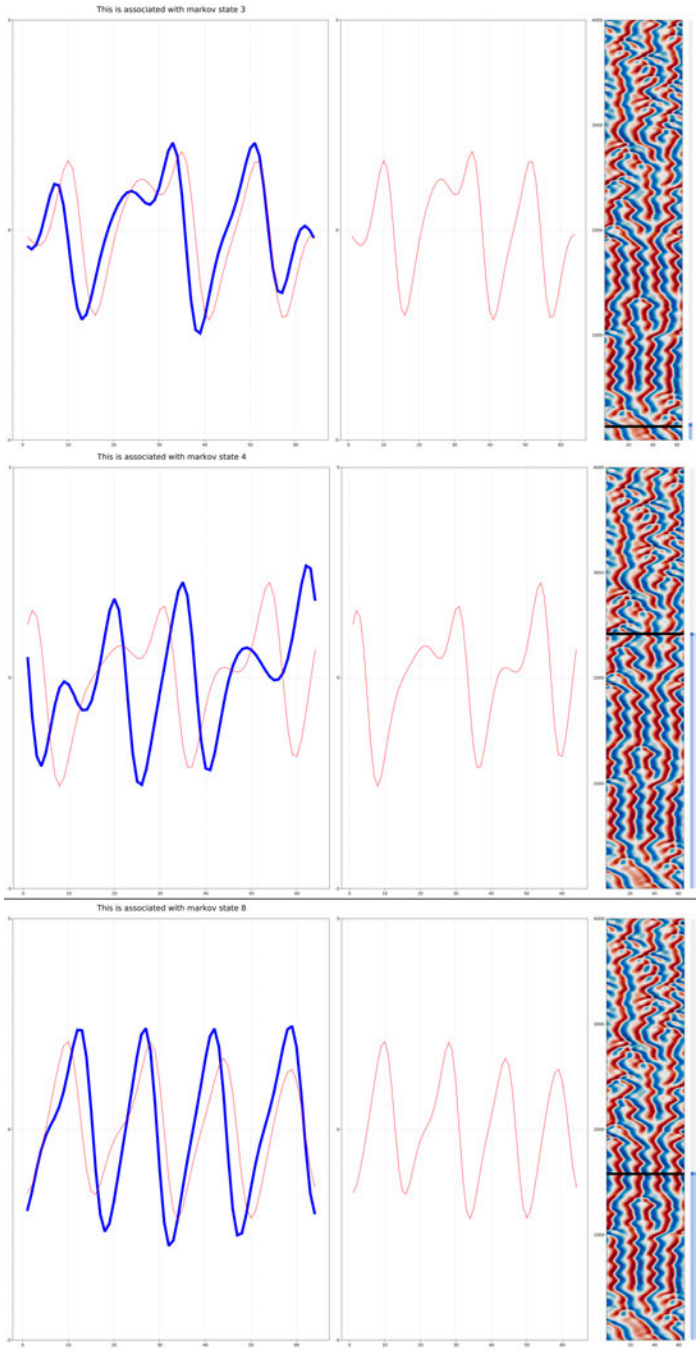


Figure 1: Example of clusterization using the Kuramoto–Sivashinsky equation solved on a lattice of size $L = 34$. The numerical solution of the equation is shown on the right column, while the center and left columns show the snapshots corresponding to the black horizontal line printed above the numerical solution (red curves) and the center of the cluster that the kmeans algorithm assigned to each snapshots (blue curves).

$$\partial_t \rho = \mathcal{L}_{FP}\rho, \tag{3}$$

and by its discrete counterpart

$$\rho_{t+\Delta t} = P^{\Delta t}\rho_t, \tag{4}$$

where $\mathcal{L}_{FP}$ is the Fokker-Plank operator and $P^{\Delta t}$ the Perron-Frobenius operator or the transition matrix for a time step Δt.

In the limit $\Delta t \to 0$, Eq. (4) becomes

$$\lim_{\Delta t\to 0} \rho_{t+\Delta t} = (I + Q\Delta t)\rho_t, \tag{5}$$

taking the continuous limit

$$\dot{\rho} = Q\rho, \tag{6}$$

and then we obtain a forward evolution equation for ρ discrete in space and continuous in time

$$\rho_t = e^{Q(t-s)}\rho_s = P^{t-s}\rho_s, \tag{7}$$

with $t > s$.

The matrix Q can be constructed [1] from the coarse-grained data by noticing that, since we are assuming the Markov property for our system, its sojourn times in each cluster are independent and exponentially distributed with rate $r_i = 1/\tau_i \ \forall i \in [1, N]$ with τ_i the mean sojourn time in cluster i. Therefore, we can write

$$\begin{aligned} \lim_{\Delta t\to 0} \frac{1 - P_{ii}^{\Delta t}}{\Delta t} &= \lim_{\Delta t\to 0} \frac{\Pr(\tau_i < \Delta t)}{\Delta t} = r_i = -Q_{ii}, \\ \lim_{\Delta t\to 0} \frac{P_{ij}^{\Delta t}}{\Delta t} &= \lim_{\Delta t\to 0} \frac{1 - P_{ii}^{\Delta t}}{\Delta t} \frac{P_{ij}^{\Delta t}}{\sum_{j\neq i} P_{ij}^{\Delta t}} = r_i \frac{P_{ij}^{\Delta t}}{\sum_{j\neq i} P_{ij}^{\Delta t}} = Q_{ij}. \end{aligned} \tag{8}$$

An improved algorithm to construct the transition rate matrix

Even if for an autonomous Markov process Q doesn't depend on time, the fact that we constructed this matrix using statistics of the system at infinitesimal time scales often introduces errors which become relevant when we want to compute statistical properties of the system at larger time scales. In the following we propose a method to overcome this problem by correcting the values of Q using information of the system behaviour on larger time scales.

Let's assume that the correct matrix Q_{pert} is similar to the matrix Q that we constructed before. This means that we can obtain Q_{pert} by adding a perturbation to Q, $Q_{pert} - Q = \delta Q = gQ'$ with $Q'_{ij} = O(1)$ and $g \ll 1$. We can then write the eigenvalue equation for Q_{pert} as

$$(Q + gQ')(\phi_0^i + g\phi_1^i) = (\lambda_0^i + g\lambda_1^i)(\phi_0^i + g\phi_1^i), \tag{9}$$

which becomes, taking only the $O(g)$ terms

$$Q'\phi_0^i + Q\phi_1^i = \lambda_0^i\phi_1^i + \lambda_1^i\phi_0^i. \tag{10}$$

Multiplying on the left by the transpose of the unperturbed left eigenfunction of Q, $(\psi_0^i)^T$, we get

$$Q'\phi_0^i = \lambda_1^i \phi_0^i, \tag{11}$$

where we used the fact that the unperturbed left eigenfunctions of Q are orthogonal to the unperturbed right eigenfunctions and then, since the perturbed right eigenfunction can be written as a linear combination of the unperturbed right eigenfunctions, the second term in the l.h.s. of Eq. (10) cancels with the first one on the r.h.s. after the multiplication on the left by $(\psi_0^i)^T$. We can then write the perturbation δQ in function of the unperturbed eigenfunctions of Q and the perturbed eigenvalues as

$$\delta Q = \sum_i \delta\lambda^i \phi_0^i (\psi_0^i)^T, \tag{12}$$

with $\delta\lambda^i = g\lambda_1^i$.

In order to obtain $\delta\lambda^i$ we construct a D-dimensional time series from the coarse grained one by associating to each value X_n a D-dimensional vector containing in each element the X_nth value of the unperturbed left eigenfunction of Q, that is, we perform the map $X_n \to \psi^i_{X_n} = Y_n^i \ \forall i$, where we dropped the 0 in the subscript of ψ^i.

The correlation function for $\tilde{Y}^i = Y^i - \langle Y^i \rangle$ becomes

$$\begin{aligned}
\mathcal{C}_{\tilde{Y}^i}(\tau) &= \frac{(\psi^i)^T \mathrm{diag}(\phi_1)[(\psi^i)^T(P^\tau - \mathrm{diag}(\phi_1))]^T}{(\psi^i)^T \mathrm{diag}(\phi_1)\psi^i} \\
&= \frac{(\psi^i)^T \mathrm{diag}(\phi_1)\left[(\psi^i)^T\left(\sum_{k\neq 1} e^{(\lambda_0^i + g\lambda_1^i)\tau}\phi^k(\psi^k)^T\right)\right]^T}{(\psi^i)^T \mathrm{diag}(\phi_1)\psi^i} \\
&= \frac{e^{(\lambda_0^i + g\lambda_1^i)\tau}(\psi^i)^T \mathrm{diag}(\phi_1)\psi^i}{(\psi^i)^T \mathrm{diag}(\phi_1)\psi^i} = e^{(\lambda_0^i + g\lambda_1^i)\tau}.
\end{aligned} \tag{13}$$

We compute the correlation function from the data for each values of i and we obtain each time $\lambda_0^i + g\lambda_1^i$ from a least square fit. Computing the difference between each exponent and λ_0^i we estimate $\delta\lambda^i$.

The correlation function of the coarse-grained time series $\tilde{X} = X - \langle X \rangle$ becomes

$$\mathcal{C}_C(\tau) = \frac{C^T \mathrm{diag}(\phi_1)\left[C^T\left(\sum_{k\neq 1} e^{\lambda^i \tau}\phi^k(\psi^k)^T\right)\right]^T}{C^T \mathrm{diag}(\phi_1) C}, \tag{14}$$

where C is a vector containing the centers of the clusters and the λ_ks are the eigenvalues of Q_{true}.

In Fig (2) we reported the comparison between the correlation function of the coarse-grained data of the Kuramoto-Shivashinsky model with that one estimated from Q and Q_{true}. Different distances were used to construct the coarse-grained time series, while the number of cluster has been kept fixed $N = 20$. We can notice a remarkable improvement in the correlation function obtained using the eigenvalues and eigenfunctions of Q_{pert} with respect to ones that Q. Same result can also be observed in Fig. (3) for Lorenz 63.

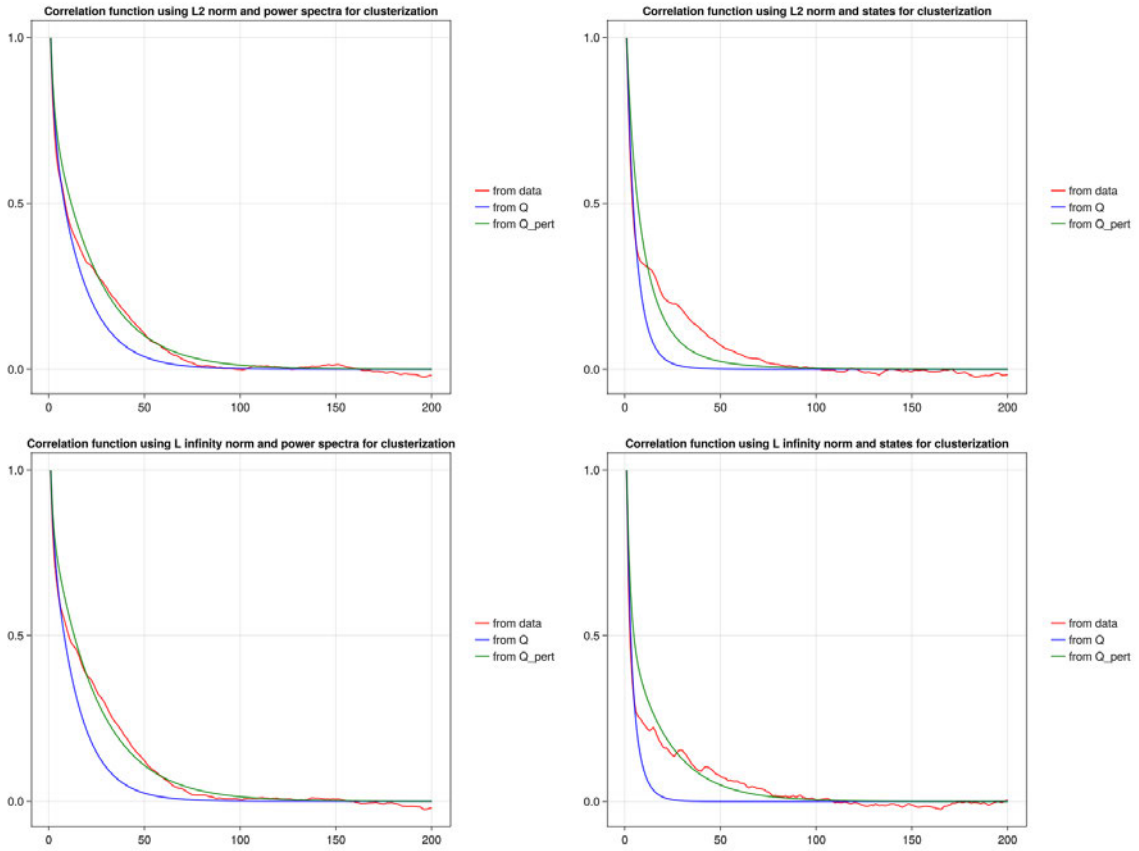


Figure 2: Comparison between the correlation function obtained from the coarse-grained data and one that is estimated from Q and Q_{true} (in the figures Q_{pert}). We used the $L2$ and L infinity norm to define the distances over the states and their power-spectra.

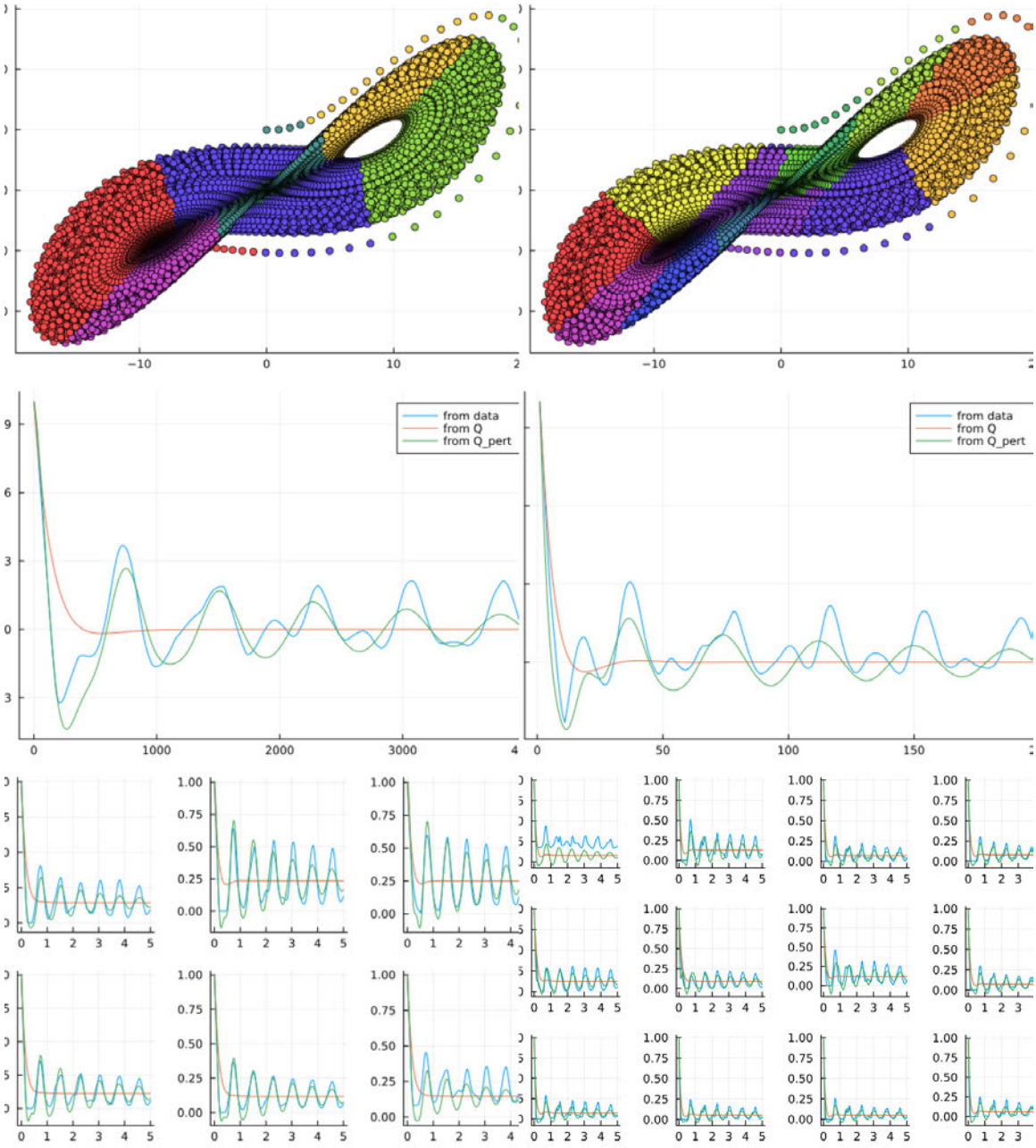


Figure 3: Correlation functions (second row) and diagonal element of the Perron-Frobenious operator (P^t_{ii}, third row) for the coarse-grained Lorenz 63 using 6 (first column) and 12 (second column) clusters. We compared these quantities obtained from the coarse-grained time series (blue curves) with those of ones obtained using Q from Eq. (8) (orange curve) and Q_{pert} from our method (green curves).

Leicht-Newman algorithm for Markov processes

Before we clusterized the snapshots of the system according to their distance in space. We can introduce a further clusterization to group together clusters close in time, that is to split the clusters into communities of connected elements which give us important information about the sojourn time of the system in each region of the phase-space and, as we will see later, about the transition from deterministic to chaotic regimes and vice versa.

To this end, we propose a modification of the directional Leicht-Newman algorithm [2], which consists in dividing a network of size N recursively into two communities. Each vertex i is labeled by $s_i = \pm 1$ depending on which community it has been assigned and the values of s_i are chosen in order to minimize the modularity parameter

$$Q = \frac{1}{2N} \sum_{ij} \left[A_{ij} - \frac{k_i^{in} k_j^{out}}{N} \right] (s_i s_j + 1) = \frac{1}{2N} \sum_{ij} s_i B_{ij} s_j, \tag{15}$$

with A the adjacency matrix, $k_i^{in} k_j^{out}$ the in- and out- degrees of the vertices, and B_{ij} the modularity matrix

$$B_{ij} = \frac{1}{N} \left(A_{ij} - \frac{k_i^{in} k_j^{out}}{N} \right). \tag{16}$$

In our case we consider a network with N vertices and m edges, each of them representing a transition probability of $1/m$. We can write

$$B_{ij}^t = \frac{1}{mN} \left(\#_{ij} - \frac{\#_i^{out} \#_j^{in}}{mN} \right) = \frac{1}{N} \left(\frac{\#_{ij}}{m} - \frac{1}{N} \frac{\#_i^{out}}{m} \frac{\#_j^{in}}{m} \right) \tag{17}$$

By taking the limit $m \to 0$ we obtain

$$B_{ij}^t = P_{ij}^t - \frac{1 - P_{ii}^t}{N} \left(\sum_k P_{kj}^t \right). \tag{18}$$

We have then to minimize the modularity parameter obtained using the modularity matrix defined before.

In Fig. (4 left panel) we plotted $X(t)$ $t \in [1, 2000]$ for the Kuramoto-Shivashinsky model together with the value of s assigned to each X. In this case the clusterization has been performed using $L2$ norm on the power spectra of the snapshots. We can notice how our modified algorithm is able to correctly distinguish between states that exhibit chaotic behaviour from states where the system's behaviour is deterministic. In Fig. (4 right panel) we plotted the two dimensional time series obtained evolving a Browninan motion with noise amplitude $\sigma = 0.75$ inside the potential

$$U(x, y) = (x - 1)^2 (x + 1)^2 (x^2) + (y - 1)^2 (y + 1)^2 \tag{19}$$

together with its clusterization obtained from our version of the Leicht-Newmann algorithm. We can notice how on short time scales the algorithm produces six different clusters coinciding with the six minima of the potentials; for time scales longer than the first passage time

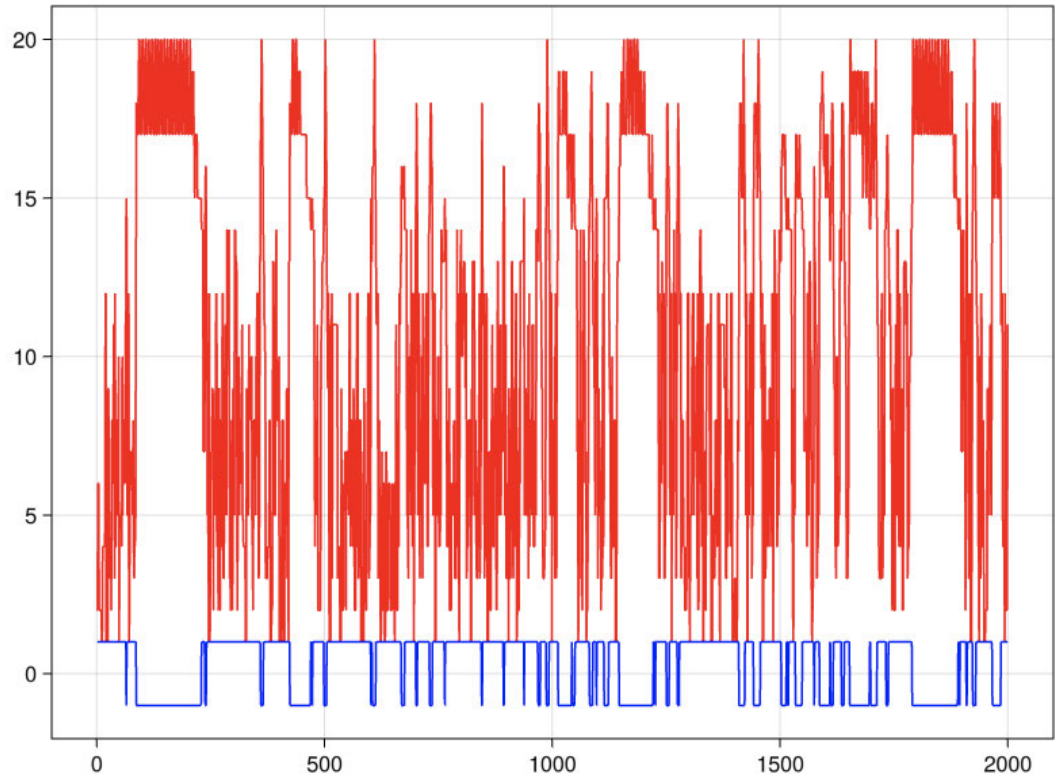


Figure 4: Time behaviour of the coarse grained time series X (red curve) and of the community to which each element of the time series is assigned (blue curve) using the modified version of the Leicht-Newman algorithm for the Kuramoto-Shivashinsky model.

between adjacent minima on the y-axis they become indistinguishable and three clusters are obtained. For time scales even longer than the mean first passage time between minima on the x-axis the algorithm produces only one cluster. This algorithm can also be generalized to other dynamical systems (see Fig. (5)).

