UNIVERSITY OF AMSTERDAM

UvA-DARE (Digital Academic Repository)

Stochastic models for unresolved scales in ocean flows

Verheul, N.

Publication date
2022

Document Version
Final published version

Link to publication

Citation for published version (APA):
Verheul, N. (2022). *Stochastic models for unresolved scales in ocean flows*. [Thesis, fully internal, Universiteit van Amsterdam].

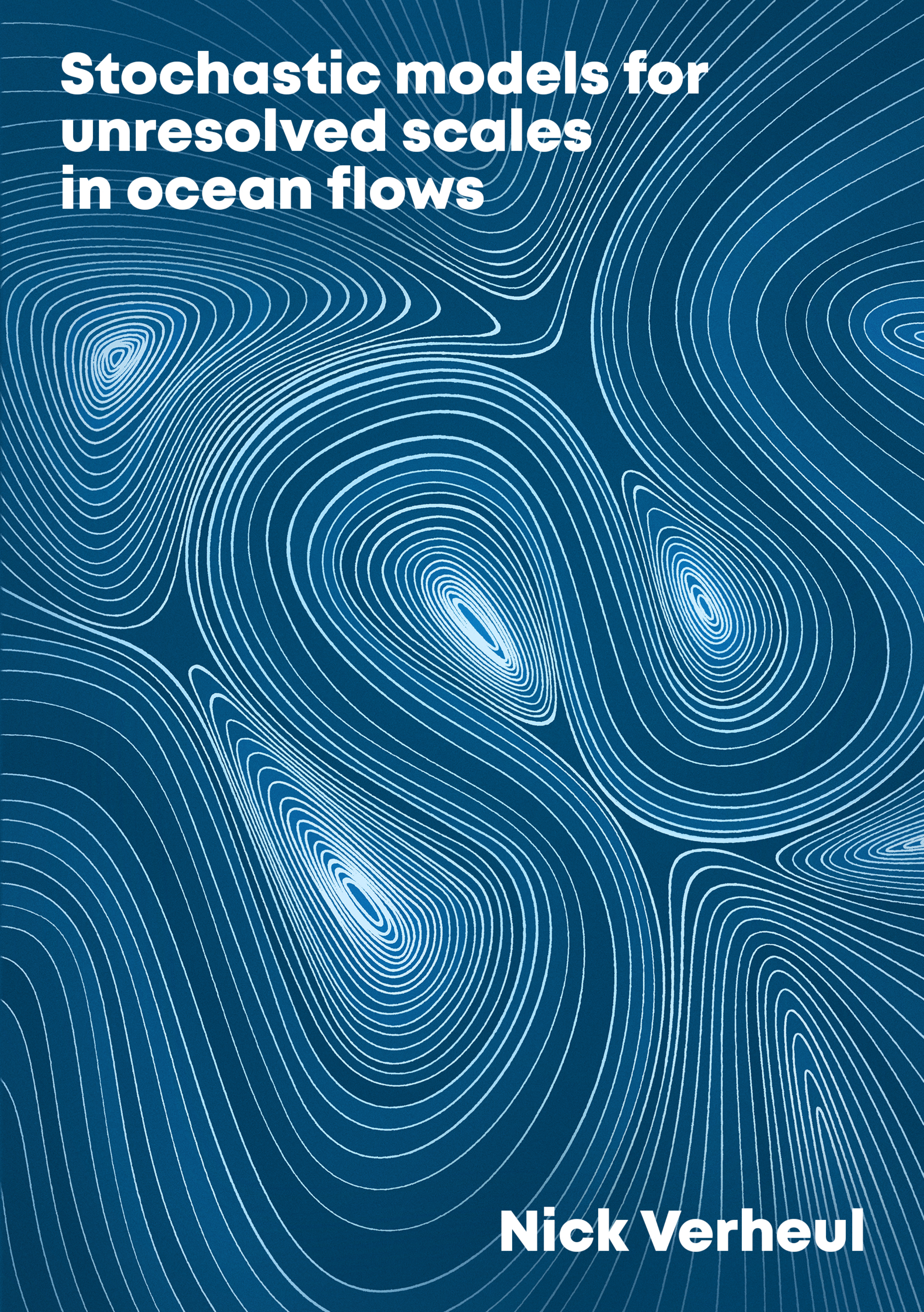
General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: https://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

UvA-DARE is a service provided by the library of the University of Amsterdam (https://dare.uva.nl)

Download date:11 Feb 2023

Stochastic models for unresolved scales in ocean flows



Nick Verheul

Stochastic Models For Unresolved Scales In Ocean Flows

by

Nick Verheul

This research was funded by the Netherlands Organisation for Scientific Research (NWO) through the Vidi project “Stochastic models for unresolved scales in geophysical flows”. The research work was carried out in the Scientific Computing group of the Centrum Wiskunde & Informatica (CWI), the Dutch national research institute for mathematics and computer science.

ISBN: 978-94-6421-968-5
© Copyright by *Nick Verheul*, 2022. All Rights Reserved

Stochastic Models for Unresolved Scales in Ocean Flows

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. ir. P.P.C.C. Verbeek
ten overstaan van een door het College voor Promoties ingestelde comissie,
in het openbaar te verdedigen in de Agnietenkapel
op maandag 19 december 2022, te 14:00 uur

door Nick Verheul
geboren te Naarden

Promotiecommissie

Promotor:

prof. dr. D.T. Crommelin	Universiteit van Amsterdam

Copromotor:

prof. dr. ir. J.E. Frank	Universiteit Utrecht

Overige leden:

prof. dr. R.P. Stevenson	Universiteit van Amsterdam
prof. dr. P.J.C. Spreij	Universiteit van Amsterdam
prof. dr. ir. A.G. Hoekstra	Universiteit van Amsterdam
prof. dr. K.P. Veroy-Grepl	Technische Universiteit Eindhoven
prof. dr. ir. H.A. Dijkstra	Universiteit Utrecht
dr. ir. W.N. Edeling	Centrum Wiskunde & Informatica

Faculteit:

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

Voor mijn ouders,
Marco en Mieke
die altijd vooraan staan om mij te steunen en aan te moedigen

Acknowledgements

To come out of the gates strong, I want to start by thanking my parents, Marco and Mieke, for always being there for me, come rain or sunshine. Your love, care, and humor pulled me through some tough times. These, and many others, are traits I can only hope to emulate myself.

I would like to thank my supervisor Daan Crommelin for his support, both scientifically and socially, during my PhD. Thank you for always treating me with kindness, respect, and trust. During the times of research that could feel like a slog, I could always find a drive in wanting to share new findings with you. I feel lucky to have had you as my supervisor.

Thanks to Ramana Kleyn for designing and illustrating the cover of this thesis. It was a joy to collaborate with you.

Thanks also to Jan Viebahn, co-author of the second chapter of this thesis and, briefly, office-mate at our institute. It was very motivating to work with you: discussing mathematics and physics, but also the emotional and philosophical aspects of both life and research. I would love to work with you again some day.

Also a huge thanks to Julie-Anne, Marloes, and Sana: for putting up with me through the late nights working on this thesis. Thank you for your support, encouragement, and for your respective parts in pulling me through to the end.

For the close friendships and joy in sharing in our hobbies I want to thank Balder, Lex, Luuk, Niels, and Yin. The late nights in Uilenstede (or whatever bar wasn't actively trying to destroy our eardrums), the many, many games, and the D&D adventures, have kept me sane throughout the last couple years. Thanks to Elo, Maerten, Ramana, and Rik for your continued close friendship. If you had asked me when we were six, even in my naivety, I couldn't have imagined we'd still be so close after all these years. But it makes me very happy that we have managed to subvert my own expectations so well. I look forward to more adventures with you all (both on some more city trips and those of the fictional kind).

Thank you Angela and Sophie for friendships that, regardless of how much time has passed, always effortlessly pick up right where we left off.

Thanks also go out to Davey, Dylan, and Robin, the closest people I have to brothers (on top of being cousins). May we continue to enjoy sharing our love for music and many more gaming weekends.

To my TWINFO friends: Ian, I want to thank you for many intellectually motivating discussions and your friendship. Remco, thanks for the amazing unlikely friendship ever since we met. I hope we continue to be terrible influences on each other and let's follow through on our ridiculous future ambitions for more degrees or project deadlines!

Special thanks to Angela, Ingeborg, Lex, Maerten, and Sophie for being a safety net in difficult times and getting me back on my feet. Your warmth and kindness has me forever grateful and humbled.

For the wonderful atmosphere at our institute CWI, I would like to thank all my former colleagues. In particular, I'm immensely thankful for the friendships made along the way: Debarati (Deba), Jesse, Keith, Krzysztof, and Wander. Wander, I always enjoyed how you would put our struggles and endless discussions into perspective, you were a great example. You and Keith's parent-esque guidance over me and Deba was a warm welcome to the institute and continued to be a wonderful environment long after the initial stages. Jesse, thank you for introducing me to the world of oceanography. Being your office mate during my introduction at CWI was a lot of fun. The most fun we had together, in my opinion, was preparing for your PhD defense together with Deba. Krzysztof, I love the discussions we have had about life, work, and everything in between. Our little frog project is still one of the most fun work experiences I have ever had. Keith, you were always hilarious. Your random facts and anecdotes made every lunch a blast. I hope to see you again soon. Deba, I always related most to you. I really value the endless conversations we have had about everything under the sun. There has never been a topic too uncomfortable, although not all of our colleagues agree with that sentiment. Even when we disagree, which happens frequently, we always have fun.

Thanks also go out to my ASML colleagues for welcoming me into a very collaborative work environment that continues to provide engaging intellectual challenges. In particular, a special thanks goes out to Alexandru (Alex), Dmitry, and Remco for your collaboration and friendship over the past few years. Also, thank you for joining efforts in fighting for structural improvements that are both pragmatic and have the right level of integrity. Alex and Remco, thank you for taking me on board some very interesting projects, you have been great mentors and work partners. Alex, you have always been generous in sharing your technical insights and intuitions. I have learned a lot from our collaborations. Remco, thank you for fighting for a way of working that makes it worth it at the end of the day. Also, for daring me “to be the exception”. If you are reading this, then it means it worked out! Dmitry, thank you for the many interesting talks that never fail to also be entertaining as hell. I hope we can have many more occasions where Alex enters the conversation at precisely the most inappropriate moment (seriously Alex, how do you do it?). I hope to share many more late night beers with you all.

Summary

Stochastic Models For Unresolved Scales In Ocean Flows

If you stir the water in a bathtub, the water will quickly be populated by *eddies*: chaotic and unpredictable water whirls. These eddies make the water "rich in texture". That is to say, eddies of all sizes whirl around the water. The broader, deeper eddies whirl across the bathtub (approximately 1 meter in length), and the smallest, fastest eddies can be just 1 millimeter in size.

Now, let us increase the scope of our example to the Earth's oceans, or, in terms of our previous example: a very big bathtub being "stirred" by the wind and density differences related to salinity and temperature. Ocean eddies transport quantities such as heat and momentum across the flow. Similar to the bathtub example, the smallest ocean eddies are also around 1 millimeter in size. But an important difference is that the largest ocean circulations can range up to 10.000 kilometers in size. This makes the ratio between the smallest and largest scales of these ocean circulation features an immense 10 billion. Across this range one can characterize several types of scales. Most important for this thesis are the *mesoscale* eddies (mésos is Greek for "middle"), with sizes in the range 10-100 kilometers. Eddies with sizes of a few kilometers or less are called *submesoscale*. At the other end of the spectrum are the major ocean gyres, which span entire oceans.

Ocean flows are very complex in nature. Therefore, some ocean features are typically approximated in order to model ocean flows. One such approximation results from a choice of resolution so that eddies smaller than some scale are effectively absent from the ocean model. Mesoscale eddies play an important role in choosing such a resolution. While mesoscale eddies are significantly smaller than ocean-spanning phenomena such as the North Atlantic Gyre, they are large enough that they cannot be ignored. In fact, the mesoscales are very important for ocean models to characterize the large scale ocean flows accurately. Unfortunately, their small size and fast-evolving nature makes them too complex

(i.e. computationally demanding) for most ocean models. In this way ocean modeling is part of the field of *multiscale modeling* wherein interactions between features across multiple scales or resolutions are considered.