From the transition matrix to the deterministic dynamics

We have seen how a dynamical system can be characterised by a deterministic (and stochastic) force field or by a transition matrix. The former is able to correctly reproduce the system's dynamics and evolve it forward in time in order to obtain reliable forecast, but it is extremely difficult to reconstruct, in particular if the system under study is highly non-linear. The latter, instead, can be easily estimated from the coarse-grained model as we described before and it is able to reproduce the main equilibrium and dynamical and equilibrium features of the system, but completely lacks any predictive performances. In the following we propose a method to combine them in order to use the knowledge of the transition matrix to put some constraints over the force fields which will facilitate their estimation and, moreover, it will force them to generate a time series with the same statistical

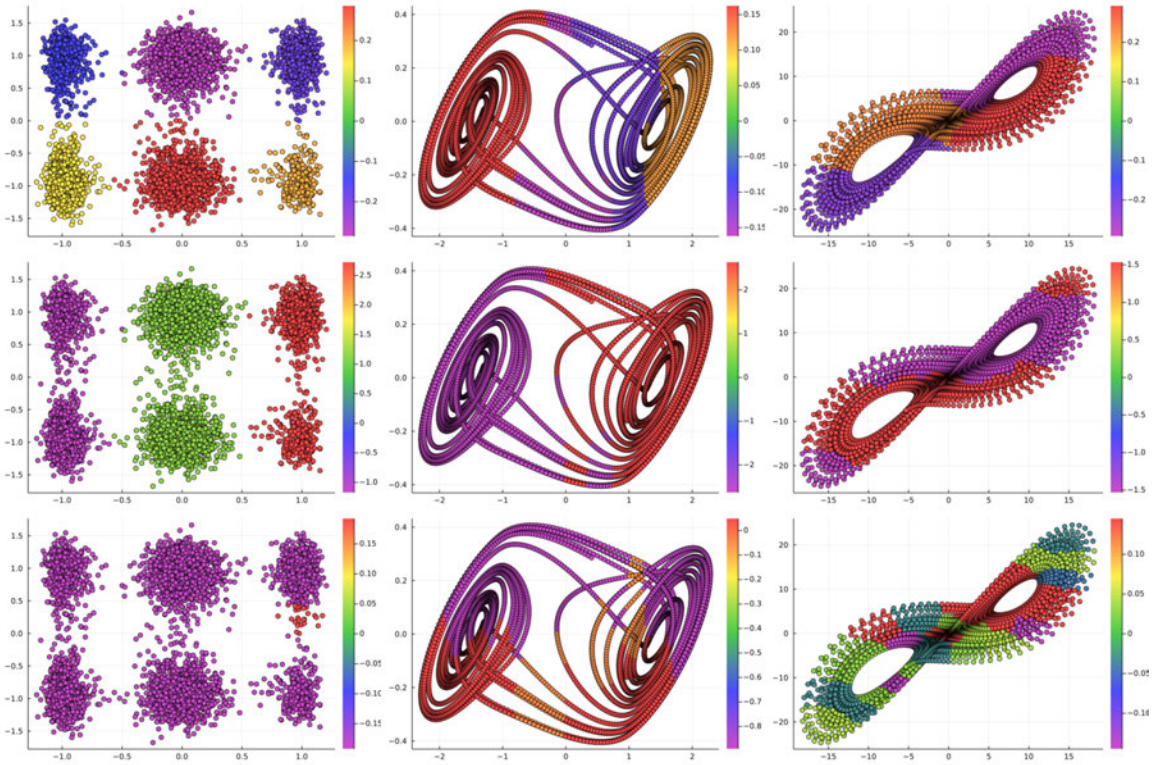


Figure 5: Clusterization using the modified version of the Leicht-Newmann algorithm for the 2D system of Eq. (19) (left panels, from top to bottom $t = 5, 50, 1500$) for the Chua circuit (centers panels, from top to bottom $t = 5, 50, 500$) and for Lorenz 63 (right panels, from top to bottom $t = 5, 50, 100$).

and dynamical behaviour of the real one.

Using a time-discretized version of Eq. (1), $x_{n+1} = f(x_n)$, and the conservation of probability, we can write the probability for the system to be in a region of the phase-space Σ after $n+1$ time steps as

$$\int_\Sigma \rho_{n+1}(x)dx = \sum_{k=1}^{M} \int_{\Sigma_k = f_k^{-1}(\Sigma)} \rho_n(x)dx = \sum_{k=1}^{M} \int_\Sigma \frac{\rho_n(f_j^{-1}(y))}{|J(f_j^{-1}(y))|}dy, \tag{20}$$

where we summed over all the M preimages of $x = f_k^{-1}(y)$ under f and J is the Jacobian determinant of the transformation. Eq. (20) becomes

$$\rho_{n+1}(x) = \sum_{k=1}^{M} \frac{\rho_n(f_k^{-1}(x))}{|J(f_k^{-1}(x))|}. \tag{21}$$

On the coarse-grained system the previous equation becomes

$$\begin{aligned} \rho_{n+1}^i &= \sum_{k=1}^{M} \frac{1}{|J(f_k^{-1}(C_i))|} \sum_{j=1}^{N} p_k^i(j)\rho_n^j \\ &= \sum_{j=1}^{N} \left(\sum_{k=1}^{M} \frac{1}{|J(f_k^{-1}(C_i))|} p_k^i(j) \right) \rho_n^j \\ &= \sum_{j=1}^{N} P_{ij}\rho_n^j \end{aligned} \tag{22}$$

with C_i the center of the ith cluster and $p_k^i(j)$ the probability for the preimmage $f_k^{-1}(C_i)$ of being mapped in cluster j. We assumed that the preimages of all the points in the phase-space mapped into the ith cluster are in a neighborhood of the $\tilde{C}_j$s, defined as the centers of the M preimages of each cluster obtained taking the limit $N \to \infty$. If the system is instead stochastic, this assumption is modified by saying that the probability for a preimages not being in a neighborhood of one of the $\tilde{C}_j$s is negligible.

We can then write the following relation between the deterministic forcing f and the Perron-Frobenious operator

$$P_{ij} = \sum_{k=1}^{M} \frac{1}{|J(f_k^{-1}(C_i))|} p_k^i(j). \tag{23}$$

We can notice that if $k = 1$, $p^i(j)$ coincides with the transpose of the adjoint of the Perron-Frobenious operator (the Koopman operator). In this case the Jacobian determinant can be estimated directly from the transition matrix. If $k > 0$ we define

$$A_i = \sum_{j=1}^{N} P_{ij} = \sum_{k=1}^{M} \frac{1}{|J(f_k^{-1}(C_i))|} \sum_{j=1}^{N} p_k^i(j) = \sum_{k=1}^{M} \frac{1}{|J(f_k^{-1}(C_i))|}, \tag{24}$$

and

$$B_j = \max_i P_{ij} = \max_i \sum_{k=1}^{M} \frac{1}{|J(f_k^{-1}(C_i))|} p_k^i(j) \leq \frac{1}{|J(C_j)|}. \tag{25}$$

While the first relationship is not affected by noise, increasing the noise magnitude will determine an increase in the standard deviation of the $p_k^i(j)$s, lowering in this case their maximum value. If f^{-1} is not constant, there will be some regions of the image mapped in a smaller region of the preimage and vice versa. Therefore, by artificially adding white noise to the system, since the number of clusters is finite, we will observe that some of the B_js will remain unchanged, while others will be affected. B_j can be extrapolated from the the values that are not changed by noise. In these cases $B_j \simeq \frac{1}{|J(C_j)|}$ and the Jacobian determinant can be obtained.

We have then found the searched relationship between the forcing f and the Perron-Frobenious operator.

In the following we apply the method described above to different one-dimensional dynamical systems and in each of them we will study the effect of adding white noise in the determination of the functions A_i, B_j.

We consider the Ulam map

$$x_{n+1} = 1 - 2x_n^2, \tag{26}$$

defined on the phase space $x_n \in [-1, 1]\ \forall n$. We have

$$\rho_{n+1}(C_i) = \frac{1}{4}\left(\frac{2}{1-C_i}\right)^{\frac{1}{2}}\left[\rho_n\left(\left(\frac{1-C_i}{2}\right)^{\frac{1}{2}}\right) + \rho_n\left(-\left(\frac{1-C_i}{2}\right)^{\frac{1}{2}}\right)\right] \tag{27}$$

and then

$$\sum_k \frac{1}{|J(f_k^{-1}(C_i))|} = \frac{\sqrt{2}}{2}\left(\frac{1}{1-C_i}\right)^{\frac{1}{2}} \tag{28}$$

and

$$\frac{1}{|J(C_j)|} = \frac{1}{4|C_j|}. \tag{29}$$

We study a stochastic version of the Ulam map obtained by adding at each time step a Gaussian random number ξ_n with zero mean and variance σ. Since the process is defined in the closed interval $[-1, 1]$, the values of ξ_n that bring the system outside the interval are discarded. We integrated the system numerically, we clusterized it, we constructed the Perron-Frobenius operator from the trajectory and then the functions A_i and B_j.

From a least square fit of A_i and B_j we can estimate Eqs. (28, 41) and the Jacobian determinant (or the derivative of $f(x)$ as in the one-dimensional case we are considering).

The same procedure has been applied also for the continued fraction map and the cusp map. For the former we have

$$f(x) = \frac{1}{x} - \left\lfloor \frac{1}{x} \right\rfloor \tag{30}$$

defined on the phase space $x_n \in [0, 1]\ \forall n$,

$$\rho_{n+1}(C_i) = \sum_{k=1}^{\infty} \frac{1}{(k+C_i)^2}\rho_n\left(\frac{1}{C_i+k}\right), \tag{31}$$

$$\sum_k \frac{1}{|J(f_k^{-1}(C_i))|} = \Psi^1(1+C_i) \tag{32}$$

and

$$\frac{1}{|J(C_j)|} = C_j^2, \tag{33}$$

with $\Psi^m(x)$ the polygamma function of order m and $\lfloor x \rfloor$ the integer part of x.

For the latter we have

$$f(x) = 1 - 2|x|^{\frac{1}{2}} \tag{34}$$

defined on the phase space $x_n \in [-1, 1]\ \forall n$,

$$\rho_{n+1}(C_i) = \frac{1 - C_i}{2}\left[\rho_n\left(\frac{(1-C_i)^2}{4}\right) + \rho_n\left(-\frac{(1-C_i)^2}{2}\right)\right] \tag{35}$$

$$\sum_k \frac{1}{|J(f_k^{-1}(C_i))|} = 1 + C_i \tag{36}$$

and

$$\frac{1}{|J(C_j)|} = \sqrt{|C_j|}. \tag{37}$$

Finally, we consider the logistic map

$$x_{n+1} = r x_n (1 - x_n), \tag{38}$$

for $r > 0$. We have

$$\begin{aligned}\rho_{n+1}(C_i) =& \frac{1}{\sqrt{r^2 - 4rC_i}}\rho_n\left(\frac{\sqrt{r^2 - 4rC_i}}{2r}\right) \\ &+ \frac{1}{\sqrt{r^2 - 4rC_i}}\rho_n\left(-\frac{\sqrt{r^2 - 4rC_i}}{2r}\right)\end{aligned} \tag{39}$$

and then

$$\sum_k \frac{1}{|J(f_k^{-1}(C_i))|} = \frac{2}{\sqrt{r^2 - 4rC_i}} \tag{40}$$

and

$$\frac{1}{|J(C_j)|} = \frac{1}{|r(1 - 2C_j)|}. \tag{41}$$

In Fig.(6) we plotted the functions A_i and B_j for each map together with their analytical estimations reported in Eqs. (28, 41, 32, 33, 36, 37). We used different noise amplitudes and we can observe how, in all the cases, the function A_i correctly reproduces its expected value in the zero noise limit. The function B_j instead is deeply affected by noise, underestimating the value of the Jacobian determinant when the noise amplitude is increased. In this case we can notice that for come values of C_j B_j is less affected by noise than others, these values corresponds to regions where the preimages are denser than the corresponding images. These regions can be identified by adding artificial noise to the system and identifying the values of B_j less sensible to noise and perform the fit over those.

Using the method described above, We showed how to reconstruct the equation governing the dynamics of the system from the Perron-Frobenious operator for a one-dimensional

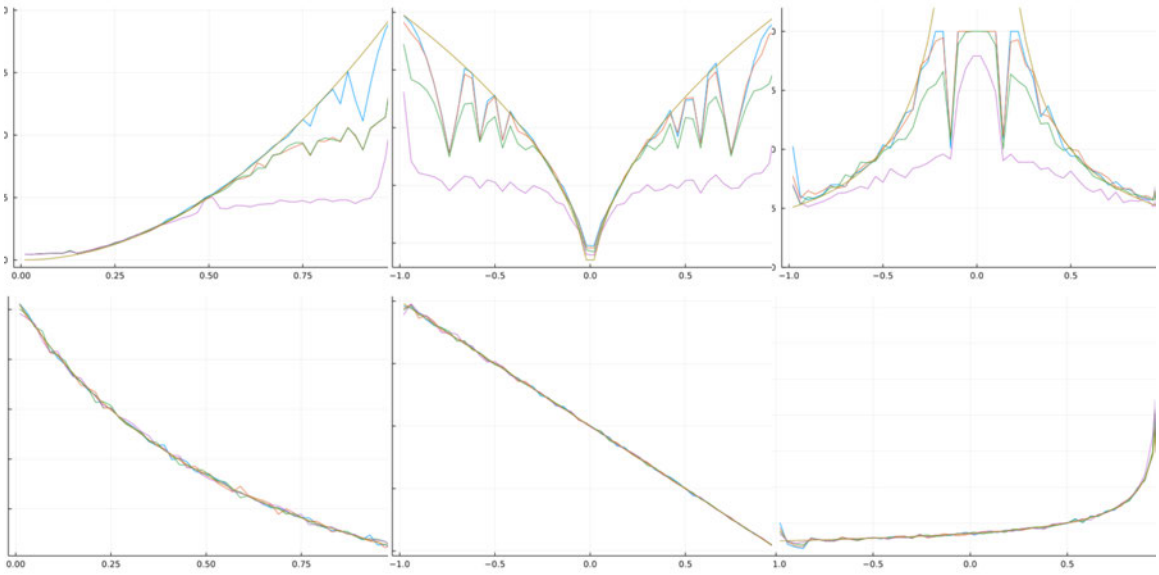


Figure 6: Plot of A_i and B_j (top and bottom rows, respectively) together with their analytical estimation (yellow lines) for the tree different maps studied (from left to right, Ulam, continued fraction and cusp maps). Different values of the noise amplitude have been used $\sigma = 0, 0.02, 0.05, 0.1$ (blue, orange, green and purple lines, respectively).

process. For more complex multi-dimensional systems it will be impossible; however, by using the relationship between the matrix elements of P and f, one can have a reliable estimation of the first order partial derivatives of f which are useful constraints that can be used to reconstruct f from a time series.

We used this method to reconstruct the logistic map with parameter $r = 4$ from data. We determined the Jacobian determinant from the Perron-Frobenious operator and we used this information to train a neural network imposing that the hidden state Jacobian determinant agrees with the one obtained from the Perron-Frobenious operator. From Fig. (8) we can notice that the hidden state of the neural network is able to correctly reproduce the map and the statistics of its time series.

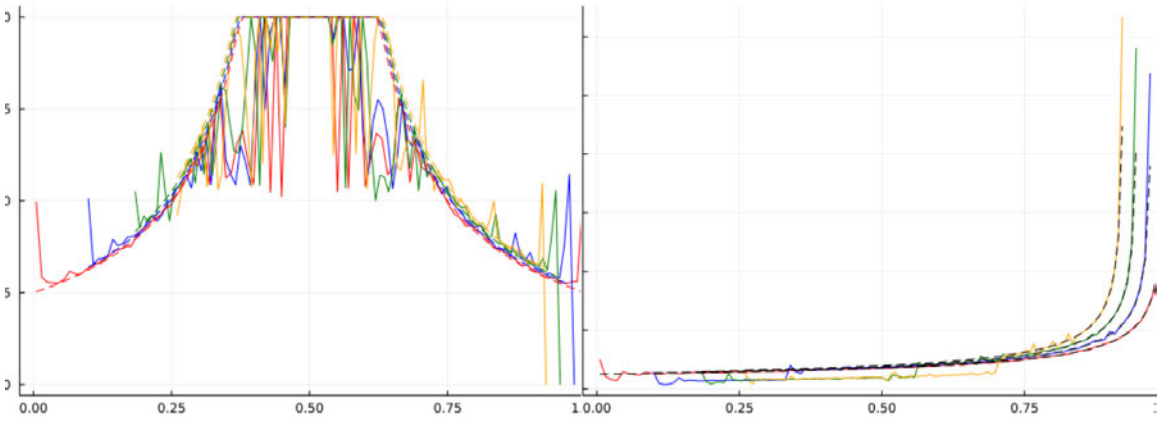


Figure 7: Same as Fig. (6) but for the logistic map for different values of $r = 3.7, 3.8, 3.9, 4$ (red, blue, green and orange curves, respectively).

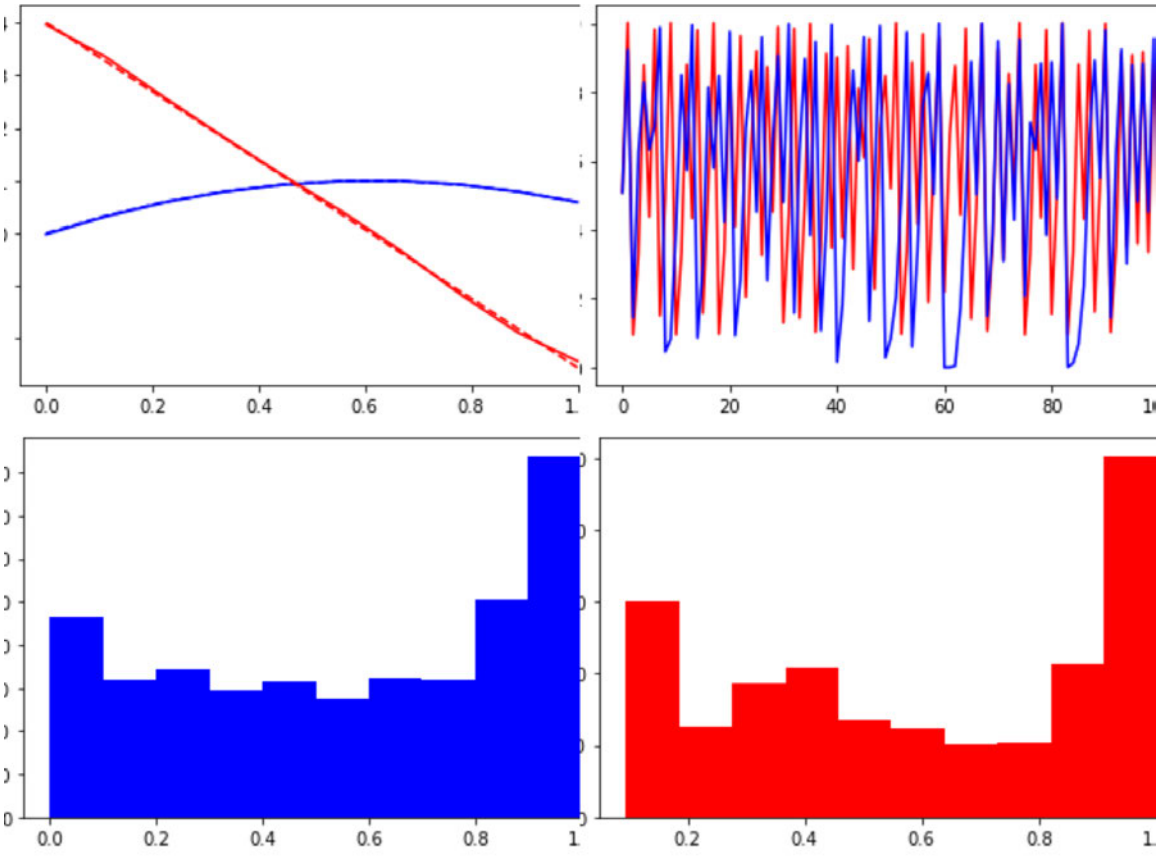


Figure 8: From left to right and top to bottom: Panel 1: reconstruction of the logistic map (blue dashed curve) and its Jacobian determinant (red dashed curve) using a neural network (solid lines), Panel 2: sample of the logistic map evolved using Eq. (38) and a neural network (red and blue curve, respectively), Panel 3,4: p.d.f.s of the trajectories of the previous panel.

Appendix

Here we present the dynamical systems we referred in the manuscript

The Kuramoto–Sivashinsky equation

$$u_t + u_{xx} + u_{xxxx} + \frac{1}{2}u_x^2 = 0. \tag{42}$$

Lorenz 63 system

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= -xz + \rho x - y \\ \dot{z} &= xy - \beta z \end{aligned} \tag{43}$$

with $\sigma = 10$, $\rho = 28$ and $\beta = 8/3$.

Chua system

$$\begin{aligned} \dot{x} &= a[y - h(x)] \\ \dot{y} &= x - y + z \\ \dot{z} &= by \end{aligned} \tag{44}$$

where

$$h(x) = m_1 x + \frac{1}{2}(m_0 - m_1)(|x + 1| - |x - 1|) \tag{45}$$

with $a = 15.6$, 25.58, $m_0 = -8/7$ and $m_1 = -5/7$.

Acknowledgment

I would like to thank Peter Schmid and Andre Souza for their thorough supervision and support, Predrag Civitanovic and Mauricio Barahona for their fruitful discussions and all the GFD staff and fellows for the great summer.

References

[1] S. KLUS, P. KOLTAI, AND C. SCHUTTE, *On the numerical approximation of the Perron-Frobenius and Koopman operator,* J. Comp. Dyn., 51, (2016), PP. 51-79.

[2] E. A. LEICHT AND M. E. J. NEWMAN, *Community structure in directed networks,* Phys. Rev. Lett. 100 (2008), 118703, DOI: https://doi.org/10.1103/PhysRevLett.100.118703.

Experiments on the Instability of Buoyancy-driven Coastal Currents

Sam Lewin

1 Introduction

In a rotating flow with a lateral boundary, such as a coastline in the ocean, the associated no-flux boundary condition removes the component of the Coriolis force parallel to the boundary, thus favouring the along-boundary spread of fluid. Coriolis forces deflect such 'coastal currents' to flow with the boundary on their right in the Northern hemisphere (equivalently on the left in the Southern hemisphere). A common driving force for coastal currents is the buoyancy force arising from the difference in density between two water masses, which occurs naturally when a freshwater river runs out into the ocean. Classical examples include the Leeuwin Current [16], the Chesapeake Bay Outflow [14], the Norwegian Coastal Current [12] and the East Greenland Current [21]. The evolution of buoyancy-driven coastal currents has been studied extensively over the last 45 years through a combination of theoretical, experimental and numerical studies. Of particular interest is the tendency of these currents to become unstable to wave-like disturbances, which may grow to form meanders and sometimes detaching eddies.

Describing buoyant outflows in full generality is difficult because the dynamics are highly varied near the source [11] and nose [6] of the current. Moreover, the behaviour of the buoyant fluid is strongly dependent on how it interacts with sloping bottom bathymetry [23, 15]. To simplify matters, it is common to focus on a region downstream of the source and upstream of the nose where the current is assumed to be steady and geostrophically balanced. Although the assumption that such a region exists at all is questionable [4], it provides a natural starting point for a theoretical stability analysis of the flow.

Buoyant coastal currents above a finite-depth ambient fluid are unstable to both baroclinic and barotropic instability, with the dominant mode depending on the parameter $\gamma = h_1/h_2$, which measures the ratio of the maximum depth of the current h_1 to the depth of the ambient fluid h_2 [8]. The case of a flat bottom was first studied experimentally by Griffiths & Linden [7] (hereafter GL81), who showed that a purely baroclinic two-layer quasi-geostrophic model (i.e. neglecting the lateral density front where the buoyant current outcrops at the surface) captured the essential features of the primary linear instability reasonably well for $0.07 \leq \gamma \leq 1$. The surprising effectiveness of the quasi-geostrophic approximation in this context can also be verified theoretically by comparison with a shallow water model which captures the outcropping front [1]. Though we will not consider it here,

we note that later work has demonstrated that instability may be significantly modified by variable bathymetry [2, 22, 19].

A sufficient condition for the baroclinic instability of a buoyant gravity current in geostrophic balance is that the lateral width of the current w_0 be suitably wide relative to the Rossby radius of deformation $L_R = \sqrt{g'h_1}/f$, where g' is the reduced gravity and f is the Coriolis parameter which is taken to be constant, whilst the depth ratio γ must also be suitably large. The measured properties of the instability, such as its wavelength and propagation velocity, are therefore expected to be a function of the parameter $F = w_0^2/L_R^2$ (which is often referred to as the Froude number in the literature, but might more appropriately be considered as an inverse Burger number [3]), as well as the parameter γ. However, there is an implicit assumption that the geostrophically balanced current evolves to be sufficiently wide in the first place. Indeed, experimental set-ups are often controlled so that the outflow current is either widening continuously [7, 2] (achieved by use of a ring source), or starts off in a supercritically wide state [9] (achieved by dam-break). This has the advantage of greater control over the experimental parameters important for instability, but it is doubtful as to whether such flow states can be accessed by real river plumes.

Modelling the river outflow as a point source at the boundary, Thomas & Linden [20] (hereafter TL07) derive an analytical solution predicting that the steady current evolves to have $F = 2$, a value that corresponds to stability or unresolvably long waves in the majority of existing experimental set-ups. Though this model is built on several highly simplifying assumptions (which will be discussed later), it nonetheless raises the question: under what conditions, if any, can we expect realistic river outflows (i.e. those that emerge from a localised source) to evolve to become sufficiently wide to be unstable to baroclinic instability?

To attempt to answer this question, we perform laboratory experiments in a cylindrical rotating tank with the buoyant gravity current produced by a small source at the boundary so that its width and depth adjust freely according to the source flux Q, the initial depth of the ambient fluid H, and the parameters g' and f. In agreement with TL07, we find that, for small values of the depth ratio γ, the current never becomes sufficiently wide to be reliably unstable. To access regimes of instability, we find it is necessary to greatly increase γ by reducing the depth of the bottom layer, to the point where it becomes important to consider the flow generated in this layer from the displacement by the buoyant current. This means that the instability accesses a very different regime of parameter space to most existing experimental studies, characterised by small Froude number F and large depth ratio γ.

We also present some qualitative results regarding the modification of the instability by wavy lateral boundaries, as might arise in the ocean in the form of bays, headlands and other coastal features. In the case of a sinusoidal boundary with characteristic wavelength Λ, it is natural to expect that the behaviour of a growing instability of wavelength λ may be different depending on the ratio λ/Λ. In the case $\lambda/\Lambda \ll 1$ or $\lambda/\Lambda \gg 1$, the instability is unlikely to 'feel' the influence of the boundary. When $\lambda \sim \Lambda$, however, we might expect the possibility of resonance.

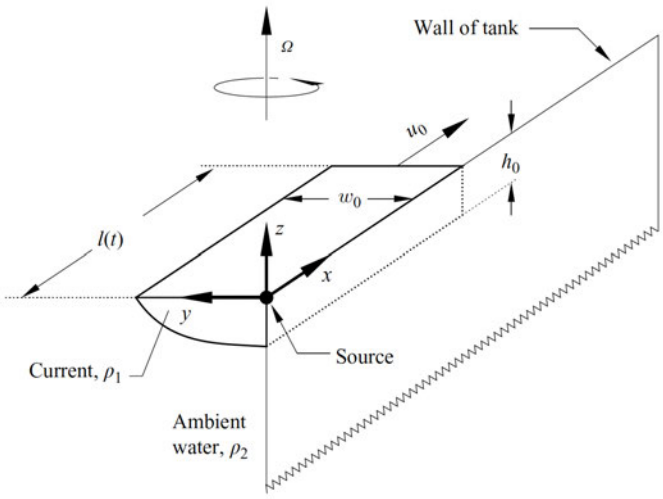


Figure 1: Schematic illustrating the theoretical model of the current used (placeholder from TL07).

2 Theory

We neglect the curvature of the cylindrical tank wall and model the flow as an inviscid current of density ρ_1 flowing steadily and uniformly along a wall at $y = 0$, with velocity $\mathbf{u}_1 = (u_1(y, z), 0, 0)$ in the along-wall x-direction and depth $h_1(y)$. The density, velocity and depth of the fluid underneath are similarly then ρ_2, $\mathbf{u}_2 = (u_2(y, z), 0, 0)$ and $h_2(y)$. The system is assumed to be rotating about the z-axis with constant angular velocity $\Omega = f/2$. Reduced gravity is defined as $g' = 2g(\rho_2 - \rho_1)/(\rho_2 + \rho_1)$, where g is the regular acceleration due to gravity. In the absence of the buoyant current, the ambient height of the bottom fluid is denoted H. We assume a rigid lid, that is, that the displacement of the free surface is small relative to $h_1(y)$, so that we can write $h_2(y) = H - h_1(y)$. The depth of the current is defined to be $h_0 = h_1(0)$ i.e. the depth at the wall, whilst the width of the current w_0 is defined by the 'outcropping' boundary condition $h_1(w_0) = 0$. The (upper layer) Rossby radius associated with the current, the corresponding Froude number F, and the layer depth ratio γ are then defined as

$$L_R = \frac{\sqrt{g'h_0}}{f}, \quad F = \left(\frac{w_0}{L_R}\right)^2, \quad \gamma = \frac{h_0}{H - h_0}. \tag{1}$$

Flow is assumed to originate from a point source far upstream with volume flux Q.

2.1 A quasi-geostrophic model for instability

To conduct a stability analysis, GL81 further simplify the flow geometry, constraining the flow to a channel of width w_0. The outcropping front is neglected so that the density interface intersects $y = w_0$ at some finite height, and the velocities $u_1 = U_1$, $u_2 = U_2$ are assumed to be constant, giving rise to a vertical shear $U = U_2 - U_1$. The equilibrium heights of each layer are $h_1 = h_0$ and $h_2 = H - h_0$. In the quasi-geostrophic approximation,

perturbations to the interface are included as terms in the total streamfunction for the velocities: $h_i(x,y,t) = (g'/f)\psi_i(x,y,t)$, where

$$\psi_i = \varphi_i + \phi_i(x,y,t), \quad \varphi_i = -U_i y. \tag{2}$$

The linearised equations for the perturbation streamfunctions $\phi_i(x,y,t)$ are (see e.g. [17])

$$\left(\frac{\partial}{\partial t} + U_1\frac{\partial}{\partial x}\right)\left[\nabla^2\phi_1 - F(\phi_1 - \phi_2)\right] + FU\frac{\partial \phi_1}{\partial x} = -\frac{r}{2}\nabla^2\phi_1, \tag{3}$$

$$\left(\frac{\partial}{\partial t} + U_2\frac{\partial}{\partial x}\right)\left[\nabla^2\phi_2 - \gamma F(\phi_2 - \phi_1)\right] - \gamma FU\frac{\partial \phi_2}{\partial x} = -\frac{\gamma r}{2}\nabla^2\phi_2. \tag{4}$$

The terms on the right hand side describe friction due to the horizontal top, bottom and interfacial boundaries in the case where interfacial friction dominates, where the constant r can be written in terms of an upper layer Ekman number (see GL81 for details).

By seeking solutions of the form $\phi_i = A_i \exp\left[ik(x-ct)\right]\cos(\ell y)$ and substituting into the linearised equations, it is possible to determine $\mathrm{Im}(c) = g(k,\ell)$. The calculations are detailed in GL81; here, it suffices to say that viscosity introduces a minimum shear U that must be exceeded in order for instability to exist, i.e. $g(k,\ell) > 0$. Provided U is sufficiently large, there is then a finite band of wavenumbers $K = \sqrt{k^2+\ell^2}$ for which instability is possible. Within this band, the most unstable mode $\min_{k,\ell} g(k,l)$ has $\ell = \ell_m = \pi/2$ and the corresponding $k = k_m$ is given by $k_m^2 = K_m^2 - \ell_m^2$, where

$$K_m = \mathrm{argmin}_K \frac{\gamma\sigma(K)K^2}{4\gamma(K^2-\ell_m^2)G(K)}, \tag{5}$$

$$G(K) = \gamma(K^2 + F(\gamma+1))(F+K^2)^2 - (K^2+2\gamma F)(F+K^2)\sqrt{\sigma(K)} + F\sigma(K), \tag{6}$$

$$\sigma(K) = (K^2+\gamma F)^2 + \gamma^2(K^2+F)^2 + 2\gamma(K^2+F)(K^2+\gamma F). \tag{7}$$

Note the quoted formula for K_m is incorrect in the original GL81 text: the above is the corrected version. In theory, for a steady current of depth h_0 and width w_0, we can determine F and γ via (1) and hence the expected wavelength K_m of instability.

2.2 An inviscid model for the current

A natural way to think about the inviscid problem is in terms of potential vorticity (PV) in the upper and lower layer, defined as $q_i(y) = (f + \zeta_i(y))/h_i(y)$, where $\zeta_i = -\partial u_i/\partial y$ is the vertical vorticity. It is quite common (see e.g. TL07) to assume that the PV of the discharging upper layer is vanishingly small. In the experimental set-up (described below), this is partially justified by the fact that the discharging buoyant fluid originates from a deeper reservoir. In the ocean, currents may 'lift-off' the base of the source channel as the river discharges and reduces in depth, thus again falling roughly in line with the assumption. However, we stress that the validty of this assumption is at best highly questionable and we consider it as a starting point for the sake of simplicity. A treatment of non-vanishing PV in the upper layer can be found in [4] for the case where the lower layer is inactive. The case for an active lower layer is left as an important consideration for future study.

As the buoyant fluid enters and displaces the ambient denser fluid, its effective height decreases (assuming a deep upstream reservoir) and hence an anticyclonic flow in the positive x direction is established to maintain zero PV. Similarly, the height of the displaced lower layer decreases from its initial height H which also generates an anticyclonic flow to conserve PV. The relative vertical shear between the two layers gives rise to the possibility of instability as described in §2.1. Precisely, conservation of PV q_i in the upper and lower layer gives us

$$q_1 = \frac{f - \partial u_1/\partial y}{h_1} = 0, \quad q_2 = \frac{f - \partial u_2/\partial y}{H - h_1} = \frac{f}{H}, \tag{8}$$

Hence, we have that $u_1(y) = -fy + u_1(0)$. Appealing to the fact that the velocity perpendicular to the wall is zero and $h = h(y)$, it follows from the shallow water equations that $Du/Dt = 0$, i.e., the velocity of the current does not change following the flow. For a point source, the velocity parallel to the wall is zero at the source and hence $u_1(0) = 0$.

We assume the Margules equations govern the evolution of the front, that is, the slope of the density interface is balanced by the difference in velocity between the layers:

$$f(u_1 - u_2) = g' \frac{\mathrm{d}h}{\mathrm{d}y}. \tag{9}$$

Taking a partial derivative with respect to y and substituting in from (8), we find h satisfies the following second order ordinary differential equation:

$$\frac{\partial^2 h}{\partial y^2} = \frac{f^2}{g'H} h - \frac{f^2}{g'}. \tag{10}$$

We have the outcropping boundary condition $h(w_0) = 0$. For another boundary condition, we note by continuity that $u_2(w_0) = 0$ since there is no upper layer beyond $y = w_0$. Since $u_1(w_0) = -fw_0$, (9) then gives the boundary condition $h'(w_0) = -f^2 w_0/g'$. If we define $\mu^2 = f^2/(g'H)$, the solution to (10) is

$$h(y) = H\left[1 - \cosh(\mu(y - w_0)) - \mu w_0 \sinh(\mu(y - w_0))\right]. \tag{11}$$

Finally, the current width w_0 is determined by the condition that the volume flux in the current is constant:

$$\int_0^{w_0} uh(y)\, dy = Q. \tag{12}$$

Note that we can expand in powers of μ to find

$$h(y) = h(0) - \frac{H}{2}\mu^2 (y - w_0)^2 - H\mu^2 w_0 (y - w_0) + O(H\mu^3) = h_0 - \frac{f^2}{2g'} y^2 + O(\frac{1}{H}), \tag{13}$$

so that $h(y)$ tends towards the parabolic profile found by TL07 in the limit as $H \to \infty$. Note that, as H decreases, the solution (11) becomes invalid when $h(0) > H$: at this point the current becomes attached to the bottom. We note that it is expected that the solution will likely be inaccurate before this situation occurs due to frictional effects, where the height of the bottom layer is similar to the height of the bottom boundary layer.

The profile (11) is plotted in figure 2a) for a range of values of the ambient lower layer height H, at nominal values of the parameters f, g' and Q. The primary effect of an active

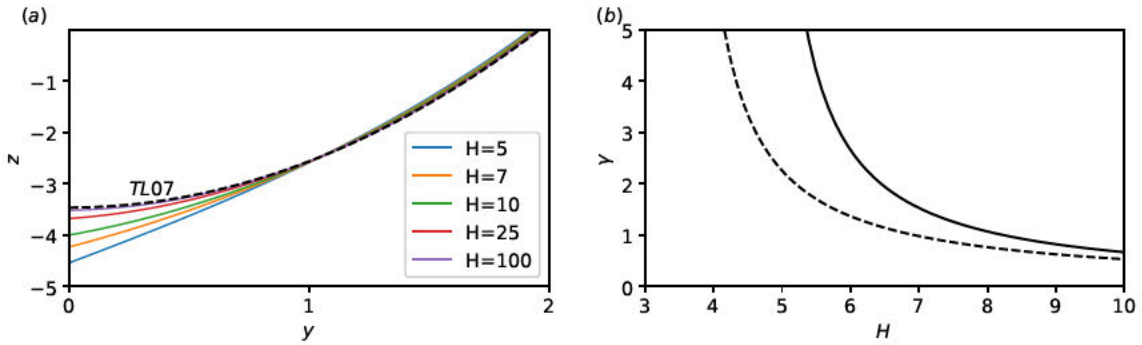


Figure 2: (a) Plots of the solution profile (11) for different values of H. (b) Variation of $\gamma = h(0)/(H - h(0))$ with the initial lower layer height H. For both panels, we take $f = 3$, $g' = 5$ and $Q = 10$. The solution from TL07 is shown by the black dashed line.

lower layer is that it permits a deeper upper layer for a given H. The current width w_0 is seen to decrease slightly, although remains very close to the TL07 value of $w_0 = (8g'Q/f^3)^{1/4}$. Figure 2b) shows how the value of the depth ratio γ changes with decreasing H. For small H, the change in h_0 becomes significant and large values of γ can be achieved for larger values of H than in the TL07 model with a passive bottom layer.

2.3 How do buoyant currents from a point source become unstable?

The TL07 model predicts $w_0 = (2gh_0/f^2)^{1/2} = \sqrt{2}L_R$ which gives a corresponding $F = (w_0/L_R)^2 = 2$. We combine this with the stability analysis from §2.1, as shown in figure 3. Curves showing the most unstable wavelength for a given F are plotted for various values of γ. The range of results from the ring source experiments of GL81 where the width of the current grows continuously in time are highlighted for reference. Recall these experiments have $0.07 \leq \gamma \leq 1$. The theoretical value for F predicted by TL07 falls well outside of the range of values of F measured in these experiments. It is seen that, for $\gamma < 1$, such currents are predicted to either be stable, or support very long waves that are difficult to measure in the lab due to size restrictions. Therefore, since, in theory, they do not grow in width over time, point source currents must have sufficiently small depth ratio γ in order to be unstable to measurable disturbances. In this regime, the motion of the bottom layer becomes important to the dynamics as discussed in §2.2, and as has recently been shown by [13] for a similar problem.

It is also important to note the precise mechanism by which ring source currents are proposed to become unstable. As the current grows in width and depth, so does the relative shear between the two layers. Friction restricts the growth of instability to occur only once as a minimum value of this shear has been achieved, at which point the wavelength of the intability is selected by the value of F according to marginal stability. However, if the critical shear is already achieved when the current develops, the instability is supercritical. Neglecting frictional effects, the wavelength of fastest growing supercritical baroclinic mode is $\lambda/w_0 = 2\pi F^{-1/2}\gamma^{-1/4}$ [7, 18]. The curves for the most unstable supercritical wavelength are shown by the dashed lines in figure 3. It is not immediately clear which mechanism

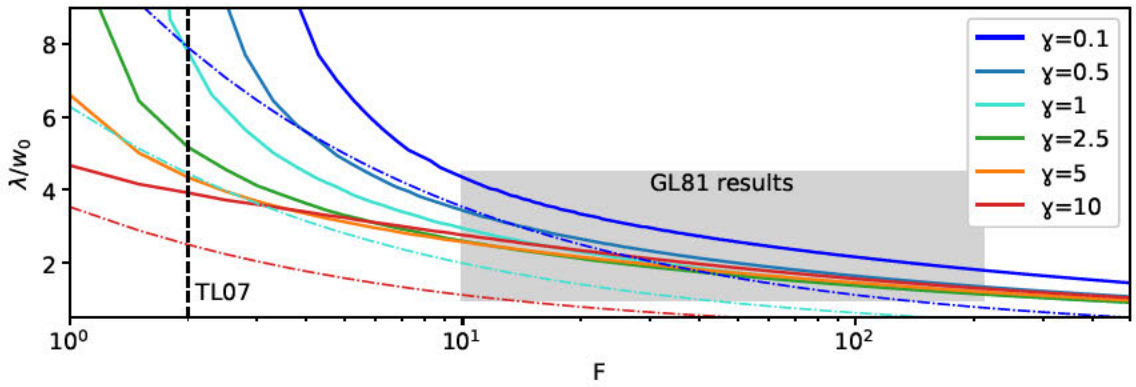


Figure 3: Solid lines show the instability curves from (5) for different values of γ. Dashed lines show the same instability curves but in the absence of friction, where the instability is now assumed to be supercritical. Here $\gamma/w_0 2\pi/k_m$ is the most unstable non-dimensional wavelength for a given $F = (w_0/L_R)^2$. The rough range of results from the ring-source experiments of GL81 are indicated by the grey rectangle. $F = 2$ corresponding to the TL07 prediction of w_0 is shown by the dashed vertical line.

is relevant to the point source experiments: this will depend on whether or not a critical shear between the two layers exists by the point at which the established current becomes sufficiently wide to be unstable.

Another limitation of the point source model is that it relies on the assumption of zero PV in the buoyant current. It has been shown that, for an infinitely deep bottom layer, a constant non-zero value of PV gives rise to wider currents [4] with larger F. This effectively favours measurable instability (i.e. smaller λ/w_0) for smaller values of γ as can be seen in figure 3.

To summarise the results of this section, we combine a model for a geostrophic buoyant gravity current originating from a point source with the classical stability analysis of GL81 in order to investigate in what regimes such currents may become unstable. In theory, given a current with specified f, g', Q and H we can predict the depth h_0, the width w_0 and hence the parameters γ and F that determine the stability properties. Due to the small predicted F, large $\gamma \geq 1$ is required for instability, a regime in which the motion of the bottom layer becomes important.

3 Experimental Set-up

Experiments were conducted in a 1m wide cylindrical tank mounted on the medium rotating table in the Geophysical Fluid Dynamics Laboratory at Woods Hole Oceanographic Institution. The tank was approximately 50cm deep with walls made of transparent perspex. A transparent plastic lid was placed on top during experiments to minimise wind stress on the surface of the fluid. The tank was filled with ocean salt water ($\rho_2 \approx 1.022\text{kg}\,\text{cm}^{-3}$) to a specified depth H. Densities were measured with a model DM058 Anton Paar densitometer. The table was then spun at a rotation rate $\Omega = f/2$ (note this caused the surface of the

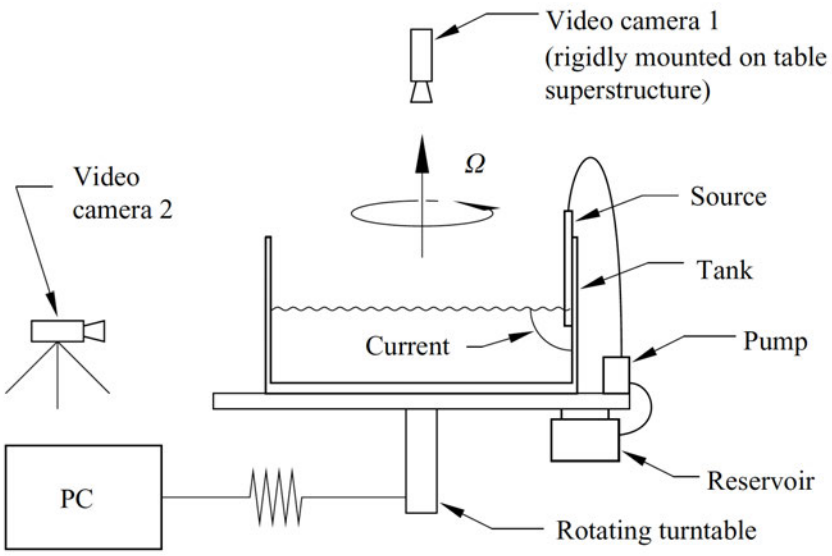


Figure 4: The experimental set-up (from TL07).

fluid to become concave and thus the height at the boundary to increase slightly; this 'true height' was the height recorded and was measured using a ruler attached to the side of the tank). Buoyant water of density ρ_1 - which was a mixture of salt water and fresh water - was injected with constant flux Q at the surface of the ambient using a metal pipe with a radius pf approximately 1 cm, which was attached to the side of the tank. The buoyant water was dyed red with a concentration of approximately $1\text{ml}1^{-1}$. The source pipe was connected to a large bucket that acted as the reservoir of buoyant water. A small piece of porous foam was placed around the end of the pipe to minimise mixing of the outflowing buoyant fluid with the ambient as it emerged. The source was turned on once the ambient salt water had reached solid body rotation; this took around 30 minutes. The current would then travel around the boundary of the tank in an anticlockwise direction, and the experiments were stopped when the nose of the current had travelled all the way around back to the source.

The evolution of the emerging dyed current was viewed from above and from the side using co-rotating cameras that took one picture each second. The camera viewing the side of the tank was placed to view the current just downstream of the source, where instability was generally first viewed to occur in those experiments that became unstable, and a ruler was placed in the shot as a reference for measuring the depth of the current. Post-processing was used to add a ruler to the shots from the camera above, which was possible knowing the width of the tank. To improve visualisation, the tank was lit from underneath using a uniform light source. Another method for enhancing visualisation was to very slowly inject a thin stream of blue dye at the surface using a syringe just downstream of the source. This dye would then be passively advected with the current and capture the motion of the instability. The wavelength of instability when it appeared was measured by averaging the number of instabilities (counted by eye) over the total distance they occupied around the circumference of the tank. Using the images available, it was possible to calculate, for

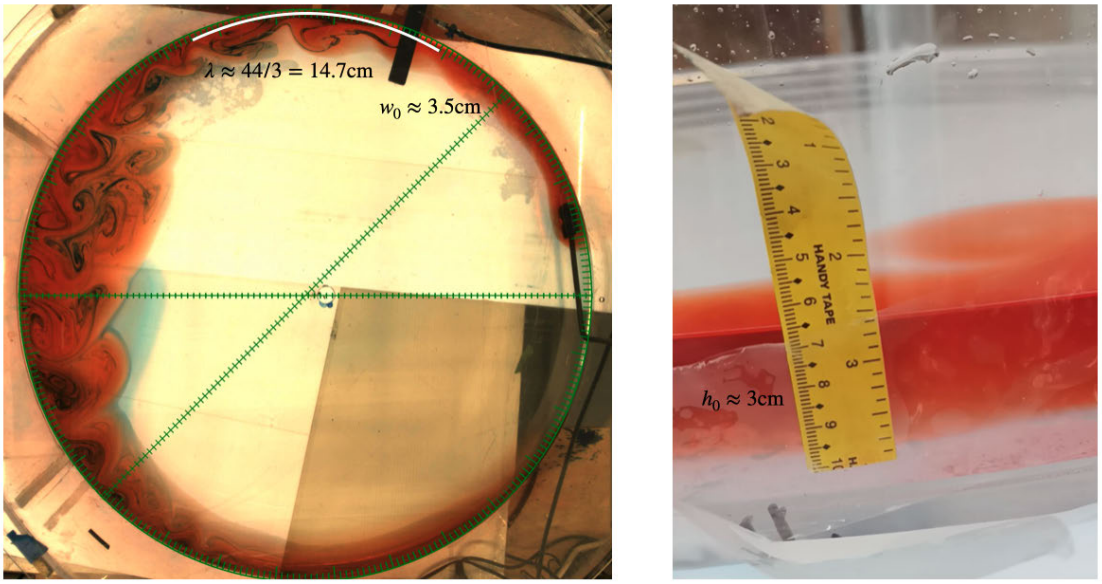


Figure 5: Example post-processed images demonstrating how various properties of the current and instability were measured. The left panel is a picture taken from above where the green rulers are added in post-processing so that the gap between each small tick represents 1cm. The right panel is a picture taken from the side-mounted camera measuring the depth of the current just downstream of the source.

example, the width of the current, the depth of the current at a single location close to the source downstream, the propagation velocity of the nose of the current and the wavelength of and propagation speed of instability. Figure 5 illustrates how some of these measurements were made in practise using the post-processed images.

One prominent difficulty was the tendency of the current to form a recirculating 'bulge' region near the source, as is typical in experiments (e.g. [10]) and indeed many realistic river outflows [11]. The problem with this feature for our purposes is that it reduces the effective flux of fluid into the geostrophic part of the current downstream in a manner that is hard to predict, thus making it difficult to test our theory which relies on a constant, known current flux. To mitigate this issue, a plexiglass barrier was placed around the source to prevent the bulge and force the flow along the wall, as in [19]. However, a caveat of this approach was then that the width of the current emerging at the edge of the barrier was determined by the width of the barrier. Hence, if the barrier was sufficiently far from the edge of the tank, the current could be supercritically wide and become unstable without, in theory, freely geostrophically adjusting. Thus, the width of the barrier W_{source} became an additional parameter in the experiments.

The data from a total of 45 experiments are used in the following analyses. Of these, 12 had a wavy lateral boundary (discussed below) and 33 had a smooth boundary. The range of parameters used in the experiments are shown in table 1.

Parameter	Value
f	1.5 - 3.5 s^{-1}
g'	3 - 10 $\mathrm{cm\,s}^{-2}$
Q	10 $\mathrm{cm}^3\,\mathrm{s}^{-1}$
H	3 - 22 cm
W_{source}	1 - 5 cm

Table 1: The range of parameter values used in the experiments.