In this thesis I propose several new mathematical models, called *parameterizations*, to approximate the effects of mesoscale eddies in ocean models. Without parameterizations, these effects would be missing. Much of the work in the field of mesoscale eddy parameterization is focused on deterministic models to represent the mesoscale eddies. In a deterministic relationship a certain sequence of operations transforms a set of variables into another variable with no randomness to the result. That is to say, a deterministic model will always produce the same output for a given input. However, such deterministic models can at best reproduce the *mean effect* of the eddies. Unfortunately, the fluctuations around the mean of the mesoscale effects are instrumental in driving ocean and atmosphere flows. Instead, the work in this thesis is focused on probabilistic/stochastic models for the mesoscale eddies. Stochastic models introduce a degree of randomness that allows, for example, incorporating seemingly random behavior of millions of particles, and filtering of noise from observations.

A starting point for the developed methodologies here is the availability of sample data for the mesoscale eddies. Such data can be obtained in various ways: e.g. observational data from satellite measurements or highly accurate simulated data on short intervals. This sample data is used to inform our stochastic parameterizations with practical examples of the mesoscale eddies without needing to calculate their complex effects.

Chapter 1 gives a brief introduction to several key topics for this thesis. In Chapter 2 we introduce three stochastic methodologies with the purpose of modeling mesoscale effects: an empirical approach that resamples the data based on conditional probabilities, a conditional Markov Chain approach, and an approach using the Ornstein-Uhlenbeck process. Each of these methodologies are tested on the prototype multiscale Kac-Zwanzig heat bath model. All three methodologies are able to reproduce the first four statistical moments of the variable of interest as well as the temporal correlations.

In Chapter 3 we extend the resampling method proposed in Chapter 2 to reproduce the unresolved eddy forcing in modal quasi-geostrophic potential-vorticity equations in double-gyre configuration. The resampling stochastic simulations are able to accurately reproduce all tested statistical and physical criteria of the reference simulation. However, two difficulties arise in developing this methodology further. One is handling the large amount of data, the other is the challenge of reproducing spatial correlations. This motivates the work in the next chapter.

In Chapter 4 we focus on the development of a stochastic parameterization that explicitly reproduces spatial correlations without requiring large computer memory. To this end we introduce a constrained vector autoregressive process that imposes structures on each of its drift coefficient matrices so that they are sparse. This methodology is shown to be effective when tested on the Lorenz '96 model under different configurations.

In Chapter 5 we extend the VARX formulation from Chapter 4 to be fully sparse by using sparse matrices for calculation of its noise term. To this end we compare several state-of-the-art methodologies for estimating sparse approximations of the covariance matrix root or the inverse covariance matrix root. We find the Convex Sparse Cholesky Selection (CSCS) method to be the most effective in introducing sparsity while giving accurate approximations. With the CSCS method, a sparse approximation of the inverse covariance matrix root is estimated directly, without first estimating the covariance or its inverse.

Samenvatting

Stochastische modellen voor onopgeloste schalen in oceaanstromen

Als het water in een badkuip geroerd wordt, dan wordt het water snel gevuld met *eddies*: chaotische en onvoorspelbare waterwervelingen. Deze eddies geven het water een "complexe samenhang". Met andere woorden, eddies van vele groottes wervelen rond in het water. De bredere, diepere eddies wervelen over de lengte van de badkuip (ongeveer 1 meter) en de smallere, snellere eddies kunnen zo klein zijn als 1 millimeter in grootte.

Vervang nu het domein van ons voorbeeld met de oceaanstromen van de Aarde, of, in termen van ons vorig voorbeeld: een heel grote badkuip die "geroerd" wordt door de wind en dichtheidsverschillen gerelateerd aan zoutgehalte en temperatuur. Eddies in de oceaan transporteren hoeveelheden zoals hitte en impuls door de stroming. Net zoals in het voorbeeld van de badkuip, zijn de kleinste eddies in de oceaan ongeveer 1 millimeter groot. Maar een belangrijk verschil is dat de grootste oceancirculaties zo groot kunnen worden als 10.000 kilometer. Hierdoor is de verhouding tussen de kleinste en grootste kenmerken in oceaancirculaties een gigantische 10 miljard. Van klein tot groot kan men meerdere schalen typeren voor eddies. Voor dit proefschrift zijn de *mesoschalige* eddies (mésos is Grieks for "midden") het belangrijkst, deze schalen variëren in grootte van 10 tot 100 kilometer. Eddies met een maximale omvang van een paar kilometer zijn de *submesoschalen*. Aan het andere uiterste bestaan er grote eddies die over gehele oceanen stromen.

Oceanstromen zijn zeer complex, daarom worden eigenschappen van deze stromingen vaak benaderd in oceaanmodellen. Een voorbeeld van zulke benaderingen resulteert direct uit de keuze voor resolutie, hierdoor worden eddies kleiner dan een bepaalde schaal effectief weggelaten. Mesoschaal eddies zijn belangrijk bij het kiezen van een resolutie. Ondanks dat mesoschaal eddies significant kleiner zijn dan oceaanbrede fenomenen zoals de Noord-Atlantische gyre, zijn zij groot genoeg dat zij niet genegeerd kunnen worden. De mesoschalen zijn zeer belangrijk voor een accurate karakterisering van grootschalige oceaan-stromen in oceaanmodellen. Helaas zijn de mesoschalen te complex (dat is, computationaal veeleisend) door hun kleine omvang en snelle bewegingen voor de meeste oceaanmodellen. Op deze manier is oceanmodellering een onderdeel van het veld *multischaal modellering* waarin interacties beschouwd worden tussen meerdere schalen of resoluties.

In dit proefschrift stel ik verscheidene nieuwe mathematische modellen voor, zogeheten *parametrisaties*, om de effecten van mesoschaal eddies te benaderen in oceaanmodellen. Zonder parametrisaties zouden deze effecten ontbreken. Deterministische modellen worden vaak gebruikt om mesoschaal eddies te representeren. Een deterministische relatie transformeert met een reeks operaties een verzameling variabelen naar een andere variabele zonder willekeur in het resultaat. Met andere woorden, een deterministisch model zal altijd dezelfde uitkomst genereren voor een gegeven input. Maar zulke deterministische modellen representeren, op hun best, het *gemiddelde effect* van de eddies. Helaas zijn de fluctuaties om het gemiddelde cruciaal om de oceaan- en atmosfeerstromen aan te sturen. Daarentegen richt het werk in dit proefschrift zich op probabilistische/stochastische modellen om de mesoschaal eddies te karakteriseren. Stochastische modellen introduceren een willekeur die ze in staat stelt om, bijvoorbeeld, het ogenschijnlijk willekeurige gedrag van miljoenen deeltjes te verklaren en observatieruis weg te filteren.

Een startpunt voor de methodologieën die hier ontwikkeld worden is de beschikbaarheid van voorbeelddata voor de mesoschaal eddies. Zulke data kan verkregen worden op verscheidene manieren, bijvoorbeeld: geobserveerde data van satellietmetingen of zeer accurate gesimuleerde data op korte intervallen. Deze voorbeelddata wordt gebruikt om onze stochastische parametrisaties te informeren met praktijkvoorbeelden voor de mesoschaal eddies zonder hun complexe effecten te hoeven berekenen.

Hoofdstuk 1 introduceert kort een aantal hoofdonderwerpen voor dit proefschrift. In hoofdstuk 2 introduceren we drie stochastische methodologieën met het doel de mesoschaal effecten te modelleren: een empirische aanpak die data herbemonsterd gebaseerd op conditionele kansverdelingen, een aanpak van conditionele Markovketens en een aanpak die een Ornstein–Uhlenbeck proces gebruikt. Elk van deze methodologieën zijn getest op het prototype multischaal

Kac–Zwanzig hittebad model. Alle drie de methodologieën zijn in staat de eerste vier statistische momenten en de temporale correlaties van de doelvariabele te herproduceren.

In hoofdstuk 3 breiden wij de herbemonsteringsmethode van hoofdstuk 2 uit om de onopgeloste eddy-aandrijving te herproduceren in modale vergelijkingen voor quasi-geostrofische potentiële vorticiteit in een dubbel-gyre configuratie. De herbemonsteringssimulaties herproduceren accuraat alle geteste statistische en fysische eigenschappen van de referentiesimulatie. Desondanks duiken er twee uitdagingen op om deze methodologie verder te ontwikkelen. De één is het verwerken van grote hoeveelheden data, de ander is het herproduceren van de spatiële correlaties. Dit motiveert het werk in het volgende hoofdstuk.

In hoofdstuk 4 richten wij ons op het ontwikkelen van een stochastische parametrisatie die expliciet spatiële correlaties herproduceert zonder grote hoeveelheden computergeheugen te vereisen. Om dit te bereiken, introduceren wij een ingeperkt vector-autoregressief (VARX) model dat structuren oplegt aan elk van haar driftcoëfficiënt-matrices zodat zij dunbezet zijn. Deze methodologie blijkt effectief voor meerdere configuraties van het Lorenz '96 model.

In hoofdstuk 5 breiden wij de VARX-formulering van hoofdstuk 4 uit om dunbezette matrices te gebruiken om de ruisterm te berekenen. Hiervoor vergelijken wij een aantal van de nieuwste methodologieën om de wortel van de covariantiematrix of de wortel van de inverse covariantiematrix dunbezet te benaderen. Uit deze vergelijkingen blijkt dat de CSCS-methode het meest effectief is om zowel dunbezette als accurate matrices te schatten. De CSCS-methode schat direct een dunbezette benadering van de wortel van de inverse covariantiematrix zonder eerst de covariantiematrix of zijn inverse te benaderen.

Contents

1 Introduction **1**

1 Research question 1

2 Mesoscale ocean features 2

3 Multiscale modeling 3

4 Parameterizations (deterministic and stochastic) 4

5 Data-driven modeling 5

6 Sampling from conditional distributions 6

7 Thesis layout 7

2 Data-driven stochastic representations of unresolved features in multiscale models **9**

1 Introduction 10

1.1 Background and motivation. 10

1.2 Problem description. 12

2 Kac–Zwanzig heat bath: a prototype model 14

2.1 Model description. 14

2.2 Numerical integration schemes. 15

2.3 Approximating conditional distributions by binning. . . . 16
3 Numerical methods 19
3.1 Empirical distribution. 19
3.1.1 Reproducing statistical moments of distinguished particle. 19
3.1.2 Reproducing autocorrelation of distinguished particle. 20
3.2 Conditional Markov chain approach. 21
3.2.1 Definition of the CMC. 22
3.2.2 Numerical results. 24
3.3 Ornstein–Uhlenbeck process. 26
3.3.1 Unconditional parameters. 26
3.3.2 Conditional parameters with binning. 28
3.3.3 Conditional parameters with a linearly fitted mean. 30
3.3.4 Numerical results. 31
4 Discussion 33
Appendix 37
A The equivalence of the exact and standard ML estimators in appropriate limits 37
B Explicit formulas for linearly fitted OU parameters 41

3 Covariate-based stochastic parameterization of baroclinic ocean eddies 43
1 Introduction 44
1.1 Background and motivation 44
1.2 Problem description. 48
2 Physical model 49
2.1 Spatial structure and restriction of the eddy forcing 53
3 Stochastic model 55
3.1 Conditioning procedure 55
3.2 Time-lagged covariate 57
3.3 Flow-dependent covariate 58
3.4 Sampling from empirical distribution 60
3.5 Emulated stochastic eddy forcing 61
4 Results 62

4.1 One-way coupling 62
4.2 Two-way coupling: single conditioning variable 65
4.3 Two-way coupling: double conditioning variables 66
5 Summary and discussion 75
Appendix 79
A Linear fit details 79
B Empirical distribution details 80
C Spatial smoothing 81
D Definitions of diagnostic criteria 82
E Numerical integration details 83

4 Stochastic parameterization with VARX processes 86
1 Introduction 87
1.1 Background 87
2 VARX representation 89
2.1 Mean: linear combination of covariates 89
2.2 Covariance and resulting VARX 91
2.3 Computational complexity 92
2.4 Comparison to other stochastic parameterizations 93
3 Lorenz ’96 model 95
3.1 Model parameter configurations 96
3.2 Stochastic model 96
3.2.1 Order selection - lag time choice 98
4 Practical implementation of stochastic parameterization 100
5 Numerical results 101
5.1 White noise parameterizations 103
5.2 Single regressor parameterizations 104
5.3 Double regressor parameterizations 106
5.3.1 Diagonal covariance 106
5.3.2 Fully dense covariance 108
5.4 NARMAX parameterizations 109
6 Discussion 111