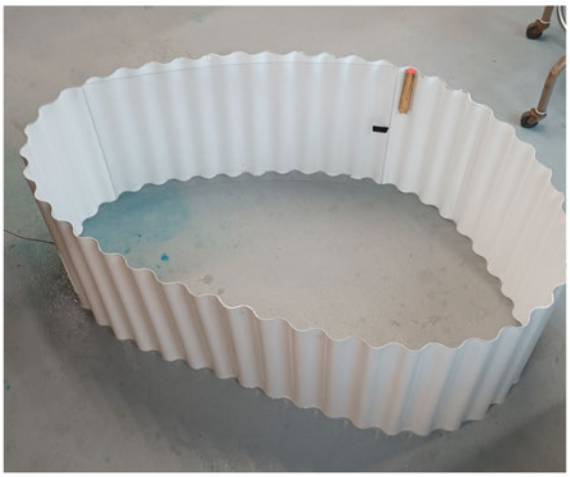


Figure 6: A picture of the wavy boundary, which consisted of a flexible sheet of smoothly corrugated plastic wrapped in a circle. This could be placed snugly inside the cylindrical tank. The source pump was attached to the smooth transparent plexiglass part of the wall, which can be seen in the photo.

3.1 Wavy wall experiments

A wavy lateral boundary was introduced using a sheet of smooth, roughly sinusoidally corrugated piece of plastic which wrapped around the inside of the vertical walls of the tank. The wavelength of the lateral topography was $\lambda_{\mathrm{topo}} = 7$ cm and the amplitude was $a_{\mathrm{topo}} = 0.7$ cm. A photo of the equipment used is shown in figure 6. Since we were limited to this one combination of λ_{topo} and a_{topo}, the ratio $\lambda/\lambda_{\mathrm{topo}}$ was modified exclusively using the parameters of the current to control the wavelength of the instability λ. Additionally, the topography was opaque meaning the height could not be measured using the side-on camera, hence the height of the current from an equivalent smooth-wall experiment was used. The topography was designed to be smooth around the source so that the current could be properly established before impinging on the lateral topography.

4 Results

4.1 Qualitative description of the smooth wall current

Figure 7 shows the evolution of a typical buoyant gravity current for a deeper ambient layer depth. The behaviour is very similar to the currents of TL07, with the depth and width of the current decreasing towards the nose and being widest near the source. In the region immediately downstream of the source, the width and height of the current adjust to remain roughly constant, suggesting that geostrophic balance is achieved. We will not consider in detail the thinner, shallower region closer to the nose since frictional effects are probably more important here: in particular we note a surface Ekman layer becomes visible at later times, as is seen by the weaker dye concentrations at the edge of the current further downstream. Some non-uniformities in the current emerge from the source region, which grow into slight meanders, as can be seen at $t = 180$s. However, these never develop into the well-formed anticyclonic bulges observed in GL81, and they emerge directly from the source rather than from the geostrophic region downstream, a behaviour that is not included in our model. Certainly, there is no consistent way to measure a wavelength and so we classify this flow as stable. There are several other experiments that behave very similarly, all with larger values of the ambient depth H.

Figure 8 shows the evolution of typical current above a shallower bottom layer. Even with the barrier, these experiments typically develop a 'bulge' region as they emerge, as is seen in the top left panel. This bulge circulates anticyclonically and eventually detaches, forming a large eddy which continues to grow and whose evolution can be clearly seen in the panels. However, this behaviour is transient and eventually a steady current width and depth are achieved downstream of the source, as can be seen in the bottom two panels. Clear waves of instability can be seen emerging at around the 12 o'clock position in the bottom two panels, which then propagate downstream. The width of the current is very similar to the equivalent current in figure 7 which has a larger H and hence smaller γ. The qualitative behaviour of the unstable current in figure 8 versus the stable current in figure 7 is consistent with the theoretical analysis of §2, which predicts that the key mechanism causing geostrophic 'point source' currents to go unstable is an increase in depth ratio γ,

4.2 Comparison with theory

4.2.1 Current shape

We compare the measured height of each current with the theoretical model (11) from §2, plotting the values against the ambient depth H in figure 9. Note that we cannot have $h_0 > H$: when $H < H^*$, where recall H^* is such that $h_0(H^*) = H^*$, the model is adjusted so that $h_0(H) = H$. To collapse the data, values are non-dimensionalised by $\hat{h} = (2Qf/g')^{1/2}$ i.e. the classical geostrophic prediction. There is an indication of an increasing trend in h_0 as H decreases, though the scatter is significant. There appears to be a systematic bias away from the geostrophic prediction $\hat{h}$, even for deeper ambient layers. The source of this bias was not clear. It could be caused by the fact that some fluid leaks backwards from the source and escapes around the rear of the barrier, reducing the effective flux into the current and hence its depth. In the case of a deep and inactive ambient fluid, breaking

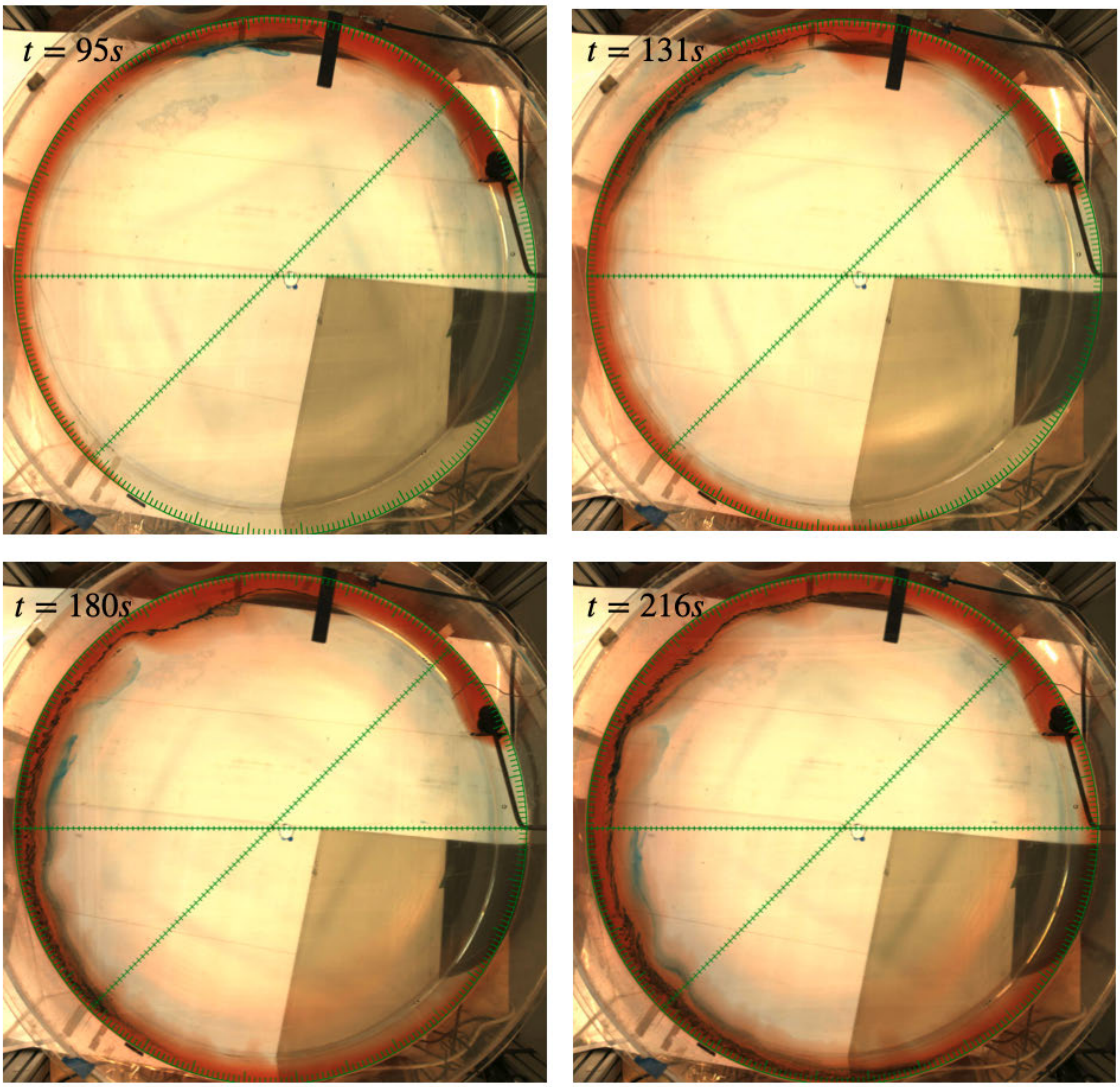


Figure 7: Sequence of snapshots showing the evolution of a typical stable buoyant current, with times shown in the top left corner of each picture. Parameters for this experiment were $f = 2.5\text{s}^{-1}$, $g = 4.99\text{cm}\,\text{s}^{-2}$, $Q = 10\text{cm}^3\text{s}^{-1}$, $H = 20.9\text{cm}$, $W_{\text{source}} = 3\text{cm}$. The measured depth of the current at the wall was $h_0 = 2.8\text{cm}$ giving a value of $\gamma = 0.14$.

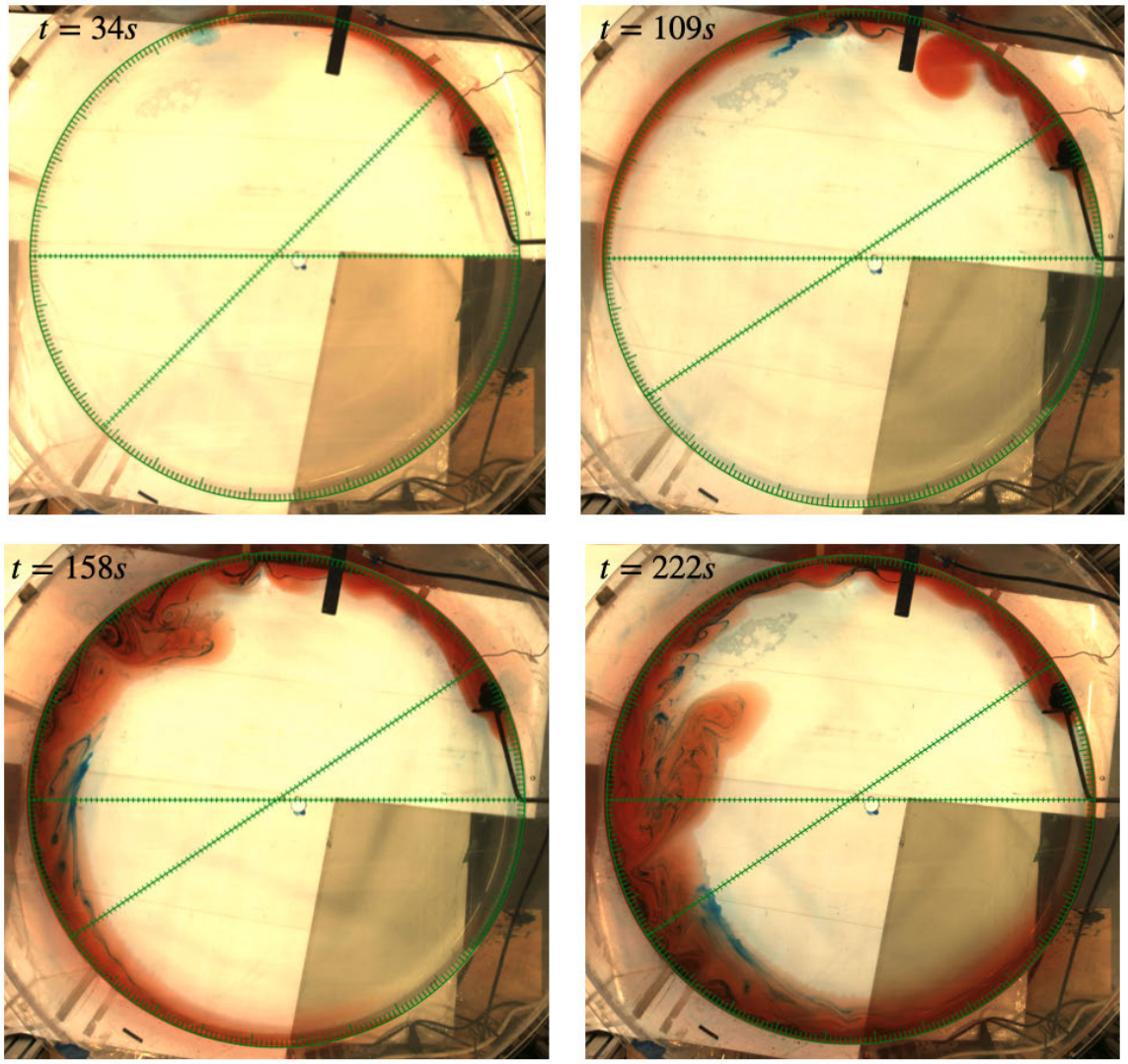


Figure 8: Sequence of snapshots showing the evolution of a typical unstable buoyant current, with times shown in the top left corner of each picture. Parameters for this experiment were $f = 2.5\text{s}^{-1}$, $g = 4.71\text{cm}\,\text{s}^{-2}$, $Q = 10\text{cm}^3\text{s}^{-1}$, $H = 5.93\text{cm}$, $W_{\text{source}} = 3\text{cm}$. The measured depth of the current at the wall was $h_0 = 3.0\text{cm}$ giving a value of $\gamma = 0.51$.

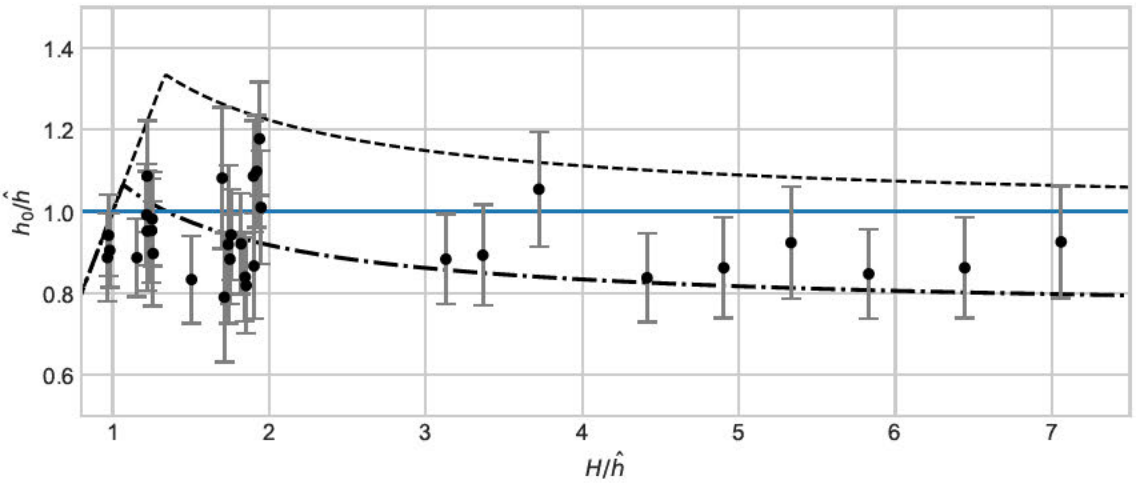


Figure 9: Experimental values of h_0 (black markers) vs theoretical predictions from (11) in §2.2 (dashed line). The dot-dashed line shows a rescaled version of the theoretical model where the scaling factor is 0.75. Both of these models are adjusted as described in the text so that we always have $h_0 \leq H$. Values are non-dimensionalised by $\hat{h} = (2Qf/g')^{1/2}$, and the thin blue line represents the classical prediction of $h = \hat{h}$.

the assumption of zero PV in the top layer does not change the prediction for the depth of the geostrophic current [4], so we pose that this does not contribute significantly to the uncertainty here since we see the error even for deep ambients. The same argument applies for frictional effects in the bottom layer. In any case, we find a rescaled version of the model in (11) is a better fit for the data, as is seen looking at the dot-dashed line in the figure. More data points, perhaps particularly for larger values of H, might be useful to confirm the apparent trend and fit in the figure. In practice, these deeper ambient experiments were limited due to the time taken to fill the tank to a larger H, and the amount of salt water it was possible to store in the laboratory overnight to equilibrate to the room temperature at once.

The measured current widths w_0 are compared to the theoretical model in figure 10, where we non-dimensionalise widths by the geostrophic value $\hat{W} = (8g'Q/f^3)^{1/4}$ from TL07. We note that almost all of the measured widths are bigger than both our theory and the theory of TL07 predicts. This was a feature also noted by TL07 and could be due to the precise shape of the surface profile of current compared to the bulk of the current underneath. A more likely explanation is probably the error due to the assumption of zero PV in the buoyant current [4]. Unfortunately, we had no way of measuring a cross-sectional profile or true PV of the current to test this. Nonetheless, there is no clear trend in w_0 as H decreases, which is consistent with our theory from §2.

4.2.2 Instability

In section §2, we outlined two possible mechanisms for instability: marginal instability and supercritical instability. The former has non-dimensional wavelength $\lambda/w_0 = 2\pi/k_m$ where

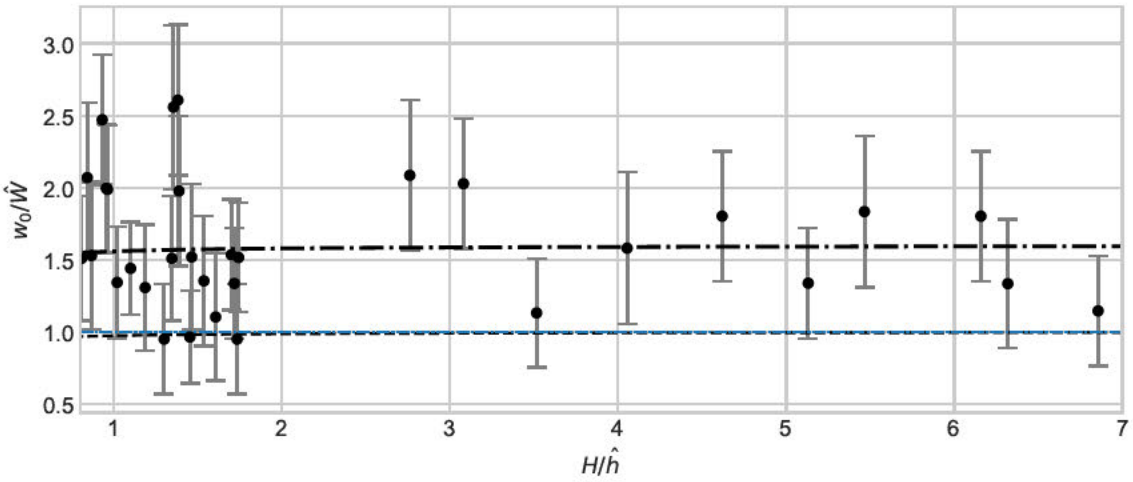


Figure 10: Experimental values of w_0 (black markers) vs theoretical predictions from the model in §2.2 (dashed line). The dot-dashed line shows a rescaled version of the theoretical model where the scaling factor is 1.5. Values of W are non-dimensionalised by $\hat{W} = (8g'Q/f^3)^{1/4}$, and the thin blue line represents the classical prediction of $W = \hat{W}$. Values of H are non-dimensionalised by $\hat{h}$ as in figure 9.

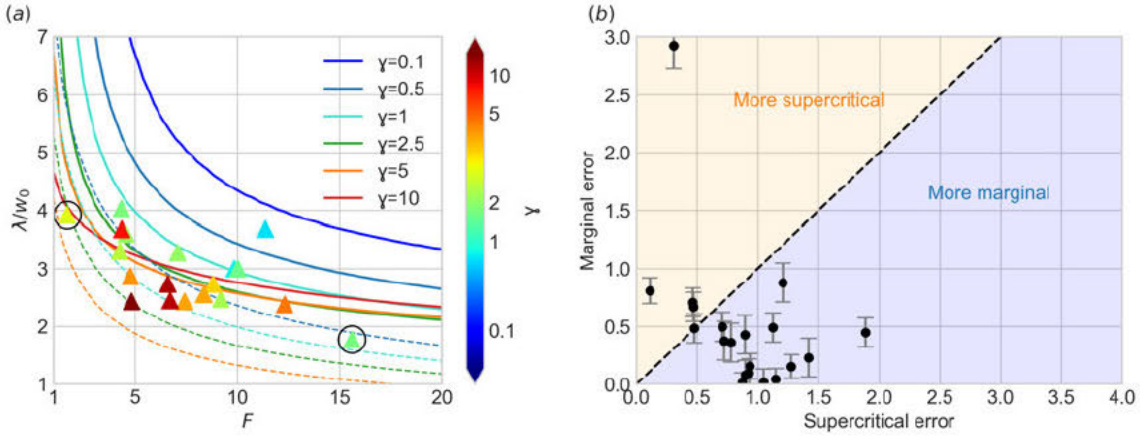


Figure 11: (a) Plots of measured instability wavelength λ normalised by the current width w_0 vs the measured Froude number $F = (w_0/L_R)^2$. Colours represent the corresponding value of the depth ratio γ. Solid lines and dashed lines are the marginal instability and supercritical instability curves from GL81 described in the text here, for values of γ whose range of colours correspond to the colour bar. The black circled points represent the two left-most points in (b), which are expected to be supercritically unstable. (b) Error between the measured value of λ/w_0 and the marginal and supercritical theoretical value calculated using the measured F and γ and the stability curves shown in (a).

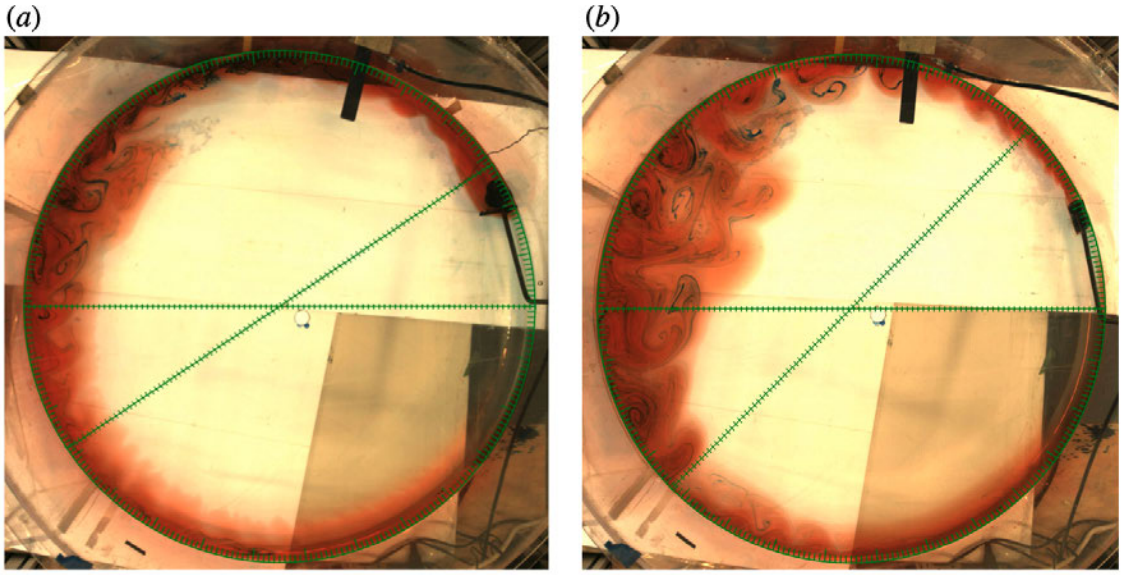


Figure 12: Post-processed images from above of the circled experiments in figure 11(a). (a) Experiment with $f = 3.5\text{s}^{-1}$, $g = 5.13\text{cm}\,\text{s}^{-2}$, $Q = 10\text{cm}^3\text{s}^{-1}$, $H = 6.8\text{cm}$, $W_{\text{source}} = 5\text{cm}$, with corresponding $F = 15.6$ and $\gamma = 0.8$. (b) Experiment with $f = 2.5\text{s}^{-1}$, $g = 9.36\text{cm}\,\text{s}^{-2}$, $Q = 10\text{cm}^3\text{s}^{-1}$, $H = 3.9\text{cm}$, $W_{\text{source}} = 2.5\text{cm}$, with corresponding $F = 1.7$ and $\gamma = 1.8$.

k_m is determined using (5), whilst the latter has $\lambda/w_0 = 2\pi F^{-1/2}\gamma^{-1/4}$. We compute the non-dimensional wavelength of instability for each current that becomes unstable in our experiments, plotting the results against the corresponding value of $F = (w_0/L_R)^2$ in figure 11a). Also included are the marginal and supercritical instability curves from figure 3. We first note that practically all of our experiments fall outside the range of ring source experiments performed by GL81, which are shown in figure 3. This highlights the fact that the point source currents are indeed in a very different regime due to their limited width. It is worth pointing out however that, due to the measured width being consistently larger than the theoretical predicted width, the predicted value of $F = 2$ is exceeded in most experiments and hence marginal instability with measurable wavelength is still possible. Indeed, we see that the experimental points are a good fit with the marginal instability curves. Additionally, there is a clear dependence of the location of the instability in parameter space on the depth ratio γ, a feature which has not been observed previously.