5 A comparison of methods for estimating sparse covariance and precision matrix roots 115
- 1 Introduction . . . 116
  - 1.1 Assumptions, limitations, and notation . . . 117
  - 1.2 Sparse covariance and precision matrix estimation . . . 120
  - 1.3 Graphical lasso . . . 123
  - 1.4 Overview of work . . . 125
- 2 Prototype model to generate data . . . 126
- 3 Sparse matrix estimation methodologies . . . 129
  - 3.1 Sparse covariance estimation . . . 129
    - 3.1.1 Thresholding methods . . . 129
    - 3.1.2 Banding methods . . . 129
    - 3.1.3 Tapering methods . . . 130
  - 3.2 Sparse precision estimation . . . 131
    - 3.2.1 Penalized likelihood methods . . . 132
    - 3.2.2 Column-by-column methods . . . 134
  - 3.3 Sparse precision root estimation . . . 136
    - 3.3.1 CSCS . . . 136
- 4 Numerical results . . . 138
  - 4.1 Practical matters . . . 139
  - 4.2 Sparse covariance results . . . 142
  - 4.3 Sparse precision results . . . 143
  - 4.4 CSCS results . . . 145
    - 4.4.1 $AR(1)$ test . . . 145
    - 4.4.2 Oracle test . . . 149
- 5 Discussion . . . 151

Bibliography 155

List of Publications 171

Chapter 1

Introduction

1 Research question

Mesoscale turbulent eddies strongly influence the large-scale ocean circulation. Global climate models need high spatial resolutions to directly resolve these turbulent motions [74]. However, such high resolutions are infeasible in current climate models due to computational limitations. Therefore, we utilize mathematical models to emulate the effects of the turbulent motions.

Numerical ocean models are based on a system of partial differential equations (PDEs) known as the Navier–Stokes equations. Generally, the Navier–Stokes equations describe motion of viscous fluids. Depending on the context, different forms of approximations of the Navier–Stokes equations are utilized. Formally, this thesis considers coupled systems of ordinary differential equations (ODEs) of the following form:

$$\frac{d\boldsymbol{x}}{dt} = f(\boldsymbol{x}, \boldsymbol{y}), \qquad \frac{d\boldsymbol{y}}{dt} = g(\boldsymbol{x}, \boldsymbol{y}), \tag{1.1}$$

where the system of coupled ODEs (1.1) is obtained by spatial discretization of the PDE for ocean flow. Here, $\boldsymbol{x}$ and $\boldsymbol{y}$ represent oceanic flow variables discretized over a spatial grid. The nonlinear functions f and g govern the temporal evolution of $\boldsymbol{x}$ and $\boldsymbol{y}$, they combine effects of advection, dissipation, forcing, and scale-to-scale interactions.

This class of ocean models (1.1) evolves different flow components $\boldsymbol{x}$ and $\boldsymbol{y}$ over time, e.g. the ocean flow decomposed into its barotropic ($\boldsymbol{x}$) and baroclinic ($\boldsymbol{y}$) components. An important problem that motivates much of the work in this thesis is that the components or features denoted by $\boldsymbol{y}$ are too computationally costly to evaluate and evolve over time. This is motivated by the comparatively small spatial and temporal scales that characterize these features. Therefore, the features $\boldsymbol{y}$ are referred to as the *unresolved* flow features. Of course, this is a simplification of the truth: there is a mix of resolved and unresolved flow features distributed over both $\boldsymbol{x}$ and $\boldsymbol{y}$. Nevertheless, the term "unresolved" gives us a convenient generalization of specific terminology (e.g. baroclinic features).

The approach proposed and studied across this thesis is to develop a stochastic process $\widetilde{\boldsymbol{y}}$ that emulates the dynamical feedback of $\boldsymbol{y}$ on $\boldsymbol{x}$ as described by the following system:

$$\frac{d\widetilde{\boldsymbol{x}}}{dt} = f(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{y}}), \tag{1.2}$$

where $\widetilde{\boldsymbol{x}}$ represents the reduced flow. Ideally, $\widetilde{\boldsymbol{x}}$ would be statistically equivalent to $\boldsymbol{x}$, but the tilde-notation ($\sim$) is used to distinguish the two subtly different representations.

In summary: the main research question answered in this thesis is: *how does one construct a data-driven stochastic process to properly represent the unresolved features in ocean flows.* The suggested approach is to replace the computationally costly or unfeasible features $\boldsymbol{y}$ from the governing system (1.1) with a stochastic process $\widetilde{\boldsymbol{y}}$ as in (1.2). The goal of the suggested approach is then to stochastically drive $\widetilde{\boldsymbol{x}}$ such that the long-term statistical behavior of $\widetilde{\boldsymbol{x}}$ represents that of $\boldsymbol{x}$ accurately.

2 Mesoscale ocean features

This thesis treats the problem that is the modeling of *mesoscale ocean eddies*: small turbulent features in ocean flows. *Mesoscale* refers here to scales that exist between the micro- and macroscales. The mesoscale flow features exist on spatial scales as small as 10 km and up to hundreds of kilometers. These scales fall largely below the grid resolution scale in many ocean models; for example, the average ocean resolution in current state-of-the-art (CMIP6) Earth System models is 58 km [74]. Mesoscale ocean eddies also propagate over very short temporal scales (days or weeks) compared to the evolution of the climate. For example, changes in the Atlantic Meridional Overturning Circulation, an important element of the climate system, take place on timescales of decades and longer. The requirement of both high-resolution temporal and spatial scales make mesoscale ocean flow features computationally very expensive to resolve in detail.

3 Multiscale modeling

Traditional modeling approaches focus on one scale of operation. Macroscale models focus on a “low resolution view” of the modeled quantities, while approximating the interactions between smaller scale features. To obtain models of the interactions between smaller sub-grid features, such macroscale models often have to make a lot of assumptions. However, to justify such assumptions one typically requires extensive physical insights and empirical observations. For complex systems these assumptions may be unfeasible or impractical, or even unproductive, to formulate.

Microscale models, instead, focus on individual processes. However, microscale models scale badly to large spatial or long temporal scales as they explicitly resolve an immense amount of features.

In multiscale modeling a system is described by a mix of models operating on different scales or resolutions. A primary difficulty of multiscale models is that there is often no clear scale separation between “slow” and “fast” features. So while one can rank the scales involved with several features, the domain of possible scales is well covered. Therefore, typical modeling techniques that consider different integration scales of slow and fast features, e.g. averaging or equation-free modeling, cannot be applied.

Research in the past decades [71, 108] has shown that the mesoscale ocean features are instrumental in driving the global ocean flow. Also, the mesoscale ocean features span most of the range between microscales and macroscales.

Therefore, the modeling of mesoscale ocean features can be categorized as a concurrent multiscale modeling problem. That is, the macroscopic model is missing essential information and needs to couple with the mesoscopic model to supply the missing information. However, as explained above, the mesoscale features cannot be resolved over the desired time scales. Therefore, some dynamical term is needed to emulate the sub-grid scale features.

4 Parameterizations (deterministic and stochastic)

Dynamical terms that are used to emulate sub-grid scale features in climate models are typically parametric mathematical models, often refered to as *parameterizations*. Early work in formulating parameterizations for climate models (at least up to the 1990s), was primarily focused on deterministic parameterizations. The most well-known and highly regarded among these are variations on the Gent–McWilliams (GM) parameterization [62]. The GM parameterization has an edge over other parameterization methods because it replicates baroclinic instability in such a way to ensure a global sink in potential energy. In contrast, for example, the small slope approximation in Cox [28] implemented diffusion along density surfaces. However, the small slope approximation needed the inclusion of artificial diffusion to remain stable in long simulations [61, 68]. Recently, research has focused on the application of neural networks for such deterministic parameterizations. For example, Bolton and Zanna [20] focused on the application of convolutional neural networks to predict the oceanic flow fields and showed robustness to some subsampling of the training data.

While deterministic approaches have proven very effective in certain models, they also have obvious limitations when there is a one-to-many relationship from resolved to unresolved states. In a basic sense, most deterministic parameterizations represent the integrated effect of the unresolved scales on the mean state. Such an integrated, or ensemble-mean, representation of the unresolved scales is incapable of representing the one-to-many variability. To overcome these limitations stochastic parameterizations have become popular in recent years [16, 92, 116, 160].

In a stochastic parameterization, the feedback of the unresolved scales is formulated as a stochastic process. In the context of climate models such a stochastic approach aims to emulate the dynamical effect of the mesoscale features on the macroscopic flow. Popular stochastic models in climate models include Markov chains and stochastic cellular automata. These stochastic processes are often made to be conditional on the macroscopic state to allow the process' trajectory to follow along with the trajectory of the macroscopic flow.

5 Data-driven modeling

For the data-driven methods developed in this thesis we rely on the availability of sample data of the ocean flow. One source of data is observations of the real oceans, for example from satellite observations or tracer experiments. Alternatively, from a synthetic perspective, it is possible to use simulation data from a highly resolved ocean model, numerically integrated over a limited domain in space or time. The sample data facilitates a data-driven learning of the stochastic parameterization. In such a data-driven approach, the stochastic process can learn to predict the relations indicated by the data. This thesis describes the development of such data-driven stochastic approaches across several application models.

Consider again the general system of ODEs described by (1.1). In the context of ocean modeling the resolved variables $\boldsymbol{x}$ and the unresolved variables $\boldsymbol{y}$ refer to the macroscopic and mesoscopic flow components, respectively. For now, let us ignore the influence of other model variables on the stochastic process $\widetilde{\boldsymbol{y}}$. If $\widetilde{\boldsymbol{y}}$ is trained on a time series for $\boldsymbol{y}$, then, in an ideal sense, the stochastic variable is sampled from the distribution of $\boldsymbol{y}$, thus $\widetilde{\boldsymbol{y}} \sim \boldsymbol{y}$. In the reality of finite time series, however, the empirical distribution will introduce errors with respect to the true distribution. In the interest of conciseness we will not explicitly separate the empirical distribution $P(\widehat{\boldsymbol{y}})$ and the true distribution $P(\boldsymbol{y})$ unless otherwise specified.

In this thesis, the goal is explicitly not to reproduce temporal trajectories of model variables. Predicting the precise temporal trajectory of variables affected by chaotic processes, such as turbulence, is numerically impossible by definition. However, by employing training on dynamically relevant time series, the goal is to replicate the long-term statistical behavior of such variables.

6 Sampling from conditional distributions

By incorporating the macroscopic flow features in the training, one can use the macroscopic flow as *covariate*, i.e. a predictor variable that correlates with the mesoscale flow features. This requires the predictor variable to be known when the stochastic process needs to be evaluated. This is the case for the macroscale variables $\boldsymbol{x}$ because these are resolved in both offline (training) and online (stochastic variable evaluation in (1.2)) states. If the stochastic process $\widetilde{\boldsymbol{y}}$ is trained including the effects of the macroscales $\boldsymbol{x}$, then, analogously to the previous section, the stochastic variable $\widetilde{\boldsymbol{y}}$ is effectively sampled from the conditional distribution $P(\boldsymbol{y} \,|\, \boldsymbol{x})$, i.e. $\widetilde{\boldsymbol{y}} \sim \boldsymbol{y} \,|\, \boldsymbol{x}$. This informs the stochastic variable $\widetilde{\boldsymbol{y}}$ with the state of the correlating variable $\boldsymbol{x}$. Sampling from this conditional distribution enables the stochastic trajectory to evolve together with the macroscopic flow variables, e.g. in case of transitions between different statistical equilibria. Such an equilibrium shift is extremely difficult to predict explicitly. However, the data-driven training of the conditional stochastic process can learn to recognize such equilibrium shifts under the condition that the time series encompasses different equilibria and that the covariates are sufficiently predictive.

More generally, one can condition the stochastic process on a function of any resolved model variable (covariate). Specific covariates are typically selected with a combination of physical insight and statistical observations. Altogether, covariate selection is a difficult but important task because the choice of covariates can have significant effects on the behavior of the stochastic process. As an illustration, let us compare the conditional distributions that $\widetilde{\boldsymbol{y}}$ is sampled from, in theory (analytically) and in practice (numerically). Theoretically, if some function $\boldsymbol{z}$ of resolved variables was completely independent (orthogonal) to the mesoscale features $\boldsymbol{y}$, then the distribution $\widetilde{\boldsymbol{y}} \sim \boldsymbol{y} \,|\, \boldsymbol{z}$ would be equivalent to $\widetilde{\boldsymbol{y}} \sim \boldsymbol{y}$. However, if, for instance, the probability space is undersampled (which is a realistic concern when dealing with limited data points due to, e.g., memory constraints), then the distribution $\widetilde{\boldsymbol{y}} \sim \boldsymbol{y} \,|\, \boldsymbol{z}$ might have significant bias compared to $\widetilde{\boldsymbol{y}} \sim \boldsymbol{y}$. This is one example of the difficulties that finite sample data introduces in approximating process distributions. Therefore, an important element of this thesis is covariate selection from both physical and statistical perspectives.

7 Thesis layout

The following chapters focus on these aspects:

- In the second chapter we propose and compare three different stochastic approaches to modeling coupled phenomena in the Kac–Zwanzig toy heat-bath model. The main novel contributions of this chapter are the proposed conditional methodologies that scale well to complex ocean models with high numbers of degrees of freedom. Specifically, we propose three categories of parameterizations: the *empirical*, Markov chain, and Ornstein–Uhlenbeck (OU) processes. Another main contribution is the construction of robust OU estimators that do not incur the finite time-step numerical errors typical to traditional estimators.

- The main contribution of the third chapter is showing effective application of the parameterization by sampling from the empirical distribution, as presented in the second chapter, to a wind-driven ocean model of baroclinic instability. Herein it is shown that the empirical process is able to drive the large scale flow dynamics in a strongly reduced ocean model. All tested statistical and physical properties of the flow are reproduced accurately with respect to reference simulations.

- In the fourth chapter we propose a different category of stochastic approaches in the prototype Lorenz '96 model (L96). The main novel contribution of this work is the successful deployment of vector autoregressive processes with exogenous variables (VARX), conditioned on sample data, using only sparse matrices for the drift components. Through this imposed sparsity, the VARX parameterization can incorporate spatial and temporal correlations in a scalable way.

- The fifth chapter answers an important open question from the fourth chapter. Namely, how to simulate the noise components of the VARX using only sparse matrices. For this purpose, this chapter presents and compares several state-of-the-art methodologies that estimate sparse (inverse) covariance matrix roots. The main contribution of the fifth chapter is to select the best performing method to estimate a sparse inverse covariance matrix root suitable to be used for parameterization in numerical ocean models.

While each following chapter builds upon previous chapters, the individual chapters are written to be stand-alone, and can therefore be read as such.

Chapter 2

Data-driven stochastic representations of unresolved features in multiscale models

In this study we investigate how to use sample data, generated by a fully resolved multiscale model, to construct stochastic representations of unresolved scales in reduced models. We explore three methods to model these stochastic representations. They employ empirical distributions, conditional Markov chains and conditioned Ornstein–Uhlenbeck processes, respectively. The Kac–Zwanzig heat bath model is used as a prototype model to illustrate the methods. We demonstrate that all tested strategies reproduce the dynamics of the resolved model variables accurately. Furthermore, we show that the computational cost of the reduced model is several orders of magnitude lower than that of the fully resolved model.

1 Introduction

1.1 Background and motivation.

Multiscale modeling is an active research topic in such fields as biomedical engineering, materials science and climate modeling. The common property of multiscale problems is the occurrence of a wide range of spatial and/or temporal scales, often resulting in an inability of numerical simulations to accurately resolve the small and/or fast scales. However, processes at these scales can be instrumental in driving the large scale processes, hence they must be represented in a simplified yet accurate manner in numerical models.

The motivation for this study comes primarily from atmosphere-ocean science, where the problem of formulating suitable representations of unresolved processes is well-known. In the field of atmosphere-ocean modeling, such representations are known under the name *parameterizations*. In this field, early developments on multiscale problems used deterministic methods to represent the effect of unresolved processes. However, although deterministic methods can reproduce the mean effect of the unresolved processes conditioned on the resolved variables, they lack the ability to reproduce the fluctuations around this mean. Recent work has focused on overcoming this limitation by using stochastic methods to model this noise-like behavior, particularly in atmospheric context [55, 69, 104, 116, 152]. Notable examples for the present study include [30] and [39], which propose data-inferred conditional Markov chains to represent atmospheric convection in coarse climate models. Recently, stochastic parameterizations have also started to receive attention in oceanic research, e.g. [13, 14] and [117], which investigate stochastic eddy-forcing in ocean currents.

In this study we investigate data-driven stochastic methods to drive *reduced* multiscale models. In atmosphere-ocean modeling, there are many scales but no strong scale separation (or scale gap), so that techniques that rely on such a scale gap to achieve computational efficiency gains (e.g. averaging, equation-free modeling [86], heterogeneous multiscale methods [44]) are less attractive. A data-driven approach can be an interesting alternative in such cases. The idea of such an approach is to infer a suitable stochastic process from data (time series) of the feedback from the small/fast scales, and to couple this process to a reduced model for the large/slow scales. The statistical inference step is performed off-line, i.e. the stochastic process for the unresolved scales is precomputed. Thus, it can be considered a "sequential coupling" method [44]. As we will demonstrate, the computational gain of this data-driven methodology can be very substantial.

We emphasize that the methodology studied here is different from inferring a stochastic process for the large scale dynamics itself. Rather, it is aimed at situations where an available but incomplete model for the large scale dynamics needs to be augmented with a model for small scale feedbacks (as is the case in e.g. atmosphere-ocean modeling). In general, a suitable stochastic model for the small scale feedbacks must be dependent (conditioned) on the state of the large scale degrees of freedom. The statistical inference step for such a conditioned stochastic process is not straightforward. We approach this issue by considering the large scale state as a covariate for the stochastic process that needs to be inferred.

The data-driven methodology studied in this paper builds on the work presented in [30]. There, finite-state Markov chains were used to model feedback from unresolved scales in the context of the Lorenz '96 model. This conditional Markov chain approach gave good results but involved the estimation of many parameters. Furthermore, in [30] no experiments were performed with different sets of conditioning variables (or covariates). In the current study we explore methods that require far less parameters to be estimated (or even none at all). For completeness, a method that stays close to [30] is included in this exploration. We also investigate the effect that varying the set of conditioning variables has on the resulting reduced model.

In the remainder of the introduction we formally pose the discussed problem and the questions this work attempts to answer. Section 2 describes the prototype multiscale model and details on its numerical implementation. Section 3 presents the three different strategies used to fit the stochastic process to the sample data: the empirical, conditional Markov chain and Ornstein–Uhlenbeck approaches, respectively. Lastly, the results, and their implications for future work, are discussed in section 4.

1.2 Problem description.

Given a stationary time series $\boldsymbol{X} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_M)$, for $\boldsymbol{x}_i \in \mathbb{R}^d$, we wish to formulate a model such that when we integrate this model numerically, we generate a time series $\widetilde{\boldsymbol{X}} = (\widetilde{\boldsymbol{x}}_0, \widetilde{\boldsymbol{x}}_1, \ldots, \widetilde{\boldsymbol{x}}_N)$, for $\widetilde{\boldsymbol{x}}_i \in \mathbb{R}^d$, whose statistics accurately resemble those of $\boldsymbol{X}$. Throughout this paper we compare given data sets, where variables are denoted normally (e.g. $\boldsymbol{x}$), with data sets, denoted with a tilde (e.g. $\widetilde{\boldsymbol{x}}$), generated by reduced models.

For the stochastic approach discussed here we assume that the given sample data consists of both $\boldsymbol{X}$ and $\boldsymbol{R}$, where $\boldsymbol{R}$ represents small-scale features. As an example, one can think of fluid flow, with $\boldsymbol{X}$ and $\boldsymbol{R}$ time series of the resolved-scale flow and the subgrid-scale stress term, respectively. Let $\widetilde{\boldsymbol{X}}$ be generated by a reduced model g together with a stochastic process $\widetilde{\boldsymbol{R}} = (\widetilde{\boldsymbol{r}}_0, \widetilde{\boldsymbol{r}}_1, \ldots, \widetilde{\boldsymbol{r}}_N)$, for $\widetilde{\boldsymbol{r}}_i \in \mathbb{R}^d$, that is fitted to $\boldsymbol{R}$. This construction describes the class of systems:

$$\dot{\widetilde{\boldsymbol{x}}} = g(\widetilde{\boldsymbol{x}}) + \widetilde{\boldsymbol{r}}, \ \dot{\widetilde{\boldsymbol{r}}} = h(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{r}}), \tag{2.1}$$

where $\dot{\widetilde{\boldsymbol{x}}}$ denotes the temporal derivative of $\widetilde{\boldsymbol{x}}$ (and analogously for $\dot{\widetilde{\boldsymbol{r}}}$). This class of systems finds practical applications in, e.g, modeling the eddy forcing term with $\widetilde{\boldsymbol{r}}$ in ocean flow models [13], and was the inspiration for this work.

Note that we assume analytic solutions to the discussed problem to be unknown. Therefore, we will make use of numerical integration schemes. Let us introduce the following notations: $t_i = i\Delta t$, $\boldsymbol{x}_i = \boldsymbol{x}(t_i)$ denotes the $(i+1)$-th entry in the time series $\boldsymbol{X}$, and $\Delta\boldsymbol{x}_i = \boldsymbol{x}_{i+1} - \boldsymbol{x}_i$.

Although we have no rigorous proof, we expect the statistics of $\boldsymbol{X}$ to be accurately emulated by $\widetilde{\boldsymbol{X}}$ if it were possible to sample $\widetilde{\boldsymbol{r}}_{i+1} = \widetilde{\boldsymbol{r}}(t_{i+1})$ from the conditional distribution of $r_{i+1} \mid (\boldsymbol{x}_i = \widetilde{\boldsymbol{x}}_i, \ldots, \boldsymbol{x}_0 = \widetilde{\boldsymbol{x}}_0, \boldsymbol{r}_i = \widetilde{\boldsymbol{r}}_i, \ldots, \boldsymbol{r}_0 = \widetilde{\boldsymbol{r}}_0)$. In general, however, such distributions are not known exactly, and the size of sample data needed to accurately approximate conditional distributions increases drastically with the number of conditions. Therefore, we investigate how well the statistics of $\widetilde{\boldsymbol{X}}$ approximate those of $\boldsymbol{X}$ when conditioning $\widetilde{\boldsymbol{r}}_{i+1}$ on a selection of past values of $\boldsymbol{x}$ and $\boldsymbol{r}$. The approximation quality of $\widetilde{\boldsymbol{X}}$ is measured by the degree to which specific sample moments and autocorrelations of $\boldsymbol{X}$ are captured by $\widetilde{\boldsymbol{X}}$.

Formally, let $\widetilde{\boldsymbol{r}}_{i+1}$ be sampled from the distribution of

$$r_{i+1} \mid (\boldsymbol{x}_i = \widetilde{\boldsymbol{x}}_i, \ldots, \boldsymbol{x}_{i-i'} = \widetilde{\boldsymbol{x}}_{i-i'}, \boldsymbol{r}_i = \widetilde{\boldsymbol{r}}_i, \ldots, \boldsymbol{r}_{i-i''} = \widetilde{\boldsymbol{r}}_{i-i''})$$

, with $0 \leq i', i'' \leq i$, and consider the following questions:

- Let the sample mean and standard deviation of $\boldsymbol{X}$ be denoted by $\gamma_1(\boldsymbol{X}) = \mathbf{E}(\boldsymbol{x}_i)$ and $\gamma_2(\boldsymbol{X}) = (\mathbf{E}(\boldsymbol{x}_i^2) - \mathbf{E}(\boldsymbol{x}_i)^2)^{1/2}$, respectively (with $\mathbf{E}$ denoting expectation). Let the s-th sample moment of $\boldsymbol{X}$ (with $s \geq 3$) be given by:

  $$\gamma_s(\boldsymbol{X}) = \mathbf{E}\left[(\boldsymbol{x}_i - \mathbf{E}(\boldsymbol{x}_i))^s\right] (\mathbb{V}\mathrm{ar}(\boldsymbol{x}_i))^{-s/2}.$$

  Let $\epsilon(\gamma_s) := \gamma_s(\boldsymbol{X}) - \gamma_s(\widetilde{\boldsymbol{X}})$ be the error of the s-th sample moment as reproduced by $\widetilde{\boldsymbol{X}}$, and let S be the maximum moment one aims to reproduce. How does $\epsilon(\gamma_s)$ depend on the number of past values of $\boldsymbol{x}$ and $\boldsymbol{r}$ conditioning $\boldsymbol{r}_{i+1}$, i.e. how does $\epsilon(\gamma_s)$ depend on i' and i''? Particularly, let E denote a maximum error one is willing to permit, for what i' and i'' does $\epsilon(\gamma_s) \leq E$ hold for $1 \leq s \leq S$?