We also compare the location in parameter space to the instability curves for supercritical instability shown by the dashed lines in figure 11a). To get a quantitative measure of which instability is more relevant, given the measured value of γ and F for each point, we compute the predicted value of (non-dimensional) instability wavelength $\hat{\lambda}_{\text{super}}$ and $\hat{\lambda}_{\text{marg}}$ for the supercritical and marginal instabilities and calculate the difference from the measured value $\hat{\lambda} = \lambda/w_0$. The errors are shown in figure 11(b), where it can be seen that the majority of points lie closer to the marginal instability curve than the supercritical instability curve, suggesting that this is the dominant mechanism for instability.

There are two notable exceptions for which the supercritical instability prediction is very

accurate: these points are circled in black in figure 11(a). Pictures from the corresponding experiments are shown in figure 12. Figure 12(a) shows the experiment corresponding to the right circled point with $F = 15.6$, $\gamma = 0.8$, $\lambda/w_0 = 3.9$, which had a marginal error of 0.8. This experiment had a wide source barrier $W_{\text{source}}/w_0 = 3.9$ meaning the current emerging was supercritically wide, and instability can be seen emerging directly downstream of the barrier. Interestingly, the left circled point, which corresponds to an experiment with $F = 1.7$, $\gamma = 1.8$ and $\lambda/w_0 = 4$ shows a very different scenario, as seen in figure 12(b). This experiment had a narrow source barrier $W_{\text{source}} = 1.3$, so that large γ is the driver of instability. It appears that in this scenario, supercritical instability is the dominant mechanism. The instability forms further downstream of the source than in figure 12(a), and there is evidence of unsteady, small scale fluid motions at the edge of the current near the source. Interestingly, even though the geostrophic width of the current emerging from the source is narrower, the instabilities grow nonlinearly to a much greater amplitude and as a result more lateral mixing takes place.

In order for both of the currents in figure 12 to be supercritically unstable, the shear at the density interface must be sufficiently large. According to the geostrophic theory, the velocity of the upper layer at the interface is $u_1(w_0) = -fw_0$, so that the shear can be expected to be larger for wider currents. This may explain the strongly supercritical behaviour of the current in figure 12(a). For the current in figure 12(b), which is very narrow, we propose that there is an additional ageostrophic component of the velocity of the current emerging from the narrow source, which occurs due to an equivalent volume flux over a smaller area. Indeed, the small scale perturbations to the interface near the source may be due to barotropic instability growing in a region of high shear.

4.3 Current over a wavy boundary

Figure 13 shows the qualitative behaviour of instability for a wavy lateral boundary and a smooth wall in a parameter regime where $\lambda_{\text{topo}}/\lambda \approx 0.5$ for the smooth wall. As can be seen, there is very little observable difference between the two experiments: both have an initial transient instability that grows to a larger amplitude as was discussed above and in both cases the current reaches a roughly steady regime, becoming unstable downstream of the source. The wavelength of the instability is practically identical between the two experiments, despite the wavelength of the topography being half of the wavelength of instability. The nonlinear evolution of instability is also similar, with the anticyclonic waves growing to similar amplitudes for both the smooth and wavy walls. We found the qualitative differences between smooth and wavy wall experiments to be very small for all of the parameter choices presented here.

To investigate whether there is a quantitative difference between the wavy boundary experiments and the smooth boundary experiments, figure 14 shows the predicted wavelength of instability according to the marginal stability curve (5), versus the measured wavelength of instability, for both smooth and wavy wall experiments. The marginal stability prediction was shown to be a good fit to the smooth wall experiments in §4.2.2. In theory, if the stability in the wavy wall experiments is driven by the presence of the boundary we might expect to see the wavelength of the instability deviate from the marginal stability theory towards the wavelength of the boundary, as shown by the solid black line in the figure. Instead,

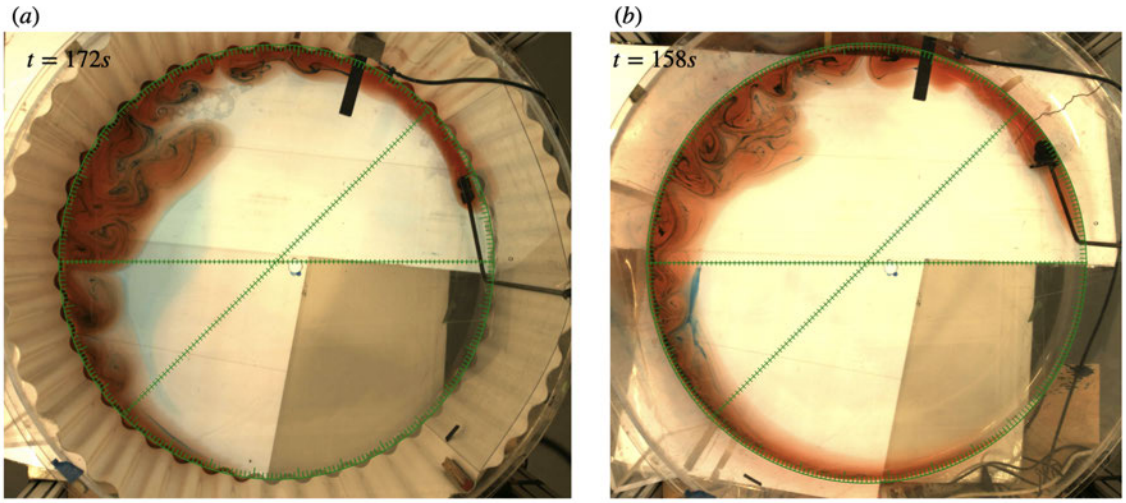


Figure 13: Pictures showing the evolution of instability for two experiments with identical parameters $f = 2.5\text{s}^{-1}$, $g = 4.7\text{cm}\,\text{s}^{-2}$, $Q = 10\text{cm}^3\text{s}^{-1}$, $H = 3.9\text{cm}$, for (a) a wavy wall and (b) a smooth wall.

there is no clear difference in trends between the smooth wall experiments and the wavy wall experiments. It is important to point out that the range of values of $\lambda_{\text{measured}}/\lambda_{\text{topo}}$ are limited in our experiments and we do not achieve the limiting regimes $\lambda_{\text{measured}}/\lambda_{\text{topo}} \ll 1$ and $\lambda_{\text{measured}}/\lambda_{\text{topo}} \gg 1$ described in the introduction, making it difficult to draw any broad conclusions about the behaviour for the wavy wall experiments. However, in the case $\lambda_{\text{measured}}/\lambda_{\text{topo}} \sim 1$ there is no clear qualitative or quantitative difference in behaviour.

To investigate further, we injected a small amount of blue dye into the current near the wall and mounted a GoPro Hero 7 camera just above the current downstream of the injection point to get a close-up view of the dynamics. A snapshot of the flow is shown in figure 15. The trajectory of the blue dye suggests that fluid is being trapped in recirculating 'bay' regions of the topography, with the current flowing over the top. This reduces the effective amplitude of the topography and may explain why we do not see large differences in flow evolution between the wavy wall and smooth wall experiments.

5 Conclusion

Laboratory experiments were conducted to investigate the instability of a rotating buoyant gravity current emerging from a localised source at the boundary. Combining existing theoretical models for the evolution and subsequent instability of such currents demonstrated that instability should only be achievable provided the ratio of the current depth to the bottom layer depth γ is sufficiently large. In this scenario, it is natural to expect that the motion of the bottom layer induced by the impinging current may be important to the dynamics. We proposed a simple theoretical model for an active bottom layer based on the work of TL07 that predicts the main effect of an active bottom layer is to permit deeper currents than a stationary bottom layer for a given initial ambient height H. This leads

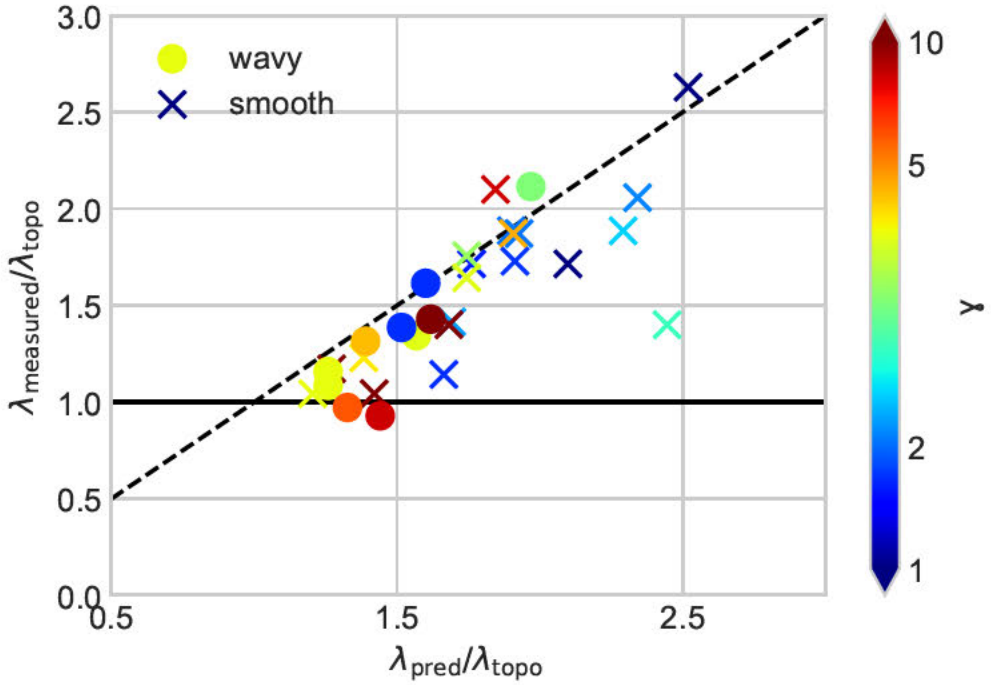


Figure 14: Predicted wavelength of instability λ_{pred} calculated from F and γ using the marginal stability quasi-geostrophic model (5), assuming a smooth wall versus the measured wavelength $\lambda_{\text{measured}}$. Points are normalised by the wavelength of the topography $\lambda_{\text{topo}} = 7\text{cm}$. Also plotted is the line $y = 1$ reresenting those points with $\lambda_{\text{measured}} = \lambda_{\text{topo}}$ (solid) and the line $\lambda_{\text{measured}} = \lambda_{\text{pred}}$ (dashed). Points are coloured according to the corresponding value of γ.

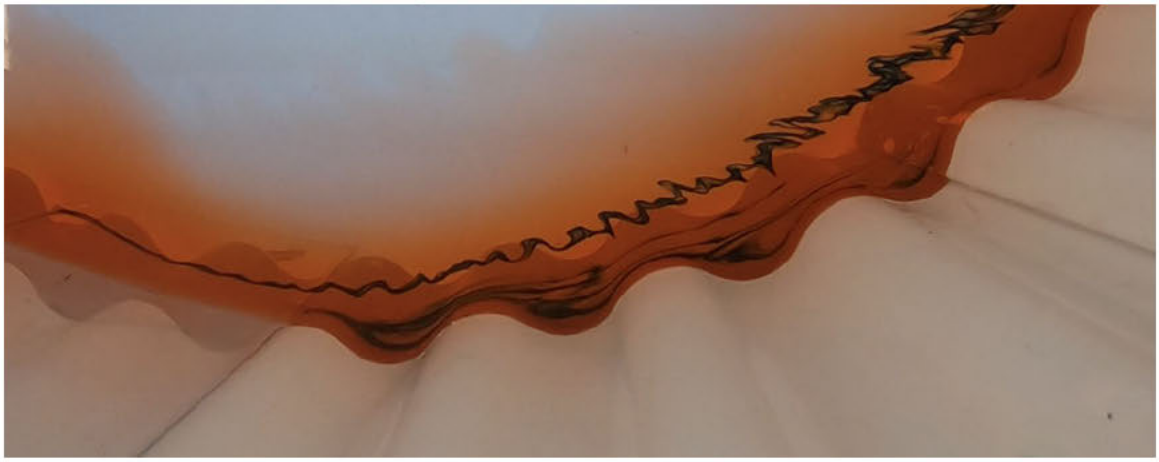


Figure 15: Photo taken from a GoPro camera mounted directly above the wavy wall near the source in an experiment with $f = 2.5\text{s}^{-1}$, $g = 4.7\text{cm}\,\text{s}^{-2}$, $Q = 10\text{cm}^3\text{s}^{-1}$, $H = 14.9\text{cm}$.

to larger values of γ and hence favours instability. In general, the trends in the height h_0 and width w_0 of the current predicted by the model were reproduced in the experiments, though we found that the depth was consistently smaller than the model predicted, whilst the width was consistently larger. It seems likely that the latter observation could be a result of non-zero potential vorticity in the top layer, as has been explored by [4].

Once they are established, geostrophic currents emerging from a point source are not expected to grow in width over time. Provided they are wide enough, they may become unstable to the marginal instability described in [7] once the shear between the two layers becomes sufficiently large. If the shear is already sufficiently large when the geostrophic current is established, supercritical instability as described in [9] is possible. These two types of instability behave independently on the Froude number $F = (w_0/L_R)^2$ describing the width of the current relative to its Rossby radius, and the depth ratio γ. We found in general that marginal instability curves gave a better prediction of the wavelength of instability observed than those of supercritical instability in the majority of our experiments. Our experiments fall in a different regime of (F, γ)-parameter space to classical ring source experiments. This being said, since the width of the current was consistently wider than predicted, F was large enough that measurable instability could be achieved at moderate values of γ, thus the mechanism for instability appears similar to the ring source experiments of [7]. Indeed, that study also performed a small number of point source experiments and made the same conclusion. We greatly extend the parameter space in γ however, and clear dependence of the instability on γ was observed.

Additionally, a couple of our experiments demonstrated convincing supercritical behaviour. Wider geostrophic currents naturally have a larger velocity at their boundaries, thus having a larger component of mean geostrophic shear potentially leading to supercritical instability. On the other hand, narrow currents may still become supercritically unstable by possessing an additional component of ageostrophic shear. In the lab, we hypothesise this may have been provided by the momentum of the impinging current near the source. In the ocean, tidal dynamics or variations in the source strength may have the same effect. As discussed in [7], various measured coastal current instabilities have wavelengths that more closely match the supercritical theory, such as the East Greenland Current [21], the Ligurian Sea coastal current [5] and the Norwegian Coastal Current [12]. It would be interesting to explore the areas of parameter space that gave rise to this behaviour in our experiments further in future work.

A limited set of experiments with a wavy lateral wall, that is, where the wavelength of instability was similar in size to the wavelength of the topography, demonstrated that there was no conclusive qualitative or quantitative difference between the flow instability over the smooth wall. This might be attributed to fluid being trapped in recirculating 'bay' regions created by the topography. It would be interesting to explore the parameter space with the wavy wall more fully, although we note that this would be difficult to achieve in the current lab set-up in which we had only one wavelength of topography available, meaning the ratio of wavelengths was controlled by the wavelength of instability alone. In particular, very small wavelengths of instability require very narrow and shallow currents, for which friction may be important, and possibly dominant, in the 1m tank. Similarly, we are unable to achieve wavelengths of instability of more than a few times the 7cm wavelength of the topography due to scale limitations.

6 Acknowledgements

First and foremost, thanks to Claudia Cenedese, my advisor, for proposing the project and for making my first experience working in the lab a great one - it was a pleasure working with you. Thanks to Anders Jensen for providing the all-important technical support and for the rapid construction of various devices to assist with the experimental set-up. I must also give a big thank you to Glenn Flierl for his many helpful suggestions, for his help with the theory, and most importantly for providing the Friday morning bakery selection without fail! Invaluable input was also offered by Jim McElwaine and Jack Whitehead. Thanks to Colm, Stefan, the principal lecturers Peter Schmid and Laure Zanna, and all of the GFD staff for putting on a fantastic summer. Finally, thanks to my fellow Fellows for some great times, and some slightly less great trivia night scores. And a special shout-out to Tilly for keeping me sane in the lab during the quieter weeks.

References

[1] E. Boss, N. Paldor, and L. Thompson, *Stability of a potential vorticity front: from quasi-geostrophy to shallow water*, J. Fluid Mech., 315 (1996), pp. 65–84.

[2] C. Cenedese and P. F. Linden, *Stability of a buoyancy-driven coastal current at the shelf break*, J. Fluid Mech., 452 (2002), pp. 97–121.

[3] G. Chabert d'Hières, H. Didelle, and D. Obaton, *A laboratory study of surface boundary currents: Application to the algerian current*, J. Geophys. Res. Oceans, 96 (1991), pp. 12539–12548.

[4] T. J. Crawford, *An experimental study of the spread of buoyant water into a rotating environment*, PhD thesis, University of Cambridge, 2017.

[5] M. Crépon, L. Wald, and J. Monget, *Low-frequency waves in the ligurian sea during december 1977*, Journal of Geophysical Research: Oceans, 87 (1982), pp. 595–600.

[6] R. W. Griffiths, *Gravity currents in rotating systems*, Annu. Rev. Fluid Mech., 18 (1986), pp. 59–89.

[7] R. W. Griffiths and P. F. Linden, *The stability of buoyancy-driven coastal currents*, Dyn. Atmos. Oceans, 5 (1981), pp. 281–306.

[8] ——, *The stability of vortices in a rotating, stratified fluid*, J. Fluid Mech., 105 (1981), pp. 283–316.

[9] ——, *Laboratory experiments on fronts: Part I: Density-driven boundary currents*, Geophys. & Astrophys. Fluid Dyn., 19 (1982), pp. 159–187.

[10] A. R. Horner-Devine, D. A. Fong, S. G. Monismith, and T. Maxworthy, *Laboratory experiments simulating a coastal river inflow*, J. Fluid Mech., 555 (2006), pp. 203–232.

[11] A. R. Horner-Devine, R. D. Hetland, and D. G. MacDonald, *Mixing and transport in coastal river plumes*, Annu. Rev. Fluid Mech., 47 (2015), pp. 569–594.

[12] M. Ikeda, J. A. Johannessen, K. Lygre, and S. Sandven, *A process study of mesoscale meanders and eddies in the Norwegian Coastal Current*, J. Phys. Oceanog., 19 (1989), pp. 20–35.

[13] N. Lahaye, A. Paci, and S. G. Llewellyn Smith, *Instability of lenticular vortices: Results from laboratory experiments, linear stability analysis and numerical simulations*, Fluids, 6 (2021), p. 380.

[14] S. J. Lentz, S. Elgar, and R. T. Guza, *Observations of the flow field near the nose of a buoyant coastal current*, J. Phys. Oceanog., 33 (2003), pp. 933–943.

[15] S. J. Lentz and K. R. Helfrich, *Buoyant gravity currents along a sloping bottom in a rotating fluid*, J. Fluid Mech., 464 (2002), pp. 251–278.

[16] A. F. Pearce and R. W. Griffiths, *The mesoscale structure of the Leeuwin Current: A comparison of laboratory models and satellite imagery*, J. Geophys. Res. Oceans, 96 (1991), pp. 16739–16757.

[17] J. Pedlosky, *Geophysical Fluid Dynamics*, vol. 710, Springer, 1987.

[18] N. A. Phillips, *Energy transformations and meridional circulations associated with simple baroclinic waves in a two-level, quasi-geostrophic model*, Tellus, 6 (1954), pp. 274–286.

[19] D. A. Sutherland and C. Cenedese, *Laboratory experiments on the interaction of a buoyant coastal current with a canyon: Application to the east greenland current*, J. Phys. Oceanog., 39 (2009), pp. 1258–1271.

[20] P. Thomas and P. Linden, *Rotating gravity currents: small-scale and large-scale laboratory experiments and a geostrophic model*, J. Fluid Mech., 578 (2007), pp. 35–65.

[21] P. Wadhams, A. E. Gill, and P. F. Linden, *Transects by submarine of the East Greenland polar front*, Deep-Sea Res., 26 (1979), pp. 1311–1327.

[22] C. L. Wolfe and C. Cenedese, *Laboratory experiments on eddy generation by a buoyant coastal current flowing over variable bathymetry*, J. Phys. Oceanog., 36 (2006), pp. 395–411.

[23] A. E. Yankovsky and D. C. Chapman, *A simple theory for the fate of buoyant coastal discharges*, J. Phys. Oceanog., 27 (1997), pp. 1386–1401.

Scaling with the Stars: The emergence of marginal stability in low Pr turbulence

Kasturi Shah

1 Introduction

Shear-driven turbulence in the stratified regions of planetary oceans, atmospheres, stellar interiors, and gas giants provides an important source of vertical transport of heat, momentum, and chemical tracers. Stratified turbulence in astrophysical objects differs fundamentally from geophysical turbulence because of the Prandtl number, Pr, which measures the ratio of momentum diffusivity to thermal diffusivity. Values of Pr in stably stratified geophysical systems such as Earth's atmosphere and oceans are typically 0.7 and 10, respectively. When $Pr = O(1)$, turbulent flows ($Re \gg 1$) are always close to adiabatic, i.e., thermally non-diffusive ($Pe \gg 1$). In stellar radiation zones of solar-type and intermediate-mass stars, $Pr = 10^{-9}$-10^{-5} [7]. The Peclet number, which is the ratio of the thermal diffusion timescale to the turbulent advection timescale, is also the product of the Reynolds and Prandtl number. With $Pr \ll 1$, $Pe \ll Re$, heat diffusion in stellar fluids is much more efficient than momentum diffusion at microscopic scales and the time scale for non-adiabatic effects may be potentially even shorter than the advective time scale ($Pe \ll 1$). This regime is, by contrast, not possible in geophysical turbulence where $Re \gg 1$ implies $Pe \gg 1$.

Stratified turbulence in stars is thought to be generated by horizontal shear instabilities [12]. In a horizontal shear flow, due to the high stratification and low viscosity, the turbulent eddies are flat and only weakly coupled in the vertical direction. Their characteristic vertical scale of velocity variation, H, is far smaller than the characteristic horizontal scale, L, such that the aspect ratio $\alpha = H/L \ll 1$. Their relative motion via horizontal rotation produces vertical shear on the vertical lengthscale, H, which generates vertical motion and vertical mixing. As turbulence in stars is difficult to observe, numerical simulations of strongly stratified, $Pr \ll 1$ flows yield considerable insight into the validity of the [12] mechanism. Numerical simulations at low [4] and high [6] Péclet number exhibit vertical velocity layering, supporting [12]'s horizontal shear instability mechanism for stratified stellar turbulence. The flows are strongly anisotropic and exhibit scale separation, as predicted.

Scaling relationships between the aspect ratio and modified Froude number Fr_M, the ratio of the linear wave period to the time scale of the large-scale flow [8, 11], characterise the interplay between the anisotropy and the stratification. Various scaling relationships have been proposed. At low Péclet number, the vertical velocity is the unique forcing for the buoyancy, such that $w = \nabla^2 b$ [8]. Two proposed scaling relationships are: $\alpha \sim Fr_M$ [11] and $\alpha \sim Fr_M^{4/3}$ [4]. [4] explain the scalings that emerged from DNS by balancing

the vertical advection of vertical velocity and the thermally-constrained buoyancy term ($w\partial_z w \sim (N^2/\kappa_T)\nabla^{-2}w$), while [11] recovers the $\alpha \sim Fr_M$ scaling by balancing the vertical gradient of pressure and the thermally-constrained buoyancy term, $\partial_z p \sim (N^2/\kappa_T)\nabla^{-2}w$. At large Péclet number (but low Prandtl number), [6] finds that the vertical lengthscale varies as $Fr^{2/3}$ in numerical simulations. As changes in scaling relationships predict transitions between turbulent behaviours, a self-consistent and rigorous derivation of physically reliable scalings is necessary to identify various turbulent regimes.

Existing algorithms for time integrations of slow-fast quasilinear systems provide methods to solve the derived multiscale model presented here [5, 10]. The central challenge is the integration of the reduced model on two timescales and two spatial scales. The former is typically approached by solving an eigenvalue problem for the fast-varying fields and time-stepping the slowly-varying fields on the slow timescale. The latter is addressed by considering small spatial scales only (for simplicity) and hence suppressing the large-scale derivatives. The key insight obtained from applying these algorithms is the evolution of the growth rate, represented by the eigenvalue, which indicates the stability of the flow. Additionally, the algorithm explicitly calculates the amplitude of the fast-varying fluctuation fields. Their feedback on the mean flow maintains its marginal stability.

Motivated by open questions regarding the validity of scaling relationships and identification of distinct turbulent regimes, we present a formal, multiscale analysis of governing low-Pr (Boussinesq) equations at low and high Pe_b in the limit of strong stratification. Scaling relationships between the aspect ratio and modified Froude number emerge naturally from the multiscale analysis, which is supported by prior numerical simulations revealing anisotropic, scale-separated, dynamics. We use our analysis to assess the validity of published scaling relationships and construct a full regime diagram.