- Let the autocorrelation function of $\boldsymbol{X}$ with lag l be given by:

  $$\mathrm{ACF}_l(\boldsymbol{X}) = \mathbf{E}\left[(\boldsymbol{x}_i - \mathbf{E}(\boldsymbol{x}_i))(\boldsymbol{x}_{i+l} - \mathbf{E}(\boldsymbol{x}_i))\right] (\mathbb{V}\mathrm{ar}(\boldsymbol{x}_i))^{-1}.$$

  Let $\epsilon(\mathrm{ACF}_l) := \mathrm{ACF}_l(\boldsymbol{X}) - \mathrm{ACF}_l(\widetilde{\boldsymbol{X}})$ be the error of the autocorrelation with lag l as reproduced by $\widetilde{\boldsymbol{X}}$, and let L be the maximum correlation lag time one aims to reproduce. How does $\epsilon(\mathrm{ACF}_l)$ depend on i' and i''? Particularly, let E' denote a maximum error one is willing to permit, for what i' and i'' does $\epsilon(\mathrm{ACF}_l) \leq E'$ hold for $0 \leq l \leq L$?

Rather than dealing with the technical intricacies and complications of testing methodologies directly on highly complex multiscale models, we elect to test our ideas on the simpler and more accessible Kac–Zwanzig *heat bath model* [53, 162]. This model, described below, also belongs to the class of systems in (2.1).

Assume a resolved heat bath model's sample data, $(\boldsymbol{X}, \boldsymbol{R}) = (\boldsymbol{Q}, \boldsymbol{P}, \boldsymbol{R})$, where $\boldsymbol{Q} = (q_0, q_1, \ldots, q_M)$, $\boldsymbol{P} = (p_0, p_1, \ldots, p_M)$, and $\boldsymbol{R} = (r_0, r_1, \ldots, r_M)$, for $q_i, p_i, r_i \in \mathbb{R}$, is given. The question we attempt to answer here is: how can we fit a stochastic process $\widetilde{\boldsymbol{R}}$ to $\boldsymbol{R}$ in such a way that the reduced model variables' time series, $\widetilde{\boldsymbol{Q}}$ and $\widetilde{\boldsymbol{P}}$, reproduce the statistics of $\boldsymbol{Q}$ and $\boldsymbol{P}$, respectively? With respect to this heat bath model, a thorough theoretical analysis of the questions asked in this section eludes us. Therefore, we approach these questions from a numerical perspective.

2 Kac–Zwanzig heat bath: a prototype model

2.1 Model description.

In the heat bath model, one considers the temporal evolution of a distinguished particle, moving in a potential V and coupled to J heat bath particles. The distinguished particle has unit mass, position q and momentum p. We use the set-up from [138], with a double-well potential $V(q) = 1/4(q^2 - 1)^2$ and linear coupling of the heat bath particles to the distinguished particle. The heat bath particles are oscillators, each with their own momentum u_j, position v_j, mass χ_j and stiffness ξ_j, with $1 \leq j \leq J$. Following [138], let us define the oscillators' natural frequency through $\omega_j^2 = \xi_j/\chi_j$, and choose the oscillator mass $\chi_j = G^2/j^2$ and stiffness $\xi_j = G^2$. The considered heat bath model's Hamiltonian system is then given by the following ordinary differential equations (ODEs):

$$\dot{q} = p, \quad \dot{p} = -V'(q) + G^2(r - Jq), \quad \dot{u_j} = v_j, \quad \dot{v_j} = -j^2(u_j - q), \tag{2.2}$$

where $V'(q) = dV(q)/dq$ and $r(t) := \sum_{j=1}^{J} u_j(t)$. While these ODEs can be solved numerically, the computational cost of evolving p and, more importantly, every u_j and v_j over time will significantly slow down any numerical solver. Therefore, to decrease the required computational work, we introduce a stochastic process $\widetilde{\boldsymbol{R}}$ that approximates the dynamical effect of $\boldsymbol{R}$. Writing r_m for $\sum_j u_j(t_m)$, we have

$$\boldsymbol{R} = (r_0, r_1, \ldots, r_M).$$

By using $\widetilde{\boldsymbol{R}}$ instead of $\boldsymbol{R}$, the heat bath particles (i.e., u_j and v_j) no longer need to be evolved, thus reducing the full system in (2.2) to:

$$\dot{\widetilde{q}} = \widetilde{p}, \quad \dot{\widetilde{p}} = -V'(\widetilde{q}) + G^2(\widetilde{r} - J\widetilde{q}), \quad \dot{\widetilde{r}} = h(\widetilde{q}, \widetilde{p}, \widetilde{r}), \tag{2.3}$$

where the function h that evolves $\widetilde{r}$ over time is yet to be defined.

As mentioned in 1.2, this construction is meant to provide our strategies with a test bed that naturally extends to geophysical fluid flow models. With this in mind, let us motivate our choice for the heat bath model. First, the heat bath particles span a great variety of time scales without a scale gap (because the natural frequencies range from $O(1)$ to $O(J)$), similar to the range of time scales in ocean flow models (as mentioned in 1.1). Also, the reduced heat bath (2.3) and reduced ocean flow models [13] belong to the same class of systems (2.1), in the sense that the stochastic term $\widetilde{r}$ enters in an additive fashion (i.e. $\widetilde{r}$ is added linearly to the ODE for $\widetilde{\boldsymbol{x}}$, there is no multiplication with a function of $\widetilde{\boldsymbol{x}}$). These reasons, together with its technical simplicity, make the heat bath model a suitable choice for our experiments. We remark that we do not attempt to preserve the Hamiltonian structure or the conserved quantities of (2.2) in the reduced model, as this is less relevant for applications in geophysical fluid flow. Furthermore, we do not consider the limit $J \to \infty$, as is done in e.g. [138], rather we keep J fixed at a finite value.

2.2 Numerical integration schemes.

System (2.2) is integrated in time using the symplectic Euler method, which correctly resolves the distinguished particle's motion under the condition $\omega_j \Delta t = O(1)$ [138]. Table 2.1 shows all model parameter settings used for the simulations in this paper. The discretized integration scheme for (2.2) equals:

$$\begin{aligned} p_{i+1} &= p_i - \Delta t\ V'(q_i) + \Delta t\ G^2(r_i - Jq_i), & v_{i+1,j} &= v_{i,j} - \Delta t\ j^2(u_{i,j} - q_i), \\ q_{i+1} &= q_i + \Delta t\ p_{i+1}, & u_{i+1,j} &= u_{i,j} + \Delta t\ v_{i+1,j}. \end{aligned}$$

Let $\mathcal{N}(x, y^2)$ denote a normal distribution with mean x and variance y^2; the harmonic oscillators are initialized by $v_j(0) = 0$ and $u_j(0) \sim \mathcal{N}(0, 1/(\beta k_j))$. The distinguished particle is initialized at $q_0 = 1$ and $p_0 = 0$.

Because of the chosen values for ω_j and the condition $\omega_j \Delta t = O(1)$, one sees that $J\Delta t = O(1)$ must also hold. This means that Δt must decrease as J increases for the symplectic integration scheme to properly resolve all the heat bath particle's scales. Since u_j and v_j are not evolved in the reduced model, the integration time step of a reduced simulation can generally be chosen to be much larger. Therefore, we make a distinction between Δt and $\Delta\tau$ to refer to the time steps of the resolved and reduced model, respectively. Furthermore, the resolved time series is stored with a sampling interval δt ($\geq \Delta t$), see Table 2.1. Recall from section 1.2 that, throughout this paper, we use the notation $\widetilde{q}$ to refer to a variable in the reduced model that is the counterpart of the variable q in the fully resolved model. Discretizing (2.3) results in the following integration scheme for the reduced model:

$$\begin{aligned} \widetilde{p}_{i+1} &= \widetilde{p}_i - \Delta\tau\ V'(\widetilde{q}_i) + \Delta\tau\ G^2(\widetilde{r}_i - J\widetilde{q}_i), \\ \widetilde{q}_{i+1} &= \widetilde{q}_i + \Delta\tau\ \widetilde{p}_{i+1}, \\ \widetilde{r}_{i+1} &= \widetilde{r}_i + \Delta\tau\ h(\widetilde{q}_i, \widetilde{p}_i, \widetilde{r}_i), \end{aligned} \tag{2.4}$$

where the initial conditions are chosen to be $\widetilde{p}_0 = p_0$, $\widetilde{q}_0 = q_0$ and $\widetilde{r}_0 = r_0$. The function h in (2.4) is not known analytically, but will be inferred from the data $(\boldsymbol{Q}, \boldsymbol{P}, \boldsymbol{R})$. The different stochastic methods proposed here all aim to model $\widetilde{\boldsymbol{R}}$ in such a way that $\widetilde{\boldsymbol{Q}}$ and $\widetilde{\boldsymbol{P}}$ together with $\widetilde{\boldsymbol{R}}$ reproduce the statistics of $\boldsymbol{Q}$ and $\boldsymbol{P}$. In the next section we discuss the binning procedure used in our methods.

Table 2.1: Heat bath model parameters

	Parameter	Resolved model	Reduced model
G^2	mass and stiffness scaling	1	1
β	inverse temperature	10^{-4}	–
J	number of harmonic oscillators	10^2	–
M	number of sample points	10^7	10^7
δt	sampling interval	10^{-2}	10^{-2}
Δt	integration time step resolved model	10^{-4}	–
$\Delta\tau$	integration time step reduced model	–	10^{-2}
N_B	number of bins per continuous conditioning variable	10	10

2.3 Approximating conditional distributions by binning.

In the reduced model (2.4), $\boldsymbol{R}$ is approximated with the random process $\widetilde{\boldsymbol{R}}$. The strategies discussed in this paper sample $\widetilde{r}$ from the distribution of r conditioned on a set of resolved model variables $c := c(q, p, r)$:

$$\widetilde{r}_{i+1} \sim r_{i+1} \mid (c_i = \widetilde{c}_i)\,. \tag{2.5}$$

A simple example is $c_i = \{r_i\}$; in this case $\widetilde{r}_{i+1}$ is a time-correlated stochastic process. In this work, we consider different methods of approximating the distribution $r_{i+1} \mid (c_i = \widetilde{c}_i)$, or $r_{i+1} \mid c_i$ for short, because the exact distribution is usually unknown. The majority of these methods approximate this distribution using a binning procedure, as explained further below.

Let us consider a set of conditioning variables c_i with cardinality $C + D$, where C and D are the number of *continuous* and *discrete* conditioning variables, respectively. The discrete variables only apply to the CMC approach, and are discussed in section 3.2 (in other sections $D = 0$ holds). The range between the minimum and maximum of each continuous conditioning variable is then independently partitioned in N_B equidistant intervals. This partitioning results in C-dimensional disjoint bins α_b, where $1 \leq b \leq B := (N_B)^C$. Each of these bins describes a set of r_{i+1}-values ρ_b, also with $1 \leq b \leq B$. This procedure is illustrated in Figure 2.1 for the case $c_i = \{q_i\}$ in (2.5). This figure shows that through discretizing the q_i-domain, one finds a mapping from intervals over q_i to sets of r_{i+1}-values.

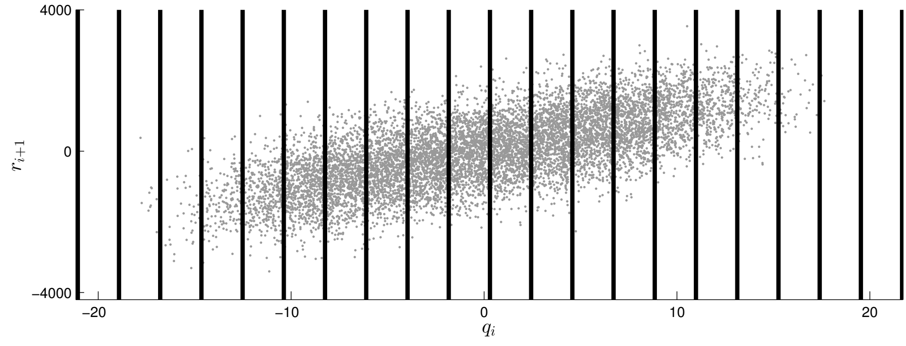


Figure 2.1: An equidistant partitioning of the range of q in 20 bins.

The major advantage of the equidistant binning strategy is its simplicity in both concept and implementation. A caveat is that bins are not guaranteed to contain sample points, in fact, bins are frequently empty in higher dimensional discretizations. One could extensively investigate strategies that describe how to handle these occurrences, however, this is beyond the scope of the current study. Here we simply let empty bins be described by the closest, in Euclidean sense, nonempty bin. In the occurrence of multiple closest bins, our implementation chooses the first closest bin listed in the storage format of the data set. While this is an *ad hoc* choice, we stress that with our chosen sample size M and bin size N_B (see Table 2.1), this is an extremely rare occurrence. This did not occur at all in most of our experiments; in the worst case ($C = 4$, see section 3.3) it affected only 0.01% of the reduced model time steps. However, this could be a point of improvement in future work.

In Figure 2.2, we show the simple algorithm used to integrate the reduced heat bath model (2.3) over time. In the following sections we discuss the stochastic methods that describe the temporal evolution of $\widetilde{r}$.

```
input:   $\boldsymbol{Q}$              : vector of sample data for $q$, length $M$.
         $\boldsymbol{P}$              : vector of sample data for $p$, length $M$.
         $\boldsymbol{R}$              : vector of sample data for $r$, length $M$.
         $c_i$             : set of conditioning variables, size $C$.
         $\alpha_b$             : $C$-dimensional bins, for all $1 \leq b \leq B$.
         $\min(\alpha_b)$       : vector of minimum values per dimension over all $\alpha_b$,
                              length $C$.
         $\mathrm{step}(\alpha_b)$      : vector of bin size per dimension, length $C$.
         method          : the stochastic approach used to approximate $\widetilde{r}$,
                              options: empirical, CMC, bin-wise OU,
                              and linear OU.

$(\widetilde{q}_0, \widetilde{p}_0, \widetilde{r}_0) = (q_0, p_0, r_0)$
$i = 0$
for $i := 0$ to $N - 1$ do
    /* Update $\widetilde{q}$ and $\widetilde{p}$ */
    $\widetilde{p}_{i+1} = \widetilde{p}_i - \Delta\tau\ V'(\widetilde{q}_i) + \Delta\tau\ G^2(\widetilde{r}_i - J\widetilde{q}_i)$
    $\widetilde{q}_{i+1} = \widetilde{q}_i - \Delta\tau\ \widetilde{p}_{i+1}$

    /* Find the bin number $b$ such that $\widetilde{c}_i \in \alpha_b$ */
    $b = \lceil \widetilde{c}_i - \min(\alpha_b) \rceil ./ \mathrm{step}(\alpha_b)$

    /* Update $\widetilde{r}$ by random sampling */
    $\widetilde{r}_{i+1} \sim \mathrm{distr}(\mathrm{method}, b)$
endfor
```

Figure 2.2: Algorithm for the time integration of the reduced model for a given set of conditioning variables c and stochastic approach.

3 Numerical methods

3.1 Empirical distribution.

In this section we discuss the method of sampling $\widetilde{r}$ directly from the sample data's *empirical* distribution, as formally defined in (2.6). This strategy has an obvious limitation in that it can only sample from the values of r observed in the fully resolved simulation. However, for a stationary process, this empirical distribution of r conditioned on past values (see section 1.2) will converge to the exact joint distribution in the limit of infinite data. Basic experiments show that simulations sampling instead from an unconditioned empirical distribution are highly unstable.

3.1.1 Reproducing statistical moments of distinguished particle.

Let us define $\mathcal{U}(\rho_b)$ to denote the uniform distribution on the discrete set ρ_b, i.e. if $U \sim \mathcal{U}(\rho_b)$ then U has equal probability of being any element of the set ρ_b. The empirical approach fits the conditional residual term $\widetilde{r}$ to r as follows:

$$\widetilde{r}_{i+1} \sim \mathcal{U}(\rho_b), \quad \text{where } b : \widetilde{c}_i \in \alpha_b. \tag{2.6}$$

Since q_i and r_{i+1} show a strong correlation, let us consider sampling $\widetilde{r}_{i+1}$ from the distribution of $r_{i+1} \mid q_i$. We integrate the reduced model by using the algorithm in Figure 2.2 and (2.6) with $c_i = \{q_i\}$, and compare the resulting distributions of $\widetilde{p}$ and $\widetilde{q}$ to those of the fully resolved p and q. Each of the distributions is plotted in Figure 2.3.

Figure 2.3 shows that sampling from the distribution in (2.6) is effective in that the general shape of the distributions is reproduced, but there is also clearly room for improvement, e.g. one notices an underestimated standard deviation for both $\widetilde{q}$ and $\widetilde{p}$. As suggested in section 1.2, one expects better results when expanding the set of conditioning variables c_i. Therefore, let us compare the previous approach to the conditioned distribution of $r_{i+1} \mid q_i, r_i$. To clearly illustrate the differences, we plot the absolute error of the resulting distributions in Figure 2.4.

Figure 2.4 shows that the distributions of $\widetilde{p}$ and $\widetilde{q}$ for $c_{i,1} := \{q_i\}$ are improved upon greatly by $c_{i,2} := \{q_i, r_i\}$. As suggested in section 1.2, the first

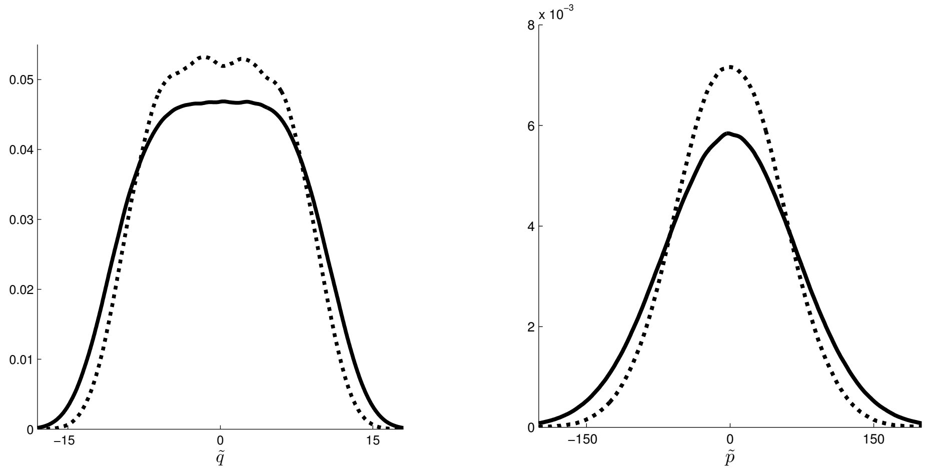


Figure 2.3: The distributions for positions $q, \widetilde{q}$ (left) and momenta $p, \widetilde{p}$ (right). The conditioned empirical distributions approximate sampling from $r_{i+1} \mid q_i$. A comparison between the distributions resulting from the reduced model (dotted lines) and resolved model (solid lines).

four sample moments of q and p, along with those of $\widetilde{q}$ and $\widetilde{p}$ for several cases are compared in Table 2.2. From this table one can conclude that conditioning on $c_{i,2}$ provides an overall improvement to $c_{i,1}$, the major improvement being the accuracy of the standard deviation for both $\widetilde{q}$ and $\widetilde{p}$, but also the kurtosis is more accurately reproduced. Since both q_i and r_i show a clear correlation with r_{i+1}, these results are expected. However, neither of the conditioning parameters improves the temporal correlation, as both condition on the same time step i. This is clearly shown in the autocorrelation functions plotted in Figure 2.5, where both of the approximations produce an inaccurate autocorrelation function. Because these procedures condition on specific time steps, the autocorrelation functions are dependent on the size of $\Delta\tau$, the integration time step of the reduced simulation; simulations discussed here use the parameter values as shown in Table 2.1.

3.1.2 Reproducing autocorrelation of distinguished particle.

Our strategy for improving the autocorrelation function is to build more temporal correlation into the conditioning, i.e. we condition r_{i+1} on system variables from previous time steps. As comparison to the results in section 3.1.1, let us

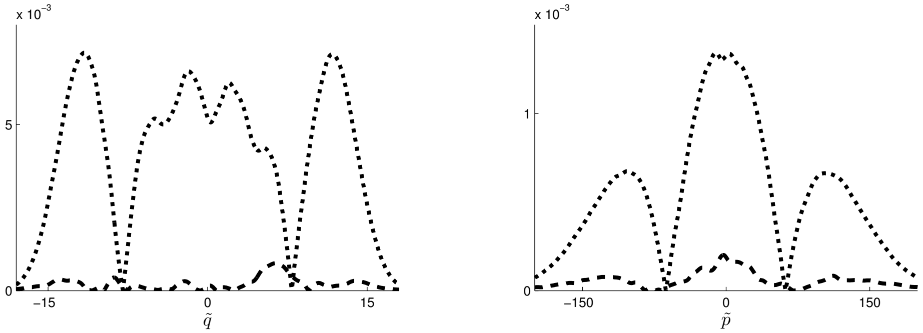


Figure 2.4: Absolute errors of the distributions for positions (left) and momenta (right). The conditioned empirical distributions approximate sampling from $r_{i+1} \mid c_i$. The absolute errors of both $c_{i,1} = \{q_i\}$ (dotted) and $c_{i,2} = \{q_i, r_i\}$ (dashed) are plotted.

Table 2.2: Sample moments for empirical approximations

$\boldsymbol{x}_i$	mean $\gamma_1(\boldsymbol{x}_i)$	std.dev. $\gamma_2(\boldsymbol{x}_i)$	skewness $\gamma_3(\boldsymbol{x}_i)$	kurtosis $\gamma_4(\boldsymbol{x}_i)$
p_i (reference)	0.00	68.4	$3.7 \cdot 10^{-4}$	3.00
$\widetilde{p}_i$ ($c_{i,1} = \{q_i\}$)	0.00	54.2	$-8.6 \cdot 10^{-4}$	2.96
$\widetilde{p}_i$ ($c_{i,2} = \{q_i, r_i\}$)	0.00	70.2	$-1.8 \cdot 10^{-3}$	3.00
$\widetilde{p}_i$ ($c_{i,3} = \{q_i, r_i, r_{i-1}\}$)	0.00	68.6	$1.5 \cdot 10^{-4}$	3.02
q_i (reference)	0.01	6.83	$-5.5 \cdot 10^{-3}$	2.18
$\widetilde{q}_i$ ($c_{i,1} = \{q_i\}$)	0.00	6.04	$-0.3 \cdot 10^{-3}$	2.16
$\widetilde{q}_i$ ($c_{i,2} = \{q_i, r_i\}$)	-0.01	6.86	$-0.5 \cdot 10^{-3}$	2.19
$\widetilde{q}_i$ ($c_{i,3} = \{q_i, r_i, r_{i-1}\}$)	0.02	6.78	$-4.8 \cdot 10^{-3}$	2.19

sample $\widetilde{r}_{i+1}$ from the distribution of $r_{i+1} \mid c_{i,3}$, with $c_{i,3} = \{q_i, r_i, r_{i-1}\}$. Both the probability distributions of the approximated $\widetilde{p}$ and $\widetilde{q}$, as well as the associated autocorrelation functions are shown in Figure 2.6. As can be seen, they resemble the distributions and autocorrelations of the fully resolved model very closely. One can conclude that adding a greater dependency on the history of the sample data is greatly beneficial for approximating the autocorrelation function. Also, the sample moments of the reduced model variables remain comparable in quality (for $\widetilde{q}$) or even improve (for $\widetilde{p}$), see Table 2.2.