2 Multiscale Model Development

Consider a three-dimensional, non-rotating, incompressible, stably stratified flow expressed in a Cartesian reference frame where z is aligned with gravity $\mathbf{g} = -g\mathbf{e}_z$. Let $\boldsymbol{u}_\perp$ denote the horizontal velocity, w the vertical velocity, p the pressure divided by a constant reference density, and b the buoyancy perturbation with respect to a linearly stratified background. The fluid has, in accordance with the Boussinesq approximation, a constant kinematic viscosity ν, thermal diffusivity κ_T, coefficient of thermal expansion β, and a constant stratification measured by the buoyancy frequency N. The governing equations for this configuration are,

$$\frac{\partial \boldsymbol{u}_\perp}{\partial t} + (\boldsymbol{u}_\perp \cdot \nabla_\perp)\,\boldsymbol{u}_\perp + w\frac{\partial \boldsymbol{u}_\perp}{\partial z} = -\nabla_\perp p + \nu\left(\nabla_\perp^2 \boldsymbol{u}_\perp + \frac{\partial^2 \boldsymbol{u}_\perp}{\partial z^2}\right) + f(z)\hat{\boldsymbol{e}}_x, \tag{1a}$$

$$\frac{\partial w}{\partial t} + (\boldsymbol{u}_\perp \cdot \nabla_\perp)\,w + w\frac{\partial w}{\partial z} = -\frac{\partial p}{\partial z} + b + \nu\left(\nabla_\perp^2 w + \frac{\partial^2 w}{\partial z^2}\right), \tag{1b}$$

$$\nabla_\perp \cdot \boldsymbol{u}_\perp + \frac{\partial w}{\partial z} = 0, \tag{1c}$$

$$\frac{\partial b}{\partial t} + (\boldsymbol{u}_\perp \cdot \nabla_\perp)\,b + w\frac{\partial b}{\partial z} + N^2 w = \kappa_T\left(\nabla_\perp^2 b + \frac{\partial^2 b}{\partial z^2}\right), \tag{1d}$$

where the horizontal gradient operator is denoted by $\nabla_\perp$ and $\perp$ represents the horizontal coordinates x and y. A body force fe_x (where f is a function of z only) is applied to drive a mean horizontally sheared flow.

2.1 Low Péclet number equations

Motivated by evidence of strongly anisotropic flows in numerical simulations of thermally diffusive stellar fluids [4, 6], we non-dimensionalise the system in (1) anisotropically by defining dimensionless (hatted) variables

$$(x,y) = L(\hat{x},\hat{y}), \quad z = \alpha L\hat{z}, \quad \boldsymbol{u}_\perp = U\hat{\boldsymbol{u}}_\perp, \quad t = \frac{L}{U}\hat{t}, \quad w = \alpha U\hat{w}, \quad p = U^2\hat{p}, \quad b = \alpha^3\frac{N^2UL^2}{\kappa_T}\hat{b}, \tag{2}$$

where U is the characteristic horizontal velocity scale. Note that the body force is of order U^2/L. At low Péclet number, the vertical velocity is the unique forcing for the buoyancy, and we expect $N^2w = \kappa_T\nabla^2 b$ [8]. Accordingly, we chose the dimensionless scaling of b in (2) to obtain this balance in the limit $Pe \to 0$. On substituting the newly defined variables in (2) and omitting the hats, (1) becomes

$$\frac{\partial \boldsymbol{u}_\perp}{\partial t} + (\boldsymbol{u}_\perp\cdot\nabla_\perp)\,\boldsymbol{u}_\perp + w\frac{\partial \boldsymbol{u}_\perp}{\partial z} = -\nabla_\perp p + \frac{1}{Re\alpha^2}\left(\alpha^2\nabla_\perp^2\boldsymbol{u}_\perp + \frac{\partial^2\boldsymbol{u}_\perp}{\partial z^2}\right) + f, \tag{3a}$$

$$\frac{\partial w}{\partial t} + (\boldsymbol{u}_\perp\cdot\nabla_\perp)\,w + w\frac{\partial w}{\partial z} = -\frac{1}{\alpha^2}\frac{\partial p}{\partial z} + \frac{\alpha^2}{Fr_M^4}b + \frac{1}{Re\alpha^2}\left(\alpha^2\nabla_\perp^2 w + \frac{\partial^2 w}{\partial z^2}\right), \tag{3b}$$

$$\nabla_\perp\cdot\boldsymbol{u}_\perp + \frac{\partial w}{\partial z} = 0, \tag{3c}$$

$$\frac{\partial b}{\partial t} + (\boldsymbol{u}_\perp\cdot\nabla_\perp)\,b + w\frac{\partial b}{\partial z} + \frac{1}{Pe\alpha^2}w = \frac{1}{Pe\alpha^2}\left(\alpha^2\nabla_\perp^2 b + \frac{\partial^2 b}{\partial z^2}\right), \tag{3d}$$

where the forcing has been non-dimensionalised by U^2/L. The following dimensionless parameters arise:

$$Re = \frac{UL}{\nu}, \quad \alpha = \frac{H}{L}, \quad Fr_M = \left(\frac{U\kappa_T}{N^2L^3}\right)^{1/4}, \quad Pr = \frac{\nu}{\kappa_T}, \quad Pe = PrRe, \tag{3e}$$

representing the Reynolds number, the aspect ratio, the modified Froude number, the Prandtl number and the Péclet number. Both the aspect ratio and the modified Froude number are emergent parameters. Crucially, to describe low Pe anisotropic flows, ($3d$) requires a small *buoyancy* Péclet number, $Pe_b = Pe\,\alpha^2 \ll 1$, not a low bare Péclet number. In this limit, ($3d$) reduces to

$$w = \alpha^2\left(\nabla_\perp^2 + \frac{1}{\alpha^2}\frac{\partial^2}{\partial z^2}\right)b. \tag{3f}$$

At low Pe_b, the vertical advection of the background buoyancy gradient is balanced by the diffusion of the individual anomaly rather than by the time tendency of the buoyancy anomaly, as at high Pe_b. Alternatively, (3f) can be derived by expanding b in the Boussinesq equations in powers of Pe and assuming an order unity velocity field [8]. At $O(Pe^0)$, the subjugation of the buoyancy to the vertical velocity emerges, i.e., $w = \nabla^2 b$, consistent with (3f). ($3abcf$) are referred to as the low Péclet number equations (LPN).

2.2 Multiple scale asymptotics

Numerical studies of strongly stratified stellar turbulence at low Pe [4] and high Pe [6] exhibit anisotropy and thus scale separation. The dominant shear instabilities have horizontal scales commensurate with the vertical scale of variability of the large-scale flow. The vertical scale of velocity variation H is much smaller than the horizontal scale. Motivated by these findings, we develop a multiscale model for low Pe_b in §2.2 and, separately, for high Pe_b flows in §2.3. Based on the anisotropy described by the aspect ratio, we formally split the horizontal spatial scales into 'slow' and 'fast' scales, such that $\boldsymbol{x}_{\perp f} = \boldsymbol{x}_{\perp s}/\alpha$ and $\boldsymbol{x}_{\perp s} = \boldsymbol{x}_{\perp}$, where subscript f denotes fast and subscript s denotes slow [3]. Based on the time scale for horizontal shear in layers separated by distance αL, we formally split the temporal scales into a 'slow' and 'fast' time scale, such that $t_f = t_s/\alpha$ and $t_s = t$. Consequently, the partial derivatives transform as

$$\frac{\partial}{\partial t} = \frac{1}{\alpha}\frac{\partial}{\partial t_f} + \frac{\partial}{\partial t_s}, \qquad \nabla_\perp = \frac{1}{\alpha}\nabla_{\perp f} + \nabla_{\perp s}. \tag{4}$$

In accordance with the multiple scale asymptotic formalism, the buoyancy, pressure and velocity fields are functions of both $\boldsymbol{x}_{\perp f}$ and $\boldsymbol{x}_{\perp s}$ and of both t_f and t_s. For a multiscale function $q(\boldsymbol{x}_{\perp f}, \boldsymbol{x}_{\perp s}, z, t_f, t_s; \alpha)$, we define a fast-averaging operator $\overline{(\cdot)}$,

$$\overline{q}(\boldsymbol{x}_{\perp s}, z, t_s; \alpha) = \lim_{T_f, L_x, L_y \to \infty} \frac{1}{L_x L_y T_f} \int_0^{t_f} \int_D q(\boldsymbol{x}_{\perp f}, \boldsymbol{x}_{\perp s}, z, t_f, t_s; \alpha) \mathrm{d}\boldsymbol{x}_{\perp f} \mathrm{d}t_f. \tag{5}$$

where D is the horizontal $\boldsymbol{x}_{\perp f}$ domain, with fast spatial periods L_x and L_y, and T_f is the fast time-integration period. Hence, $\overline{q}$ depends on slow variables only. Hence, q can be split into a slowly-varying and a fast fluctuation component, $q - q' = \overline{q}$. Here, primes denote fluctuation fields, where the fast-average of the fluctuation field vanishes, i.e., $\overline{q'} = 0$.

2.3 Multiple scale quasilinear model for low Péclet flows

We begin with the development of the multiscale model for low Pe_b flows. We proceed to asymptotically expand the pressure, horizontal velocity, vertical velocity, and buoyancy. The expansion proceeds in fractional powers of α where the exponent, γ, in the expansion is determined separately in the high Pe_b and the low Pe_b cases. We posit the following asymptotic expansions,

$$[p, \boldsymbol{u}_\perp] \sim [p_0, \boldsymbol{u}_{\perp 0}] + \alpha^\gamma [p_1, \boldsymbol{u}_{\perp 1}] + \alpha^{2\gamma} [p_2, \boldsymbol{u}_{\perp 2}] + \ldots, \tag{6a}$$

$$[b, w] \sim \frac{1}{\alpha^\gamma}[b_{-1}, w_{-1}] + [b_0, w_0] + \alpha^\gamma [b_1, w_1] + \ldots, \tag{6b}$$

which reflect our expectation that the dominant contributions to the pressure and velocity arise on large horizontal scales; accordingly, their expansions begin at $O(1)$. In contrast, in stratified turbulence, the vertical velocity is a small-scale field [2, 9, 4]. For the vertical divergence of the vertical flux of horizontal momentum associated with fluctuations to feed back on the leading-order large-scale horizontal flow, the fluctuation velocities must be appropriately small, given the 3D incompressibility of the isotropic fluctuating flow. Specifically, the vertical divergence of the fluctuation flux is $\partial_z(\overline{w'u'}) = O[(U'/U)(W'/U)(1/\alpha)]$

relative to the inertial terms, where fluctuations scales are denoted as primed capital letters. Since $W' = U'$ only from continuity, $(W')^2 = \alpha U^2$, i.e., $W' = \alpha^{1/2} U$, so $\gamma = 1/2$ and the vertical velocity expansion starts at w_{-1}. The tight coupling between vertical velocity and buoyancy in the LPN equation (3f), requires the asymptotic expansion of b to mimic w.

On substituting the two-scale derivatives (4) and asymptotic expansions (6) into the LPN equations, (3abc) and (3f), we obtain at lowest order

$$\frac{\partial \boldsymbol{u}_{\perp 0}}{\partial t_f} + \boldsymbol{u}_{\perp 0} \cdot \nabla_{\perp f} \boldsymbol{u}_{\perp 0} = -\nabla_{\perp f} p_0, \qquad \frac{\partial p_0}{\partial z} = 0, \qquad \nabla_{\perp f} \cdot \boldsymbol{u}_{\perp 0} = 0. \tag{7a,b,c}$$

Following arguments in [3], we find that $\boldsymbol{u}_{0\perp} = \overline{\boldsymbol{u}}_{0\perp}$ only. Then (7a) requires that $\nabla_{\perp f} p_0 = 0$. This combined with fast averaging (7b), from which we obtain $\partial_z \overline{p}_0 = 0$, implies that the leading-order pressure too is independent of fast horizontal and temporal scales, i.e., $p_0 = \overline{p}_0$. At next order, the governing equations are,

$$\frac{\alpha^\gamma}{\alpha} \nabla_{\perp f} \cdot \boldsymbol{u}_1' + \frac{1}{\alpha^\gamma} \frac{\partial w_{-1}}{\partial z} = 0, \tag{8a}$$

$$\frac{\alpha^\gamma}{\alpha} \left(\frac{\partial \boldsymbol{u}_{\perp 1}'}{\partial t_f} + \overline{\boldsymbol{u}}_{\perp 0} \cdot \nabla_{\perp f} \boldsymbol{u}_{\perp 1}' \right) + \frac{1}{\alpha^\gamma} w_{-1} \frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial z} = -\frac{\alpha^\gamma}{\alpha} \nabla_{\perp f} p_1, \tag{8b}$$

$$\frac{1}{\alpha^{\gamma+1}} \frac{\partial w_{-1}}{\partial t_f} + \frac{1}{\alpha^{\gamma+1}} \overline{\boldsymbol{u}}_{\perp 0} \cdot \nabla_{\perp f} w_{-1} = -\frac{\alpha^\gamma}{\alpha^2} \frac{\partial p_1}{\partial z} + \frac{\alpha^2}{Fr_M^4} \frac{1}{\alpha^\gamma} b_{-1}, \tag{8c}$$

$$w_{-1} = \left(\frac{\partial^2}{\partial z^2} + \nabla_{\perp f}^2 \right) b_{-1}. \tag{8d}$$

The balance of terms in (8ab) implies that $\alpha^\gamma/\alpha \sim 1/\alpha^\gamma$ must be true, such that the asymptotic parameter in (6) is $\alpha^{1/2}$. Hence, (8ab) provide a mathematical basis for our expectation that $\gamma = 1/2$, which arose from physical arguments about the order of magnitude of the vertical divergence of the vertical flux relative to the inertial terms. (8cd) then implies a balance $\alpha^{-3/2} \sim \alpha^{3/2}/Fr_M^4$, yielding the crucial scaling relationship $\alpha \sim Fr_M^{4/3}$. Fast averaging (8) then gives

$$\frac{\partial \overline{w}_{-1}}{\partial z} = 0, \qquad \overline{w}_{-1} \frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial z} = 0, \qquad \frac{\partial \overline{p}_1}{\partial z} = \left(\frac{\partial^2}{\partial z^2} \right)^{-1} \overline{w}_{-1}. \tag{9a,b,c}$$

From (9a) we conclude that $\overline{w}_{-1} = 0$, provided $\overline{u}_{-1} = 0$ along any given z plane. As expected for strongly stratified flow, the leading order vertical velocity is larger on small than on large horizontal scales, i.e., $w_{-1} = w_{-1}'$. Hence, (9b) is trivially satisfied and (9c) yields $\partial_z \overline{p}_1 = 0$. Given the tight coupling between the vertical velocity and buoyancy in (3f), (9a) implies that $b_{-1} = b_{-1}'$, only. We obtain the governing equations for the fluctuations by subtracting (9) from (8).

To derive the mean flow equations, we collect terms at $O(1)$ in our asymptotically

expanded equations:

$$\frac{\partial \boldsymbol{u}_{\perp 2}}{\partial t_f} + (\overline{\boldsymbol{u}}_{\perp 0}\cdot\nabla_{\perp f})\boldsymbol{u}_{\perp 2} + w_0\frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial z} + \nabla_{\perp f} p_2 =$$
$$-\frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial t_s} - (\overline{\boldsymbol{u}}_{\perp 0}\cdot\nabla_{\perp s})\overline{\boldsymbol{u}}_{\perp 0} - \nabla_{\perp s}\overline{p}_0 + \frac{1}{Re_b}\frac{\partial^2 \overline{\boldsymbol{u}}_{\perp 0}}{\partial z^2} - (\boldsymbol{u}_{\perp 1}\cdot\nabla_{\perp f})\boldsymbol{u}_{\perp 1} - w'_{-1}\frac{\partial \boldsymbol{u}'_{\perp 1}}{\partial z} + f, \tag{10a}$$

$$\frac{\partial w_0}{\partial t_f} + (\overline{\boldsymbol{u}}_{\perp 0}\cdot\nabla_{\perp f})w_0 + \frac{\partial p_2}{\partial z} - \left(\nabla^2_{\perp f} + \frac{\partial^2}{\partial z^2}\right)^{-1} w_0 = -(\boldsymbol{u}_{\perp 1}\cdot\nabla_{\perp f})w'_{-1} - w'_{-1}\frac{\partial w'_{-1}}{\partial z}, \tag{10b}$$

$$\nabla_{\perp f}\cdot\boldsymbol{u}_{\perp 2} + \nabla_{\perp s}\cdot\overline{\boldsymbol{u}}_{\perp 0} + \frac{\partial w_0}{\partial z} = 0, \tag{10c}$$

where the *buoyancy* Reynolds number is $Re_b = Re\,\alpha^2$. We have chosen to interpret the forcing as an $O(1)$ quantity. A necessary condition for bounded behaviour of the $O(\alpha)$ fluctuation fields is that the fast average of the right-hand side of $(10a)$ vanishes. On fast averaging (10) and making use of the continuity equation $(8a)$ at $O(1/\alpha^{1/2})$, we obtain equations for the leading order mean fields, $\overline{\boldsymbol{u}}_{\perp 0}$, $\overline{w}_0$, and $\overline{b}_0$.

Gathering the results of the formal multiscale asymptotic analysis, we obtain a novel two-scale model for strongly stratified, turbulent flows at low Pe_b, as summarised below.
Mean flow equations

$$\frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial t_s} + (\overline{\boldsymbol{u}}_{\perp 0}\cdot\nabla_{\perp s})\overline{\boldsymbol{u}}_{\perp 0} + \overline{w}_0\frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial z} = -\nabla_{\perp s}\overline{p}_0 - \frac{\partial}{\partial z}\left(\overline{w'_{-1}u'_1}\right) + \frac{1}{Re_b}\frac{\partial^2 \overline{\boldsymbol{u}}_{\perp 0}}{\partial z^2} + \overline{f}_0 \tag{11a}$$

$$\frac{\partial \overline{p}_0}{\partial z} = 0 \tag{11b}$$

$$\nabla_{\perp s}\cdot\overline{\boldsymbol{u}}_{\perp 0} + \frac{\partial \overline{w}_0}{\partial z} = 0 \tag{11c}$$

Fluctuation equations

$$\frac{\partial \boldsymbol{u}'_{\perp 1}}{\partial t_f} + (\overline{\boldsymbol{u}}_{\perp 0}\cdot\nabla_{\perp f})\boldsymbol{u}'_{\perp 1} + w'_{-1}\frac{\partial \overline{\boldsymbol{u}}_{\perp 0}}{\partial z} = -\nabla_{\perp f}p'_1 + \frac{\alpha}{Re_b}\left(\nabla^2_{\perp f} + \frac{\partial^2}{\partial z^2}\right)\boldsymbol{u}'_{\perp 1} \tag{11d}$$

$$\frac{\partial w'_{-1}}{\partial t_f} + (\overline{\boldsymbol{u}}_{\perp 0}\cdot\nabla_{\perp f})w'_{-1} = -\frac{\partial p'_1}{\partial z} + \left(\nabla^2_{\perp f} + \frac{\partial^2}{\partial z^2}\right)^{-1} w'_{-1} + \frac{\alpha}{Re_b}\left(\nabla^2_{\perp f} + \frac{\partial^2}{\partial z^2}\right)w'_{-1} \tag{11e}$$

$$\nabla_{\perp f}\cdot\boldsymbol{u}'_{\perp 1} + \frac{\partial w'_{-1}}{\partial z} = 0 \tag{11f}$$

Note that in $(11de)$, formally small higher-order Laplacian diffusion terms have been added to regularize the fluctuation dynamics in the possible presence of sharp vertical gradients or critical layers, as in [3]. We note that, for the dimensionless system (3), the vertical lengthscale $\alpha L = Fr_M^{4/3}L$ is, in the limit of strong stratification, so small that mean buoyancy *anomalies*, i.e., departures from the imposed linear basic state profile, do not disrupt the leading-order basic state hydrostatic balance: $\partial_z\overline{p}_0 = 0$. As the scaling relationship $\alpha =$

$Fr_M^{4/3}$ describes the short vertical scale between horizontal eddies (c.f. [12]), $\partial_z \overline{p}_0 = 0$ only on these short scales. Higher order mean pressure terms, however, do depend on mean buoyancy via gradients in mean vertical velocity, for instance, $\partial_z \overline{p}_2 = ((\nabla_{\perp s})^2 + \partial_z^2)^{-1} \overline{w}_2 - \partial_z \overline{w'_{-1} w'_{-1}}$. This higher-order buoyancy dependence offers a possible path for (weak) buoyancy effects to be incorporated into the mean dynamics of the reduced order model (11), which may be important on *larger vertical* scales. We do not pursue this path in the present study, but instead briefly outline it here for future work. In the horizontal momentum equation, $\overline{p}_0$ can be replaced by a composite pressure, $\overline{p}_c = \overline{p}_0 + \alpha \overline{p}_2$. For consistency, the corresponding horizontal momentum equation accurate to $O(\alpha)$ should be derived in terms of a composite horizontal velocity, $\overline{\boldsymbol{u}}_{\perp c}$. Finally, we emphasize that buoyancy anomalies *do* affect the fluctuation dynamics.

The closed system (11) tightly couples the mean flow to the fluctuations. The mean flow modifies the fluctuation dynamics via advection by $\overline{\boldsymbol{u}}_{\perp 0}$ and by modifications of the vertical shear. The fluctuation equations are linear in the fluctuations themselves. However, the fluctuations feed back non-linearly on the mean flow via the divergence of the Reynolds stress term in the horizontal momentum equation (11a). Therefore, the multiscale system (11) has a (generalised) quasilinear form. A central result of this study is the emergence of this quasilinearity as a consequence of the strong stratification and the associated formal asymptotic derivation: nowhere in the multiscale expansion do we invoke nor impose quasilinearity as an *adhoc* closure for the mean dynamics.

2.4 Summary of multiple scale quasilinear model for high buoyancy Péclet flows

Next, we develop a multiscale model for high Pe_b flows, focusing only on those points of difference with the multiscale model for low Pe_b. At high Pe_b, we non-dimensionalise (1) using the same scalings as in (2), but replace the buoyancy scaling with $b = \alpha N^2 L \hat{b}$. At large Pe_b, the vertical advection of the background buoyancy gradient is balanced by the time tendency of the buoyancy anomaly rather than by the diffusion of the individual anomaly, as at low Pe_b. We perform a multiscale analysis of the resulting dimensionless governing equations by substituting the two-scale derivatives (4). The asymptotic expansions used are as in (6), except for buoyancy. b is no longer forced by w and hence b is a large-scale field; accordingly the asymptotic expansion for b begins at $O(1)$, i.e., $b = b_0 + \alpha^\gamma b_1 + \alpha^{2\gamma} b_2$. Consequently, the derivation follows [3], yielding the scaling relationship $\alpha = B^{-1/2} \equiv Fr$, a well-known scaling result for strongly stratified geophysical turbulence [1, 2, 3]. Here, the Froude number, $Fr = U/NL$, is the ratio of the buoyancy period to the time scale of the large-scale flow. The resulting closed, generalised quasilinear two-scale system for strongly stratified, turbulent high Pe_b flows is identical to the system presented in [3]. We note that in the mean flow and fluctuation buoyancy equations,

$$\frac{\partial \overline{b}_0}{\partial t_s} + (\overline{\boldsymbol{u}}_0 \cdot \nabla_s)\overline{b}_0 + \overline{w}_0 \frac{\partial \overline{b}_0}{\partial z} = -\frac{1}{Pe_b}\overline{w}_0 - \frac{\partial}{\partial z}\left(\overline{w'_{-1} b'_1}\right) + \frac{1}{Pe_b}\frac{\partial^2}{\partial z^2}\overline{b}_0, \tag{12a}$$

$$\frac{\partial b'_1}{\partial t_f} + (\overline{\boldsymbol{u}}_0 \cdot \nabla_f)\, b'_1 + w'_{-1}\frac{\partial \overline{b}_0}{\partial z} = -\frac{1}{Pe_b} w'_{-1} + \frac{\alpha}{Pe_b}\left(\nabla_{\perp f}^2 + \frac{\partial^2}{\partial z^2}\right) b'_1, \tag{12b}$$

an inverse buoyancy Péclet number $1/Pe_b$ multiplies the vertical velocity (i.e., the first term on the RHS of equations (2.30) and (2.34) in [3]). However, this $1/Pe_b$ factor is an artefact of the choice of non-dimensionalisation.