3.2 Conditional Markov chain approach.

A natural evolution from the empirical approach, as described in Section 3.1, is to attempt to fit a continuous stochastic process to the sample data of r. Especially in situations where one can not be convinced that the sample data is

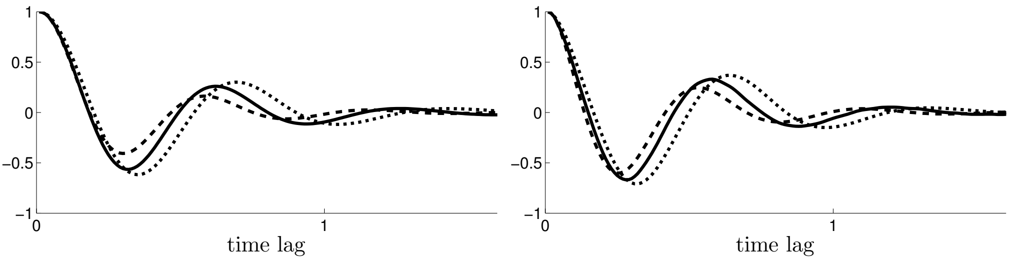


Figure 2.5: Autocorrelation functions for positions (left) and momenta (right). The conditioned empirical distributions approximate sampling from $r_{i+1} \mid c_i$. The autocorrelations for both $c_{i,1} = \{q_i\}$ (dotted lines) and $c_{i,2} = \{q_i, r_i\}$ (dashed lines) are plotted against the resolved autocorrelations (solid lines).

sufficiently representative of the entire range of possible values of the data, the empirical approach will likely not perform to specification, because the empirical distribution samples exclusively from previously observed discrete values. In this section we discuss how to use conditional Markov chains (CMCs) to model the stochastic process, similar (but not identical) to the approach from [30] and [39] (see also [109]).

3.2.1 Definition of the CMC.

Expanding on the ideas put forward in [30], we define a CMC in which $\widetilde{r}$ switches randomly between K deterministic functions f_k, with $1 \leq k \leq K$. These functions describe the strong correlation between q and r: $r_i = f_{k_i}(q_i)$, where $k_i = k(t_i)$ denotes the index of the specific function f in the i-th time step. Important here is that this method constructs $\widetilde{r}$ as a piece-wise (in time) deterministic variable, therefore, one approximates transition distributions for $k_{i+1} \mid c_i$ rather than distributions of the form $r_{i+1} \mid c_i$. The numerical integration steps for a reduced model driven by a CMC residual term are defined as:

$$\begin{aligned} \widetilde{p}_{i+1} = \widetilde{p}_i - \ \Delta\tau V'(\widetilde{q}_i) + \Delta\tau\ G^2(\widetilde{r}_i - J\widetilde{q}_i), \quad \widetilde{q}_{i+1} = \widetilde{q}_i + \Delta\tau\ \widetilde{p}_{i+1}, \\ \widetilde{k}_{i+1} \sim k_{i+1} \mid c_i = \widetilde{c}_i, \quad \widetilde{r}_{i+1} = f_{\widetilde{k}_{i+1}}(\widetilde{q}_{i+1}). \end{aligned} \tag{2.7}$$

We take linear functions f_{k_i}. An illustration of such functions fitted over a (q, r)-scatter plot is shown in Figure 2.7.

The conditioning variables c_i contain both model variables (e.g. q_i) and

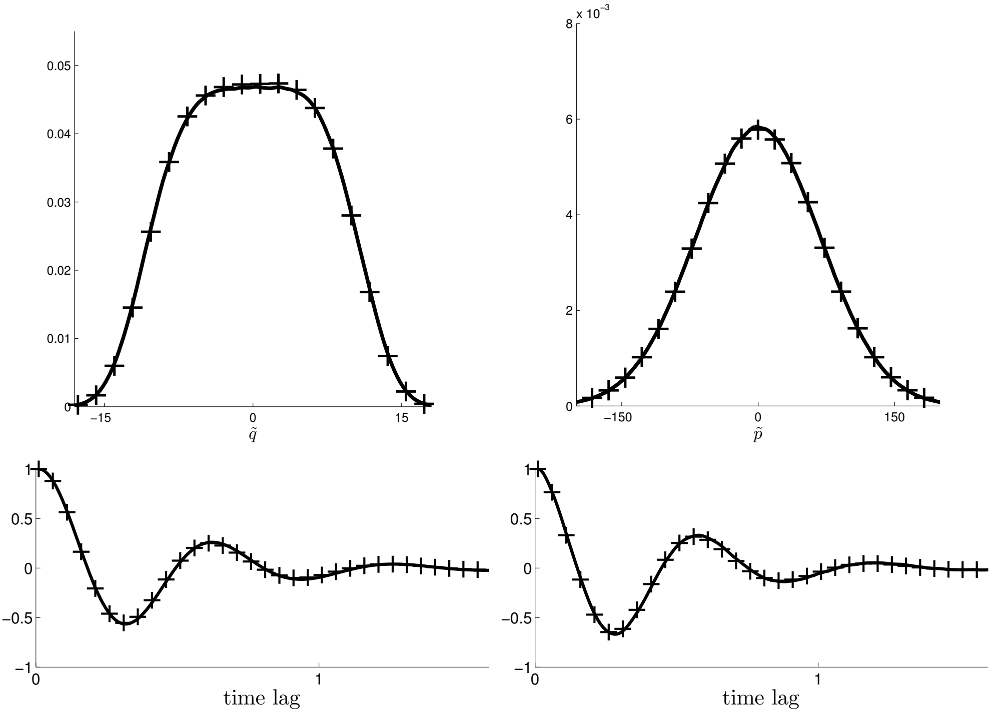


Figure 2.6: Distributions (top) and autocorrelation functions (bottom) for positions (left) and momenta (right). The conditioned empirical distributions are sampled from $r_{i+1} \mid q_i, r_i, r_{i-1}$. A comparison between the distributions and autocorrelations resulting from the reduced model (marked by +) and from the resolved model (solid lines).

indices (e.g. k_i). The model variables are continuous, so they are binned as described in section 2.3. Although many choices for c_i are possible, here we consider two sets, denoted $c_{i,3}$ and $c_{i,4}$ and defined as $c_{i,3} = \{q_i, q_{i+1}, k_i\}$ and $c_{i,4} = \{q_i, q_{i+1}, k_i, k_{i-1}\}$. We emphasize that $c_{i,3}$ and $c_{i,4}$ are not implicit conditioning sets, because $\widetilde{q}_{i+1}$ is calculated before $\widetilde{r}_{i+1}$ is updated, see (2.7). As k_i can take integer values ranging from 1 to K, the transition from k_i to k_{i+1} is governed by a set of $(K \times K)$ transition probability matrices in the case of $c_{i,3}$, one matrix for every bin α_b. There are $B = (N_B)^C$ bins in total, where C is the number of continuous variables in c_i ($C = 2$ for $c_{i,3}$ and $c_{i,4}$). With $c_{i,4}$, there are $B\,K$ transition probability matrices of size $(K \times K)$, due to the additional conditioning on k_{i-1}.

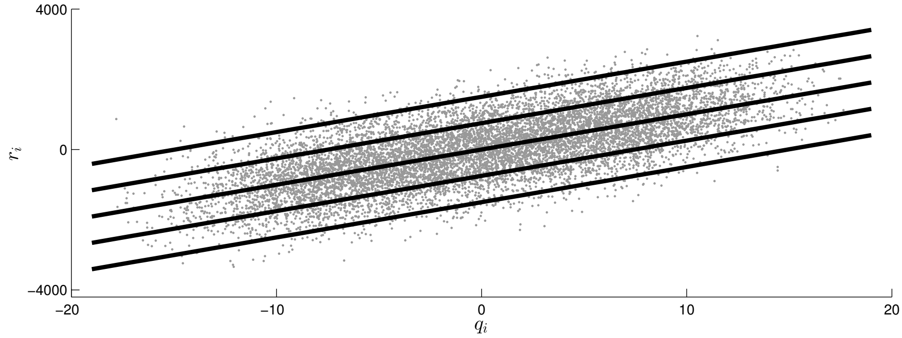


Figure 2.7: Example of five linear functions f_k fitted over the scatter plot of q_i vs. r_i.

3.2.2 Numerical results.

To approximate the bin-wise transition probabilities one first applies the mapping $(q_i, r_i) \rightarrow (q_i, k_i)$ to all data points, where $k_i := \arg\min_k |r_i - f_k(q_i)|$, i.e. k_i is chosen so that f_{k_i} is the function with minimal distance to the point (q_i, r_i) in the r-direction. After applying this mapping, one can easily count occurrences of transition paths in the sample data.

Constructing the transition probability matrices in this manner implies that k_{i+1} is dependent on all of k_i, q_i and q_{i+1}. This has as effect that, for correct usage of these transition probabilities in the reduced model, the conditioning variables should at least include q_i, q_{i+1} and k_i. In fact, we found that simulations where c_i does not include all three of these are often unstable.

Figure 2.8 compares the reduced model results of the simulations with conditioning variables $c_{i,3} = \{q_i, q_{i+1}, k_i\}$ and $c_{i,4} = \{q_i, q_{i+1}, k_i, k_{i-1}\}$. The conditioning variable k_{i-1} added in $c_{i,4}$ significantly improves the reproduced autocorrelation functions, similar to the results of the empirical distribution in section 3.1.2.

The sample moments of the resolved simulation and the reduced simulations are shown in Table 2.3. This table shows that the conditioning parameters $c_{i,3}$ give a better approximation of moments of q and p than $c_{i,4}$, although with $c_{i,4}$ the autocorrelation functions are reproduced more accurately. Because we posed in section 1.2 that additional conditional variables to the distribution of $\widetilde{r}$ should result in increased accuracy of the reduced model, this result is unexpected. However, a large number of parameters must be estimated to

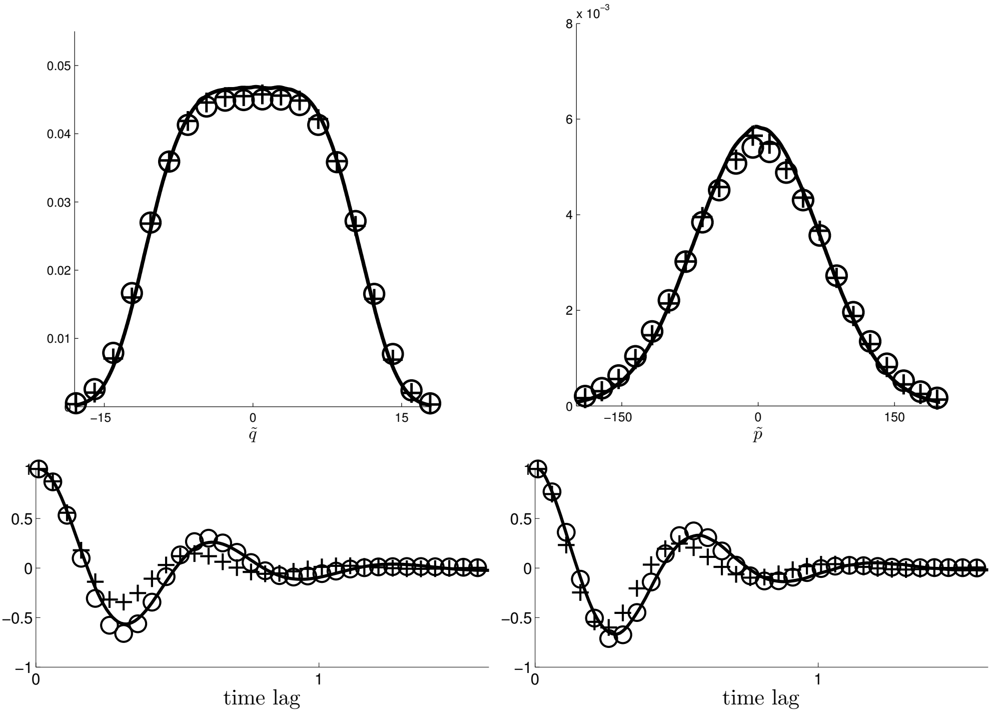


Figure 2.8: Distributions (top) and autocorrelation functions (bottom) for positions (left) and momenta (right). The CMC approach approximates sampling from $r_{i+1} \mid c_i$. A comparison between the distributions and autocorrelations resulting from the reduced models for $c_{i,3} = \{q_i, q_{i+1}, k_i\}$ (marked by $+$) and for $c_{i,4} = \{q_i, q_{i+1}, k_i, k_{i-1}\}$ (marked by $\circ$), and from the resolved model (solid lines).

approximate the distribution of $k_{i+1} \mid c_i$. We recall the following definitions: C and D are the number of continuous and discrete variables in c_i, $B = (N_B)^C$ is the total number of bins, and K is the number of different functions $f_k(q)$. The number of parameters to be estimated for the CMC approach conditioning on a set of variables c_i is given by $(N_B)^C\, K^{D+1}$.