3 Regimes Of Stratified Stellar Turbulence

One merit of the multiscale analysis detailed in §2 is that scaling relationships for the aspect ratio naturally emerge. These relationships for low Pe_b and high Pe_b flows are,

$$\alpha \sim Fr_M^{4/3}, \qquad \alpha \sim Fr, \tag{13a,b}$$

respectively. We now assess the validity of published scaling relationships. Numerical simulations in [6] imply scalings for: the vertical eddy scale $\hat{l}_z \sim Fr^{2/3}$, the root-mean square (rms) vertical velocity $\hat{w}_{\rm rms} \sim Fr^{2/3}$, and the rms temperature fluctuations $\hat{T}_{\rm rms} \sim Fr^{4/3}$, where we have translated their notation using $B = Fr^{-2}$. The hatted variables are dimensionless. Given the $\alpha \sim Fr$ scaling in (13b), there would seem to be no theoretical basis for the scalings in [6].

In the low Pe case, the $\alpha \sim Fr_M^{4/3}$ scaling is verified by the numerical simulations in [4]. Indeed, a key contribution of this study is that it provides a theoretical basis for the scaling underlying the numerical results in [4]. To date, two distinct low Pe scalings have been proposed in the literature, $\alpha \sim Fr_M^{4/3}$ (this study validated by [4]) and $\alpha \sim Fr_M$ ([11] or from hydrostatic balance in the anisotropically scaled equation (3b)). A key difference between the $\alpha \sim Fr_M^{4/3}$ and $\alpha \sim Fr_M$ relationships is the vertical scales they describe; for instance, our low Pe_b multiscale model (11) with its intrinsic $\alpha \sim Fr_M^{4/3}$ scaling describes the short vertical scales between the decoupled horizontal eddies that generate vertical shear [12]. These differences raise a natural question: which of these regimes (if either) characterises turbulence in stars?

To address this question, in Figure 1 we construct a regime diagram for stellar turbulence. Our multiscale models for low and high Pe_b flows describe stratified, anisotropic flows. The unstratified regime where our multiscale models do not apply is identified by regions where $Fr > 1$, i.e., when $B < 1$. For $Pe_b < 1$ (where the LPN approximation is valid), the demarcating line $BPe \sim 1$ (along which $\alpha \sim 1$) represents isotropic flows. We first identify the viscous and adiabatic bounds between which the low Pe_b multiscale model (11) is valid. To identify the viscous transition of the fluctuation fields, we consider the multiscale fluctuation equation in the low Pe case (11de), in which, relative to the mean dynamics, viscous diffusion of fluctuation momentum is weak by the factor α. Therefore, the viscous transition for the fluctuation fields occurs at $Re_b = \alpha$ and the viscous fluctuation regime arise for $\alpha/Re_b \ll 1$. For the adiabatic transition, the relevant parameter is Pe_b rather than Pe; accordingly, we consider the multiscale buoyancy fluctuation equation for low Pe_b but high Pe in (3d). The balance $\partial_t b'_{-1} + \cdots = (\alpha/Pe_b)\nabla^2 b_{-1}$ indicates that this transition occurs at $Pe_b = \alpha$. Adiabatic dynamics occur when $Pe_b/\alpha \gg 1$. The range of validity of the LPN multiscale model is therefore $1/Re \ll \alpha \ll 1/Pe$. For ease of comparison with previously published regime diagrams [4, 6], we express the resulting inequalities in terms of Pe and B using $Fr_M = (BPe)^{-1/4}$. On substituting the scaling relationship

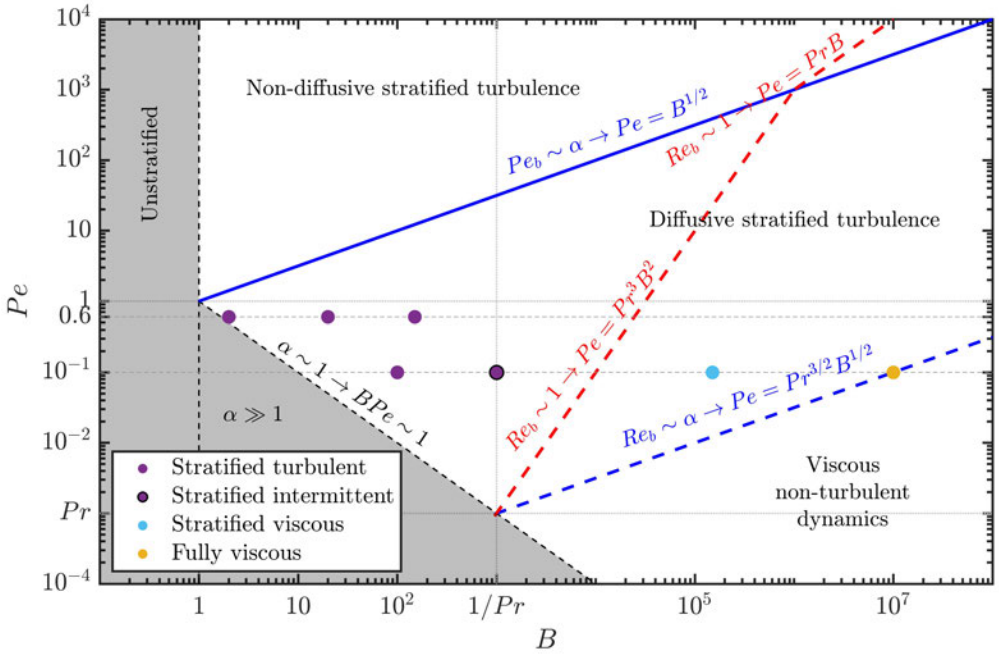


Figure 1: Regime diagram showcasing stellar turbulent behaviours for $Pr = 10^{-3}$. The blue solid and dashed lines mark the adiabatic and viscous transitions in (14). The low Pe multiscale equations (11) are valid between these blue bounds. The parameter space above the blue solid line corresponds to adiabatic stratified turbulence where the high Pe multiscale model applies. The red dashed line marks the viscous mean flow transition in (15). The coloured circles represent numerical simulations whose behaviour is classified following [4].

(13a), the region of LPN validity where the low Pe multiscale model (11) applies is

$$Pr^{3/2}B^{1/2} \ll Pe \ll B^{1/2}, \tag{14}$$

demarcating where diffusive stratified turbulence occurs. Notably, the adiabatic bound $Pe \ll B^{1/2}$ can be equivalently derived using the high Pe scaling given in (13b).

Next, we identify the boundary along which the mean flow becomes viscous. This occurs when $Re_b = O(1)$ which, on substitution of (13ab), yields

$$Pe \sim Pr^3B^2, \qquad Pe \sim PrB, \tag{15}$$

respectively. In Figure 1, the region above the red dashed line corresponds to non-viscous dynamics, below the blue dashed line to fully viscous non-turbulent dynamics, and between the red and blue dashed lines to viscous mean flow but non-viscous fluctuations.

To corroborate the theoretical predictions, two sets of simulations at $Pr = 10^{-3}$ are overlaid in coloured circles in Figure 1: $Re = 100$ and $Pe = 0.1$, which solve (3$abcd$), and $Re = 600$ and $Pe = 0.6$, which solve the LPN equations (3$abcf$). These simulations are categorised into stratified turbulent, stratified intermittent, stratified viscous, and fully viscous behaviours, classified consistently with [4]. The stratified turbulent and intermittent simulations (purple/purple with black outline) behave independently of viscosity and lie in the non-viscous region of the diffusive stratified turbulent regime. The stratified viscous simulation (cyan) is in the region of the diffusive stratified regime with a viscous mean flow and non-viscous fluctuations, while the fully viscous simulation (yellow) lies in the viscous non-turbulent regime. Hence, there is compelling agreement between the theoretical predictions and the independently classified numerical results

Returning to the question of which scaling relationship, $\alpha \sim Fr_M^{4/3}$ or $\alpha \sim Fr_M$, may be expected to be realised in stars, we consider where the latter can occur in Figure 1. As the lines indicating the viscous mean flow and fluctuation transitions, $Re_b \sim 1$ and $Re_b \sim \alpha$ respectively, intersect exactly at the isotropic transition $\alpha \sim 1$, there is evidently no region in the regime diagram where the $\alpha \sim Fr_M$ scaling in [11] applies. We note that his scaling is only dynamically consistent with the strongly stratified $\alpha \sim Fr$ relationship when there are no small scales and when $Fr_M = Fr$ interchangeably.

4 Marginal Stability Of Vertical Shear Instabilities In Low Pe Flows

4.1 Time integration of the slow-fast quasilinear system

Having established the regimes of strongly stratified stellar turbulence, we now present solutions of the slow-fast quasilinear system (11). We consider three systems: the anisotropically scaled dimensionless governing equations in (3), henceforth called direct numerical simulations (DNS), the quasilinear system integrated on one single timescale, henceforth called a single timescale quasilinear system (STQL), and the multiple scale quasilinear system (MTQL). Our focus is vertical shear instabilities. To study them, we take a vertical slice through our cuboid of low Pr fluid and henceforth consider the equations in x and z only. We assume a forcing of the form $f = 10\cos(z)/Re_b\,\hat{e}_x$.

4.1.1 Single timescale formulation (STQL)

The single timescale formulation evolves the entire system on a single timescale, which we choose to be the fast timescale. To express (11) on a single time scale, we undo the chain rule, i.e. $\partial_{t_s} = \partial_{t_f}/Fr_M^{4/3}$. For convenience, we suppress the derivatives with respect to x_s and zoom in to the small horizontal scales only. On assuming that the fast average is only horizontal, (11) becomes

$$\frac{1}{Fr_M^{4/3}}\frac{\partial \overline{u}_0}{\partial t_f} - \frac{1}{Re_b}\frac{\partial^2 \overline{u}_0}{\partial z^2} = -\frac{\partial}{\partial z}\left(\int w_1' u_1' \mathrm{d}x\right) + \overline{f}_0, \tag{16a}$$

$$\frac{\partial u_1'}{\partial t_f} + \frac{\partial p_1'}{\partial x_f} - \frac{Fr_M^{4/3}}{Re_b}\left(\frac{\partial^2}{\partial x_f^2} + \frac{\partial^2}{\partial z^2}\right) u_1' = -\overline{u}_0 \frac{\partial u_1'}{\partial x_f} - w_1' \frac{\partial \overline{u}_0}{\partial z}, \tag{16b}$$

$$\frac{\partial w_1'}{\partial t_f} + \frac{\partial p_1'}{\partial z} - b_1' - \frac{Fr_M^{4/3}}{Re_b}\left(\frac{\partial^2}{\partial x_f^2} + \frac{\partial^2}{\partial z^2}\right) w_1' = -\overline{u}_0 \frac{\partial w_1'}{\partial x_f}, \tag{16c}$$

$$\left(\frac{\partial^2}{\partial x_f^2} + \frac{\partial^2}{\partial z^2}\right) b_1' - w_1' = 0, \tag{16d}$$

$$\frac{\partial u_1'}{\partial x_f} + \frac{\partial w_1'}{\partial z} = 0, \tag{16e}$$

where the fast and slow fields co-evolve on t_f.

4.1.2 Multiple timescale stability analysis (MTQL)

We consider a small 2D domain with proportionate horizontal and vertical lengths (dimensionally, these lengths are of order H). We introduce a fluctuation streamfunction formulation,

$$u_1' = \frac{\partial \psi'}{\partial z}, \qquad w_1' = -\frac{\partial \psi'}{\partial x_f}. \tag{17}$$

On substituting (17) into (11), the mean field equations become,

$$\frac{\partial \overline{u}_0}{\partial t_s} + \left(\overline{u}_0 \cdot \nabla_s\right)\overline{u}_0 + \overline{w}_2 \frac{\partial \overline{u}_0}{\partial z} = -\nabla_s \overline{p}_0 + \frac{\partial}{\partial z}\left(\overline{\frac{\partial \psi'}{\partial z}\frac{\partial \psi'}{\partial x_f}}\right) + \frac{1}{Re_b}\frac{\partial^2 \overline{u}_0}{\partial z^2} + \overline{f}_0, \tag{18a}$$

$$\overline{b}_2 = \left(\nabla_s^2 + \frac{\partial^2}{\partial z^2}\right)^{-1} \overline{w}_2, \tag{18b}$$

$$\nabla_s \cdot \overline{u}_0 + \frac{\partial \overline{w}_2}{\partial z} = 0, \tag{18c}$$

and the fluctuation equations become,

$$\frac{\partial}{\partial t_f}\left(\frac{\partial \psi'}{\partial z}\right)+\left(\overline{u}_0\cdot\frac{\partial}{\partial x_f}\right)\left(\frac{\partial \psi'}{\partial z}\right)-\left(\frac{\partial \psi'}{\partial x_f}\right)\frac{\partial \overline{u}_0}{\partial z}=-\frac{\partial p_1'}{\partial x_f}+\frac{\alpha}{Re_b}\left(\nabla_f^2+\frac{\partial^2}{\partial z^2}\right)u', \quad (18d)$$

$$\frac{\partial}{\partial t_f}\left(-\frac{\partial \psi'}{\partial x_f}\right)-\left(\overline{u}_0\cdot\frac{\partial}{\partial x_f}\right)\left(\frac{\partial \psi'}{\partial x_f}\right)=-\frac{\partial p_1'}{\partial z}-b_1'+\frac{\alpha}{Re_b}\left(\nabla_f^2+\frac{\partial^2}{\partial z^2}\right)w', \quad (18e)$$

$$b_1'=\left(\frac{\partial^2}{\partial x_f^2}+\frac{\partial^2}{\partial z^2}\right)^{-1}\left(\frac{\partial \psi'}{\partial x_f}\right), \quad (18f)$$

$$\frac{\partial}{\partial x_f}\left(\frac{\partial \psi'}{\partial z}\right)-\frac{\partial}{\partial z}\left(\frac{\partial \psi'}{\partial x_f}\right)=0. \quad (18g)$$

On suppressing the x_s derivatives, following [3], the mean field equations reduce to a single equation

$$\frac{\partial \overline{u}_0}{\partial t_s}=\frac{\partial}{\partial z}\left(\overline{\frac{\partial \psi'}{\partial z}\frac{\partial \psi'}{\partial x_f}}\right)+\frac{1}{Re_b}\frac{\partial^2 \overline{u}_0}{\partial z^2}+\overline{f}_0. \quad (19a)$$

We eliminate the pressure in the fluctuation equations by taking ∂_z of (18d) summed with $-\partial_{x_f}$ of (18e), to obtain

$$\left(\frac{\partial}{\partial t_f}+\overline{u}_0\frac{\partial}{\partial x_f}\right)\left(\frac{\partial^2}{\partial x_f^2}+\frac{\partial^2}{\partial z^2}\right)\psi'=\left(\frac{\partial \psi'}{\partial x_f}\right)\left(\frac{\partial^2 \overline{u}_0}{\partial z^2}\right)+\frac{\partial b_1'}{\partial x_f}+\frac{\alpha}{Re_b}\left(\frac{\partial^2}{\partial x_f^2}+\frac{\partial^2}{\partial z^2}\right)^2\psi', \quad (19b)$$

$$b'=-\left(\frac{\partial^2}{\partial x_f^2}+\frac{\partial^2}{\partial z^2}\right)^{-1}\left(\frac{\partial \psi'}{\partial x_f}\right). \quad (19c)$$

We now focus on solving the slow-fast quasilinear system (19) to interrogate its approach to marginal stability. To first develop intuition before delving into the mathematical framework, consider the canonical example of self-organised criticality in which grains pour onto a flat plate from above, piling up. Generally over time, the grain pile is maintained at a special angle called the angle of repose. Mini-avalanches occur intermittently to maintain this angle. This seemingly distinct system has a direct analogy to our stellar turbulent system, in which the *amplitude of the fluctuations* (rather than mini avalanches) intermittently act to maintain the *mean flow* (rather than the slope of the grains) at the *stability criterion* (rather than the angle of repose). Here, the stability criterion corresponds to the growth rate of the system being maintained at zero.

To formulate equations which can be solved in a manner consistent with slow-fast quasilinear algorithms [10], we first express the fluctuation streamfunction and buoyancy in separable form,

$$\psi'(x_f,z,t_f,t_s)=A(t_s)\hat{\psi}(z,t_s)\exp(\sigma t_f+ik(t_s)x_f)+\text{complex conjugate}, \quad (20a)$$

$$b'(x_f,z,t_f,t_s)=A(t_s)\hat{b}(z,t_s)\exp(\sigma t_f+ik(t_s)x_f)+\text{complex conjugate}, \quad (20b)$$

where their vertical structure is denoted by hatted variables, the complex growth rate is $\sigma = \sigma_r + i\sigma_i$, and $A(t_s)$ is the amplitude of magnitude $|A(t_s)|$. The Reynolds stress terms can now be cleanly written as,

$$\frac{\partial}{\partial z}\left(\overline{\frac{\partial \psi'}{\partial z}\frac{\partial \psi'}{\partial x_f}}\right) = |A(t_s)|^2 ik\left[\frac{\partial}{\partial z}\left(\hat{\psi}\frac{\partial \hat{\psi}^*}{\partial z} - \hat{\psi}^*\frac{\partial \hat{\psi}}{\partial z}\right)\right] \equiv |A(t_s)|^2 RS. \tag{21}$$

On substituting (20) into (19), we obtain

$$\frac{\partial \overline{u}_0}{\partial t_s} = |A(t_s)|^2 RS + \frac{1}{Re_b}\frac{\partial^2 \overline{u}_0}{\partial z^2} + \overline{f}_0, \tag{22a}$$

$$(\sigma + ik\overline{u}_0)\left(-k^2 + \frac{\partial^2}{\partial z^2}\right)\hat{\psi} = ik\frac{\partial^2 \overline{u}_0}{\partial z^2}\hat{\psi} - ik\hat{b} + \frac{\alpha}{Re_b}\left(-k^2 + \frac{\partial^2}{\partial z^2}\right)^2\hat{\psi}, \tag{22b}$$

$$\hat{b} = \left(k^2 - \frac{\partial^2}{\partial z^2}\right)^{-1} ik\hat{\psi}. \tag{22c}$$

We treat the fluctuation equations (22*bc*) as a linear, autonomous eigenvalue problem. On writing the system as a linear dynamical operator $\mathcal{L}X = 0$, we obtain

$$\mathcal{L}X = \begin{pmatrix} (\sigma + ik\overline{u}_0)\left(\frac{\partial^2}{\partial z^2} - k^2\right) - ik\frac{\partial^2 \overline{u}_0}{\partial z^2} - \frac{\alpha}{Re_b}\left(\frac{\partial^2}{\partial z^2} - k^2\right)^2 & ik \\ -ik & k^2 - \frac{\partial^2}{\partial z^2} \end{pmatrix}\begin{pmatrix}\hat{\psi}\\ \hat{b}\end{pmatrix} = 0 \tag{23a}$$

with periodic boundary conditions in z. We define the inner product as

$$(X_1|X_2) = \int_0^{l_z} X_1(z)X_2^*(z)\mathrm{d}z \quad \forall \quad (X_1, X_2). \tag{23b}$$

The adjoint operator $\mathcal{L}^\dagger$ satisfies $(\mathcal{L}X_1|X_2) = (X_1|\mathcal{L}^\dagger X_2)$ and is calculated using integration by parts to obtain

$$\mathcal{L}^\dagger X^\dagger = \begin{pmatrix} (\sigma^* - ik\overline{u}_0)\left(\frac{\partial^2}{\partial z^2} - k^2\right) - 2ik\frac{\partial \overline{u}_0}{\partial z}\frac{\partial}{\partial z} - \frac{\alpha}{Re_b}\left(\frac{\partial^2}{\partial z^2} - k^2\right)^2 & ik \\ -ik & k^2 - \frac{\partial^2}{\partial z^2} \end{pmatrix}\begin{pmatrix}\hat{\psi}^\dagger\\ \hat{b}^\dagger\end{pmatrix} = 0. \tag{23c}$$

Note that $\mathcal{L}$ is not a self-adjoint operator as $\mathcal{L} \neq \mathcal{L}^\dagger$. To obtain an expression for the temporal evolution of the growth rate with respect to the slow time, we take the time derivative of (23a),

$$\mathcal{L}\frac{\partial X}{\partial t_s} = -\frac{\partial \mathcal{L}}{\partial t_s}X \tag{24a}$$

where the slow time derivative of $\mathcal{L}$ is

$$\frac{\partial \mathcal{L}}{\partial t_s} = \begin{pmatrix} \left(\frac{\partial \sigma}{\partial t_s} + ik\frac{\partial \overline{u}_0}{\partial t_s}\right)\left(\frac{\partial^2}{\partial z^2} - k^2\right) - ik\frac{\partial^2}{\partial z^2}\left(\frac{\partial \overline{u}_0}{\partial t_s}\right) & 0 \\ 0 & 0 \end{pmatrix} + \frac{\mathrm{d}k}{\mathrm{d}t_s}M. \tag{24b}$$

The above matrix is singular as it has a zero determinant. In accordance with the Fredholm alternative, for (24a) to be solvable, the RHS of (24b) must be orthogonal to corresponding null eigenvector $X^\dagger$, i.e.,

$$\left(\mathcal{L}\frac{\partial X}{\partial t_s}|X^\dagger\right) = \left(\frac{\partial X}{\partial t_s}\left|\mathcal{L}^\dagger X^\dagger\right.\right) = \left(\frac{\partial X}{\partial t_s}|0\right). \tag{24c}$$

where $X^\dagger = [\hat{\psi}(z), \hat{b}(z)]^T$. Therefore,

$$\left(\frac{\partial \mathcal{L}}{\partial t_s}X|X^\dagger\right) = 0. \tag{24d}$$

Substituting for mean flow equation (22a) into (24b), we derive a solvability condition

$$C_1\frac{\mathrm{d}\sigma}{\mathrm{d}t_s} = C_2|A(t)|^2 + C_3 + C_4\frac{\mathrm{d}k}{\mathrm{d}t} \tag{25a}$$

which describes how σ changes with with respect to slow time. Here, $C_4 = \partial_k\sigma$. As long as we insist that the fastest growing mode has a zero growth rate (i.e., that $\sigma = 0$ is a local maximum over k), then $\partial_k\sigma$ vanishes for the mode of interest. The coefficients C_1, C_2, and C_3 are

$$C_1 = \frac{1}{ik}\int_0^{l_z}\left[\hat{\psi}^{\dagger *}\left(\frac{\partial^2}{\partial z^2} - k^2\right)\hat{\psi}\right]\mathrm{d}z, \tag{25b}$$

$$C_2 = \int_0^{l_z} RS\left[\hat{\psi}\left(\frac{\partial^2}{\partial z^2} + k^2\right)\hat{\psi}^{\dagger *} + 2\frac{\partial\hat{\psi}}{\partial z}\frac{\partial\hat{\psi}^{\dagger *}}{\partial z}\right]\mathrm{d}z, \tag{25c}$$

$$C_3 = \int_0^{l_z}\left[\left(\overline{f} + \frac{1}{Re_b}\frac{\partial^2\overline{u}_0}{\partial z^2}\right)\left(\hat{\psi}\left(\frac{\partial^2}{\partial z^2} + k^2\right)\hat{\psi}^{\dagger *} + 2\frac{\partial\hat{\psi}}{\partial z}\frac{\partial\hat{\psi}^{\dagger *}}{\partial z}\right)\right]\mathrm{d}z. \tag{25d}$$

The total temporal evolution of the real part of the eigenvalue $\sigma_r(\overline{u}_0, \partial_z\overline{u}_0, k)$ is

$$\frac{\mathrm{d}\sigma_r}{\mathrm{d}t_s} = \left(\frac{\partial\sigma_r}{\partial t_s}\right)_k + \frac{\mathrm{d}k}{\mathrm{d}t_s}\left(\frac{\partial\sigma_r}{\partial k}\right)_{\overline{u}_0, \partial_z\overline{u}_0}, \tag{26}$$

however, the second term on the right-hand side vanishes for the mode of interest, provided that $k(t)$ corresponds to the fastest growing unstable mode. Hence, the total temporal evolution is $\mathrm{d}_{t_s}\sigma_r = (\partial_{t_s}\sigma_r)_k$, and simply corresponds to the evolution of the growth rate for a given wavenumber. In fact, we can obtain the evolution of the growth rate for a given k by dividing (25a) by C_1 as follows,

$$\left(\frac{\partial\sigma_r}{\partial t_s}\right)_k = \mathrm{Re}\,\left(\frac{C_3}{C_1}\right) - \mathrm{Re}\,\left(-\frac{C_2}{C_1}\right)|A(t)|^2. \tag{27}$$

The final crucial piece to the method of solution of slow-fast quasilinear systems in [10] is that once $\sigma_r = 0$, it stays zero (i.e., $\partial_{t_s}\sigma_r = 0$), thus maintaining the system at marginal stability. On rearrangement of (27) such that $\partial_{t_s}\sigma_r = 0$ is satisfied, we obtain an expression for the amplitude of the fluctuations that guarantees the marginal stability of the system,

$$|A(t)|^2 = \begin{cases}\sqrt{\mathrm{Re}\,\left(\frac{C_3}{C_1}\right)\mathrm{Re}\,\left(-\frac{C_2}{C_1}\right)^{-1}} & \text{if } \sigma_r = 0 \text{ and } \mathrm{Re}\,\left(\frac{C_3}{C_1}\right), \mathrm{Re}\,\left(-\frac{C_2}{C_1}\right) > 0, \\ 0 & \text{otherwise.}\end{cases} \tag{28}$$

Hence, the amplitude of the fluctuations is intermittently non-zero when $\sigma_r = 0$.

4.2 Nonlinear evolution of energy and growth rate

The three systems, DNS, STQL, and MTQL, are solved using the Python software package Dedalus burns2020. Pseudo-spectral methods in Dedalus are used with a second-order Runge-Kutta time-stepping scheme. The three simulations are run in a vertical domain $L_z = 2\pi$ using a Fourier basis with 128 gridpoints. The horizontal domain length for DNS and STQL simulations is $L_x = (2\pi/k_c)Fr_M^{4/3}$ (where we consider $k_c = 0.5$). In order to ensure that the MTQL simulation captures dynamics on the same scale as the DNS and STQL simulations and to thus facilitate a direct comparison between the three simulations, we implement a wavenumber cutoff, $k_{\text{cutoff}} = 2\pi Fr_M^{4/3}/k_c$. The search over wavenumbers occurs only when $k > k_{\text{cutoff}}$, such that the identified fastest growing mode does not correspond to large-scale dynamics not captured by the DNS and STQL simulations. All three simulations are forced from rest, i.e., with zero initial velocity.

We compare the total energy in the simulations, given by

$$E = \frac{1}{2}\int_0^{L_x} \mathrm{d}\boldsymbol{x}_s \int_0^{L_z} \mathrm{d}z \left(u^2 + Fr_M^{8/3} w^2 + b^2\right). \tag{29}$$

Note that for the STQL and MTQL simulations, the reconstructed fields are used to calculate the total energy, i.e., $(u, b, w) \to (u_0 + Fr_M^{2/3}u_1', Fr^{2/3}w_{-1}', Fr^{2/3}b_{-1}')$. For the dimensionless parameters $Fr_M = 0.1$ and $Re_b = 1$, the total energy from the three algorithms is compared in Figure 2. The energy in these three simulations is compared to the energy in the mean flow only scenario (termed "MF only" in the legend) governed by

$$\frac{\partial u}{\partial t} = f + \frac{1}{Re_b}\frac{\partial^2 u}{\partial z^2}, \tag{30a}$$

which when solved for $f = F_0\cos(z)/Re_b$ gives an amplitude $A(t) = F_0 Re_b(1-\exp(-k^2 t/Re_b))$ and a mean flow energy $E_{MF} = A^2/4$.

There is generally excellent agreement between the energy at which the MTQL and STQL simulations equilibrate at. Additionally, the STQL and MTQL energy falls within the 'undulations' of the energy in the DNS. When run out for longer times (not shown), the DNS equilibrates at a similar energy to the STQL and MTQL simulations.

The three simulations fall off the mean flow curve (broken line) at different times. As expected, the MTQL solution reaches steady-state the fastest as the fluctuation fields are instantaneously non-zero per (28) and adjust the mean flow instantaneously. The STQL solution takes longer to fall off the mean flow curve and reach steady-state as the adjustment of the fluctuation fields, while present, is not instantaneous. The DNS simulation takes longest to reach equilibrium; this is expected as there is no formal separation of spatiotemporal scales here.

The evolution of the growth rate in the MTQL simulation is plotted in the inset of Figure 2. Several key points bear discussion. First, as the simulation is forced from rest, the real part of the growth rate over all horizontal wavenumbers is initially negative (i.e., the system is stable). However as the flow develops with time, the growth rate approaches zero "from below", i.e., σ becomes increasingly less negative. Once $\sigma_r = 0$, the amplitude of the fluctuations in (28) maintains the system at marginal stability, such that from that timestep onward, $\partial_{t_s}\sigma_r = 0$. The sudden jump of $|A|^2$ to a finite, non-zero value when

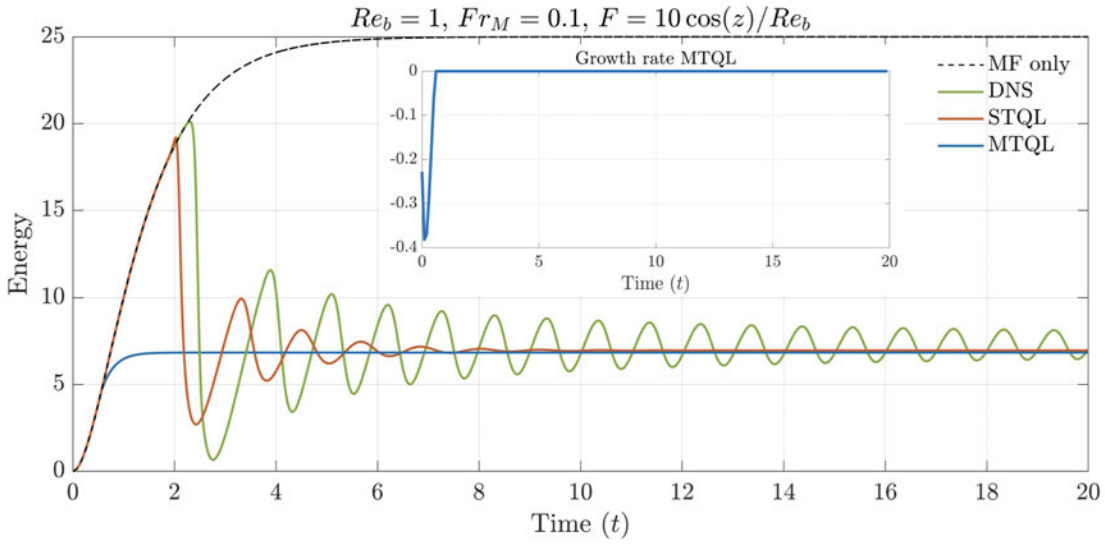


Figure 2: Total energy from DNS (green), STQL (red) and MTQL (blue) simulations for $Re_b = 1$ and $Fr_M = 0.1$, compared against the analytical solution for a system with mean flow only (black dashed line). The forcing in each simulation is given by $f = 10\cos(z)/Re_b$. Note that the time against which the energy is plotted here is the time of the DNS, i.e., the slow time t_s. The inset describes the evolution of the growth rate during the MTQL simulation.

$\sigma_r = 0$ causes a sharp change in the slow-time evolution of not just the growth rate but also the energy.

4.3 Steady exact coherent states

We now turn our attention from calculated quantities to the exact coherent states of the three systems. Given the different timescales on which the DNS, STQL, and MTQL systems reach steady-state, to effect a fair comparison between the three solutions, we consider steady-state snapshots. To identify when the simulations have reached steady-state, we plot a Hövmuller diagram for the horizontal velocity (second row in Figure 3). The simulations have clearly reached steady-state by $t = 20$.

The horizontal, vertical and buoyancy field snapshots from all three simulations are plotted in Figure 3. For notational convenience, we introduce $\epsilon = \alpha^{1/2}$. To compare the 2D anisotropic structure, the horizontal and vertical velocities are normalized by U. Hence, the velocities $(u, \epsilon^2 w)$ are plotted for the DNS, while $(u_0 + \epsilon u_1', \epsilon w_{-1}')$ are plotted for STQL and MTQL solutions. In general, there is excellent agreement between the DNS, STQL and MTQL results. This suggests that a quasilinear description for low Péclet flows is valid, at least for the values of Re_b and Fr_M in Figures 2-3.

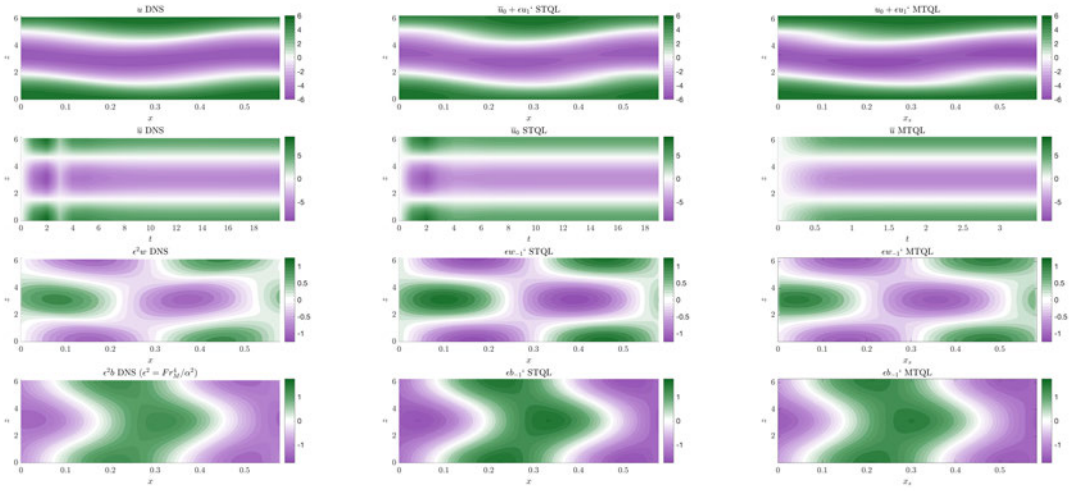


Figure 3: Snapshots at $t = 20$ for the simulations in Figure 2. [first row] u, [third row] w, [fourth row] b, plotted against x ($\boldsymbol{x}_s$ for STQL and MTQL) and z. A vertically averaged Hövmuller plot is shown for u.

4.4 Bursting events

For different domain lengths, 'bursting' events could arise in which the inverse ratio of coefficients in (28) are negative. For notational convenience, we define

$$\alpha_r = \mathrm{Re}\,\left(\frac{C_3}{C_1}\right), \qquad \beta_r = \mathrm{Re}\,\left(-\frac{C_2}{C_1}\right). \tag{31}$$

If $\alpha_r < 0$, the positive growth rate would decay on setting $|A|^2 = 0$. However, if $\beta_r < 0$ then the system undergoes a bursting event in which the positive growth rate increases with $|A|^2 = 0$, and even further with $|A|^2 > 0$. Physically, this situation means scale separation breaks down and fast transient dynamics need to be incorporated to maintain the system at marginal stability.

Given the explosive growth associated with bursting events, how can we algorithmically solve the system when $\beta_r > 0$? We outline several options here which apply the techniques developed for toy slow-fast quasilinear systems in [5] to the slow-fast quasilinear system for stellar turbulence, and save their algorithmic implementation for future work. Specifically, we outline how to: (a) evolve the system to a non-bursting state, (b) identify when to stop this evolution, and (c) re-initialise the MTQL system once the growth rate is negative.

From the previous timestep when $\alpha_r, \beta_r > 0$, we have the amplitude A, the wavenumber k corresponding to the fastest growing mode, and the streamfunction $\hat{\psi}$ (from which the 2D streamfunction can be recovered by computing the outer product of $\hat{\psi}$ and e^{ikx_f}, where $x_f = x_s/\alpha$). The techniques to deal with a bursting event usually involve co-evolving a system of equations until $\alpha_r, \beta_r > 0$, and then re-initialising the MTQL system. There are at least three possible techniques for initialising the bursting algorithm:

1. Co-evolution of DNS in a 2D streamfunction formulation

   This approach would evolve (22) on a single timescale. Evolution on the fast timescale would be the most judicious choice as A is large, is balanced by $\partial_{t_s}\overline{u}_0$, and evolves on the fast timescale. The resulting set of equations is similar to the 2D DNS system in (3), except that it splits the spatial scale into a slow and fast component (i.e., $x = \overline{x} + x'$), and retains the eddy-eddy non-linearities.

2. Co-evolution of a STQL-like system

   This approach involves evolving a system similar to the STQL equations in (16), however, in a streamfunction formulation. As such, we eliminate pressure. The evolution considers a single wavenumber k, corresponding to the fastest growing mode from the previous timestep when $\alpha_r, \beta_r > 0$, to reconstruct the streamfunction.

3. A 'gradient descent' strategy

   This approach considers the dominant balance of terms,

   $$-\frac{\partial\overline{w'u'}}{\partial z} \sim \frac{\partial\overline{u}_0}{\partial t_s} \tag{32}$$

   to adopt a hybrid eigenvalue timestepping, rather than co-evolution. An $O(1)$ positive number is arbitrarily assigned to the amplitude A, such that the amplitude is constant. The following equation is timestepped,

   $$\frac{\partial\overline{u}_0}{\partial t_s} = \frac{\partial}{\partial z}RS \tag{33}$$

   and its eigenvalue is computed to identify the fastest growing mode. While more crude than the first two approaches listed above, this approach has been shown to work well for toy problems [5] and the $Pr \sim O(1)$ system of equations [3].

4. An appropriate rescaling strategy

   This approach involves finding a scaling such that the equations are free of ϵ and non-stiff. The resulting set of equations are then evolved on the fast timescale and without a forcing term, thus guaranteeing a reduction of the growth rate to zero. Ignoring the forcing term is a valid approximation when the fluctations are large, as they are in a bursting event.

Once the co-evolution begins, the next question to address is when to stop it. In general, this involves solving the eigenvalue problem corresponding to the linearised dynamics, $\mathcal{L}X = 0$, as a diagnostic to monitor the (anticipated) decrease of the growth rate σ_r towards zero during the co-evolution. Once $\sigma_r < 0$, one would switch back to evolving the MTQL system.

The third and final question to address is how to re-initialise the MTQL once the growth rate goes negative. To identify the wavenumber corresponding to the fastest growing mode, the Fourier spectrum in x is computed. (It might be necessary to vertically integrate the output from the previous timestep prior to computing the Fourier spectrum.) Then the signs of α_r, β_r are checked and the usual conditions outlined in §4.1.2 are applied. As the wavenumber k is discrete, when the MTQL system is re-initialised, it is possible that the

wavenumber does not exactly correspond to the fastest growing mode. There are several workarounds, such as changing the domain size with time. Rather than implementing time-dependent coefficients, a linear transformation between coordinates can be used such that instead of a domain length $[0, L_x(t)]$, we consider a domain length of $[0, 1]$. For instance, $x_{\text{computational}} = 2\pi\, x_{\text{physical}}/L(t)$, and by the chain rule $\partial_{x_{\text{computational}}} = 2\pi\, \partial_{x_{\text{physical}}}/L(t)$, where $2\pi/L(t)$ is the wavenumber and the computational domain is $[0, 2\pi]$.

5 Conclusions

This study has established the main regimes of strongly stratified turbulence at low Prandtl number and demonstrated the approach of these turbulent flows towards marginal stability. Stratified turbulence in stars cannot be directly measured. Given the observational difficulties, previously published studies either simulate stellar flows at measured Froude, Prandtl and Péclet numbers (e.g. [6, 4]) or invoke physical arguments to balance terms and assess resulting scaling relationships (e.g. [11]). The present work adopts a different approach, using numerical evidence of anisotropic flows, scale separation, and velocity layering as motivation for conducting formal, multiple scale analyses of the equations governing the dynamics of stars at low Prandtl number. Two multiple scale models are developed, one each for low and high Péclet number flows. A central feature of the derivation is that the generalised quasilinear form of the asymptotically-reduced equations, in which the fluctuation dynamics are shown to be linear about the mean flow and the fluctuations influence the mean flow via their induced Reynolds stress divergence, naturally emerges and is not invoked in an ad hoc fashion to close the system. Through multiple scale asymptotics, this study provides a formal justification for the application of quasilinear approximations to descriptions of strongly stratified stellar turbulence.

The identification of distinguished limits in turbulent behaviour is a core motivation that drives the development of the multiple scale models presented here. A second key outcome of this study is the scaling relationships for the aspect ratio that emerge via the two-scale asymptotics for high Pe ($\alpha \sim Fr$) and low Pe ($\alpha \sim Fr_M^{4/3}$) flows. For low Pe flows, our $\alpha \sim Fr_M^{4/3}$ theoretical prediction is validated by numerical simulations in [4]. For high Pe flows, our $\alpha \sim Fr$ theoretical prediction indicates that there is no theoretical basis for the scalings in [6].

While a star's outerscale Péclet number can be estimated from stellar observations, the emergent turbulent Péclet number can only be deduced for a given model. As the vast majority of stars, including our Sun, have a global scale $Pe \gg 1$ but $Pr \ll 1$, pinpointing the bound between adiabatic stratified and diffusive stratified turbulence is valuable for predicting turbulent characteristics based on stellar observations of the outerscale Pe. Arguably, the primary contribution of this work is the identification of regimes of stellar turbulence. Crucially, the momentum and buoyancy fluctuation equations in the multiscale models offer a systematic theoretical basis for regime identification. We construct a full regime diagram, identifying adiabatic stratified turbulence, diffusive stratified turbulence, and non-turbulent, viscous dynamical behaviours. Our theoretical identification of regimes agrees with numerical simulations, whose behaviour we classify per [4].

Solutions of the multiscale slow-fast quasilinear system reveals its approach to and main-

tenance of marginal stability. The system is forced from rest and hence the initial growth rate is negative but becomes increasingly less negative over time. Once the growth rate is zero, the amplitude of the fluctuation fields acts intermittently to maintain the growth rate at zero. There is excellent agreement between the steady, coherent states of the DNS, STQL, and MTQL solutions. The energy in all three systems equilibrates at similar values. The MTQL system reaches steady-state first as the system adjusts instantaneously to the fluctuations via the divergence of the Reynolds stress restoring the system to marginal stability. The STQL system is the second to reach steady-state as its adjustment of the mean flow by the Reynolds stress divergence is faster than the response of the fully nonlinear DNS.

The insight obtained from the multiscale models notwithstanding, in order to develop a truly astrophysically relevant theory for stratified stellar turbulence, physical processes such as rotation must be incorporated. The vast majority of stars rotate. Indeed, differential rotation typically is the main source of shear in rotating stars. This study assumes that rotation is not needed to achieve the large horizontal scales $\boldsymbol{x}_{\perp s}$; however, on these scales Rossby numbers are small, indicating that the large-scale dynamics are strongly affected by rotation. Magnetohydrodynamics too must be incorporated, given that most stars are expected to be magnetized. Finally, a two-scale expansion in z might enable the full turbulent mechanism proposed by [12] to be realised, which we save for future work.

6 Acknowledgements

Thank you to my GFD co-advisors, Pascale Garaud, Greg Chini, and Colm-cille Caulfield, for their insight, availability, and unfailing support. I started the GFD school knowing very little about stars and not having researched turbulence; that I could give a coherent presentation by the summer's end is testament to their advising. To Keaton Burns for always being open to discussing my Dedalus-related questions, no matter how small. To all participants of the 2022 GFD Summer School for a memorable summer and for stimulating in-person discussions that I thoroughly enjoyed after a few years of remote scientific interactions during the pandemic. To Stefan Llewellyn Smith and Colm-cille Caulfield for their directorship and to Julie Hildebrandt and Janet Fields for their organization of the first summer school post-pandemic. To my GFD fellows cohort (Claire Valva, Iury Simoes-Sousa, Ludovico Giorgini, Rui Yang, Ruth Moorman, Sam Lewin, Tilly Woods) for the camaraderie, the nights spent in Walsh Cottage, the swims, and the softball. Thank you.

References

[1] P. Billant and J.-M. Chomaz, *Self-similarity of strongly stratified inviscid flows*, Physics of fluids, 13 (2001), pp. 1645–1651.

[2] G. Brethouwer, P. Billant, E. Lindborg, and J.-M. Chomaz, *Scaling analysis and simulation of strongly stratified turbulent flows*, Journal of Fluid Mechanics, 585 (2007), pp. 343–368.

[3] G. P. Chini, G. Michel, K. Julien, C. B. Rocha, and P. C. Colm-cille, *Exploiting self-organized criticality in strongly stratified turbulence*, Journal of Fluid Mechanics, 933 (2022).

[4] L. Cope, P. Garaud, and C.-c. P. Caulfield, *The dynamics of stratified horizontal shear flows at low Péclet number*, Journal of Fluid Mechanics, 903 (2020).

[5] A. Ferraro, *Exploiting marginal stability in slow-fast quasilinear dynamical systems*, tech. rep., EPFL, 2022.

[6] P. Garaud, *Horizontal shear instabilities at low Prandtl number*, The Astrophysical Journal, 901 (2020), p. 146.

[7] ——, *Journey to the center of stars: The realm of low prandtl number fluid dynamics*, Physical Review Fluids, 6 (2021), p. 030501.

[8] F. Lignieres, *The small-péclet-number approximation in stellar radiative zones*, arXiv preprint astro-ph/9908182, (1999).

[9] A. Maffioli and P. A. Davidson, *Dynamics of stratified turbulence decaying from a high buoyancy reynolds number*, Journal of Fluid Mechanics, 786 (2016), pp. 210–233.

[10] G. Michel and G. Chini, *Multiple scales analysis of slow–fast quasi-linear systems*, Proceedings of the Royal Society A, 475 (2019), p. 20180630.

[11] V. A. Skoutnev, *Critical Balance and Scaling of Stably Stratified Turbulence at Low Prandtl Number*, arXiv preprint arXiv:2205.01540, (2022).

[12] J.-P. Zahn, *Circulation and turbulence in rotating stars*, Astronomy and Astrophysics, 265 (1992), pp. 115–132.

REPORT DOCUMENTATION PAGE	1. Report No. WHOI-2025-11	2.	3. Recipient's Accession No.
4. Title and Subtitle 2022 Program of Study: Data-driven GFD			5. Report Date December 2025 6.
7. Author(s) Colm-cille Caulfield and Stefan Llewellyn Smith			8. Performing Organization Rept. No. 10. Project/Task/Work Unit No.
9. Performing Organization Name and Address Woods Hole Oceanographic Institution			11. Contract(C) or Grant(G) No. (C) (G) OCE-1829864
12. Sponsoring Organization Name and Address National Science Foundation Program Manager Dr. Baris M. Uz 15 Eisenhower Avenue Alexandria, VA 22314			13. Type of Report & Period Covered Technical Report 14.

15. Supplementary Notes

This report should be cited as: Woods Hole Oceanographic Institution Technical Report, WHOI-2025-11, https://doi.org/

16. Abstract (Limit: 200 words)

The 2022 GFD Program theme was Data-Driven GFD with Professors Peter Schmid of King Abdullah University of Science and Technology (KAUST) and Laure Zanna of New York University serving as principal lecturers. Together they introduced the masked audience in the re-opened cottage and on the better-ventilated porch to a fascinating mixture of data-driven methods and their potential application to fluid mechanics in general and GFD in particular. The first ten chapters of this volume document these lectures, each prepared by teams of the summer's GFD fellows.

.

17. Document Analysis

a. Descriptors: Geophysical fluid dynamics, Data-driven GFD

b. Identifiers/ Open-Ended Terms

c. COSATI Field/ Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Approved for public release, distribution unlimited		244
	20. Security Class (This Page)	22. Price

REPORT DOCUMENTATION PAGE	1. Report No. WHOI-2025-11	2.	3. Recipient's Accession No.
4. Title and Subtitle 2022 Program of Study: Data-driven GFD			5. Report Date December 2025 6.
7. Author(s) Colm-cille Caulfield and Stefan Llewellyn Smith			8. Performing Organization Rept. No. 10. Project/Task/Work Unit No.
9. Performing Organization Name and Address Woods Hole Oceanographic Institution			11. Contract(C) or Grant(G) No. (C) (G) OCE-1829864
12. Sponsoring Organization Name and Address National Science Foundation Program Manager Dr. Baris M. Uz 15 Eisenhower Avenue Alexandria, VA 22314			13. Type of Report & Period Covered Technical Report 14.

15. Supplementary Notes

This report should be cited as: Woods Hole Oceanographic Institution Technical Report, WHOI-2025-11, https://doi.org/10.1575/1912/72543

16. Abstract (Limit: 200 words)

The 2022 GFD Program theme was Data-Driven GFD with Professors Peter Schmid of King Abdullah University of Science and Technology (KAUST) and Laure Zanna of New York University serving as principal lecturers. Together they introduced the masked audience in the re-opened cottage and on the better-ventilated porch to a fascinating mixture of data-driven methods and their potential application to fluid mechanics in general and GFD in particular. The first ten chapters of this volume document these lectures, each prepared by teams of the summer's GFD fellows.

.

17. Document Analysis

a. Descriptors: Geophysical fluid dynamics, Data-driven GFD

b. Identifiers/ Open-Ended Terms

c. COSATI Field/ Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
Approved for public release, distribution unlimited		244
	20. Security Class (This Page)	22. Price