For the results in Figure 2.8 and Table 2.3 we used $K = 9$ and $B = 100$ (10×10 bins for q_i and q_{i+1} combined). This results in 8100 parameters when using $c_{i,3}$ and 72900 parameters when using $c_{i,4}$. This exponential scaling of the number of parameters is the bottleneck of the CMC approach: even for relatively simple problems it requires a very large data set to approximate all transition probabilities accurately.

Table 2.3: Sample moments for the CMC approximations

	#params.	mean	std.dev.	skewness	kurtosis
$\boldsymbol{x}_i$		$\gamma_1(\boldsymbol{x}_i)$	$\gamma_2(\boldsymbol{x}_i)$	$\gamma_3(\boldsymbol{x}_i)$	$\gamma_4(\boldsymbol{x}_i)$
p_i (reference)	–	0.00	68.4	$3.7 \cdot 10^{-4}$	3.00
$\widetilde{p}_i$ $(c_{i,3} = \{q_i, q_{i+1}, k_i\})$	8100	0.00	71.8	$1.2 \cdot 10^{-3}$	3.00
$\widetilde{p}_i$ $(c_{i,4} = \{q_i, q_{i+1}, k_i, k_{i-1}\})$	72900	0.00	74.3	$-3.4 \cdot 10^{-4}$	3.02
q_i (reference)	–	0.01	6.83	$-5.5 \cdot 10^{-3}$	2.18
$\widetilde{q}_i$ $(c_{i,3} = \{q_i, q_{i+1}, k_i\})$	8100	0.00	7.00	$-3.4 \cdot 10^{-3}$	2.18
$\widetilde{q}_i$ $(c_{i,4} = \{q_i, q_{i+1}, k_i, k_{i-1}\})$	72900	0.00	7.11	$-2.8 \cdot 10^{-3}$	2.19

Due to the described stability issues and exponential scaling of the number of parameters we choose not to pursue the CMC approaches any further here. Instead, in the next section we explore the use of a continuous-in-space stochastic process, so that the number of parameters remains minimal.

3.3 Ornstein–Uhlenbeck process.

As discussed in section 3.2.2, the CMC strategy requires a very large number of estimated parameters. In this section we present a stochastic representation that reduces the number of parameters significantly. Let us assume that the evolution of r can be approximated by an Ornstein–Uhlenbeck (OU) process:

$$\dot{r} = -\theta(r - \mu) + \sigma \dot{W},$$

with Wiener process W and unknown parameters μ, θ and σ. The evolution of r, as observed from the full model, is then used to approximate an OU process $\widetilde{r}$ defined by:

$$\dot{\widetilde{r}} = -\widehat{\theta}(\widetilde{r} - \widehat{\mu}) + \widehat{\sigma}\dot{W}. \tag{2.8}$$

The parameters $\widehat{\boldsymbol{\theta}} := (\widehat{\mu}, \widehat{\theta}, \widehat{\sigma})$ in (2.8) approximate the OU parameters $\boldsymbol{\theta} := (\mu, \theta, \sigma)$, thus implicitly fitting $\widetilde{r}$ to r. In the following sections we discuss different methods for defining these OU estimators. We start in section 3.3.1 with constant $\widehat{\boldsymbol{\theta}}$ (i.e., independent of c_i), whereas in later sections we let $\widehat{\boldsymbol{\theta}}$ depend on c_i.

3.3.1 Unconditional parameters.

Introduce the notations $R_c = \sum_{i=1}^{M} r_i$, $R_m = \sum_{i=1}^{M} r_{i-1}$, $R_{cc} = \sum_{i=1}^{M} r_i^2$, $R_{mm} = \sum_{i=1}^{M} r_{i-1}^2$ and $R_{cm} = \sum_{i=1}^{M} r_i r_{i-1}$. The subscripts c and m are chosen to denote *current* and *minus*, respectively. Then, assuming a zero-limit of the sampling interval δt, the *standard* discrete-in-time estimators $\widehat{\boldsymbol{\theta}}^{\text{st}} := (\widehat{\mu}^{\text{st}}, \widehat{\theta}^{\text{st}}, \widehat{\sigma}^{\text{st}})$ for the OU parameters are given by [113]:

$$\begin{aligned} \widehat{\mu}^{\text{st}} &= M^{-1} R_c, \\ \widehat{\theta}^{\text{st}} &= \frac{R_{mm} - R_{cm} - \widehat{\mu}^{\text{st}}(R_m - R_c)}{\delta t (R_{mm} - 2\widehat{\mu}^{\text{st}} R_m + M(\widehat{\mu}^{\text{st}})^2)}, \\ (\widehat{\sigma}^{\text{st}})^2 &= M^{-1} \delta t^{-1} (R_{cc} - 2R_{cm} + R_{mm}). \end{aligned} \tag{2.9}$$

Sometimes, however, a small δt can not be guaranteed because of run-time requirements, or a small δt is undesired [113]. If δt is not small, the estimators in (2.9) are biased. Therefore, let us also consider the more exact maximum likelihood (ML) estimators $\widehat{\boldsymbol{\theta}}^{\text{ex}} := (\widehat{\mu}^{\text{ex}}, \widehat{\theta}^{\text{ex}}, \widehat{\sigma}^{\text{ex}})$, as discussed in, e.g, [139]. By omitting the assumption $\delta t \to 0$ and using the Markovian nature of the OU process, these exact ML estimators follow from maximizing the log likelihood function:

$$\log \mathcal{L}(\widehat{\boldsymbol{\theta}}^{\text{ex}} \mid \boldsymbol{R}) = \log P(r_0 \mid \widehat{\boldsymbol{\theta}}^{\text{ex}}) + \sum_{i=1}^{M} \log P(r_i \mid r_{i-1}, \widehat{\boldsymbol{\theta}}^{\text{ex}}). \tag{2.10}$$

Making the additional assumption that the sample data is stationary, we know:

$$r_i \mid r_{i-1}, \widehat{\boldsymbol{\theta}}^{\text{ex}} \sim \mathcal{N}\left(r_{i-1}\eta + \widehat{\mu}^{\text{ex}}(1-\eta), (\zeta \widehat{\sigma}^{\text{ex}})^2\right),$$

where $\eta := \exp(-\widehat{\theta}^{\text{ex}} \delta t)$ and $\zeta^2 := (2\widehat{\theta}^{\text{ex}})^{-1}(1-\eta^2)$.

We assume the distribution of r_0 does not depend on $\widehat{\boldsymbol{\theta}}$, therefore we ignore the term $P(r_0 \mid \widehat{\boldsymbol{\theta}}^{\text{ex}})$ for the maximization of (2.10). Substituting the conditional probabilities and removing the conditional distribution $P(r_0 \mid \widehat{\boldsymbol{\theta}}^{\text{ex}})$ from (2.10) results in the following log likelihood:

$$\begin{aligned} \log \mathcal{L}(\widehat{\boldsymbol{\theta}}^{\text{ex}} \mid \boldsymbol{R}) \approx \sum_{i=1}^{M} \log P(r_i \mid r_{i-1}, \widehat{\boldsymbol{\theta}}^{\text{ex}}) = -\frac{M}{2} \log(2\pi) - \\ M \log(\zeta \widehat{\sigma}^{\text{ex}}) - \frac{1}{2(\zeta \widehat{\sigma}^{\text{ex}})^2} \sum_{i=1}^{M} \left(r_i - r_{i-1}\eta - \widehat{\mu}^{\text{ex}}(1-\eta)\right)^2. \end{aligned} \tag{2.11}$$

By maximizing (2.11) with respect to each of the parameters, the exact ML estimators are found to equal:

$$\begin{aligned} \widehat{\mu}^{\mathrm{ex}} &= \frac{R_c R_{mm} - R_m R_{cm}}{M(R_{mm} - R_{cm}) - R_m^2 + R_c R_m}, \\ \widehat{\theta}^{\mathrm{ex}} &= -\delta t^{-1} \log \frac{R_{cm} - \widehat{\mu}^{\mathrm{ex}}(R_c + R_m) + M(\widehat{\mu}^{\mathrm{ex}})^2}{R_{mm} - 2\widehat{\mu}^{\mathrm{ex}} R_m + M(\widehat{\mu}^{\mathrm{ex}})^2}, \\ (\widehat{\sigma}^{\mathrm{ex}})^2 &= 2\widehat{\theta}^{\mathrm{ex}} M^{-1}(1-\eta^2)^{-1} \ (\ R_{cc} - 2\eta R_{cm} + \eta^2 R_{mm} \\ &\qquad - 2\widehat{\mu}^{\mathrm{ex}}(R_c - \eta R_m)(1-\eta) + M(\widehat{\mu}^{\mathrm{ex}})^2(1-\eta)^2 \ ). \end{aligned} \tag{2.12}$$

These estimators are equivalent to the standard ML estimators (2.9) if one assumes the limits $\delta t \to 0$ and $M \to \infty$, see Appendix A. Note that the exact ML estimators (2.12) can be calculated sequentially from sample data.

Next, let us compare the quality of the respective methods by fitting both sets of estimators to sample data generated by a reference OU process with known parameters. Because both $\widehat{\mu}^{\mathrm{st}}$ and $\widehat{\mu}^{\mathrm{ex}}$ are independent of δt, we only compare approximations for σ and θ. Both the standard and exact ML estimators, fitted to this reference process, are shown in Figure 2.9. This figure shows that the standard ML estimators (2.9) indeed become strongly biased as δt increases, whereas the exact ML estimators (2.12) remain very accurate up to at least δt values of 1.5, where sampling error starts to be an issue. Therefore, the exact ML estimators are the clear choice for the rest of our experiments.

3.3.2 Conditional parameters with binning.

We now generalize the methods from section 3.3.1 to be in line with those in sections 3.1 and 3.2 by conditioning the OU parameters (and thus the process $\widetilde{\boldsymbol{R}}$), on the model variables c. Building on the binning strategy, as explained in section 2.3, we define estimators $\widehat{\boldsymbol{\theta}}^{\mathrm{pc}} := (\widehat{\mu}^{\mathrm{pc}}, \widehat{\theta}^{\mathrm{pc}}, \widehat{\sigma}^{\mathrm{pc}})$ that are piece-wise constant in c_i. It must be mentioned that this approach implicitly relies on small δt because the piece-wise constant assumption.

The c_i-dependency, being piece-wise constant, can be included in the likelihood function. First, we introduce the notation:

$$\widehat{\mu}^{\mathrm{pc}}(c_i) := \widehat{\mu}_b^{\mathrm{pc}}, \quad \widehat{\theta}^{\mathrm{pc}}(c_i) := \widehat{\theta}_b^{\mathrm{pc}}, \quad \widehat{\sigma}^{\mathrm{pc}}(c_i) := \widehat{\sigma}_b^{\mathrm{pc}}, \quad \text{if } c_i \in \alpha_b.$$

The parameters $\widehat{\boldsymbol{\theta}}_b^{\mathrm{pc}} := (\widehat{\mu}_b^{\mathrm{pc}}, \widehat{\theta}_b^{\mathrm{pc}}, \widehat{\sigma}_b^{\mathrm{pc}})$ can be calculated by restricting the esti-

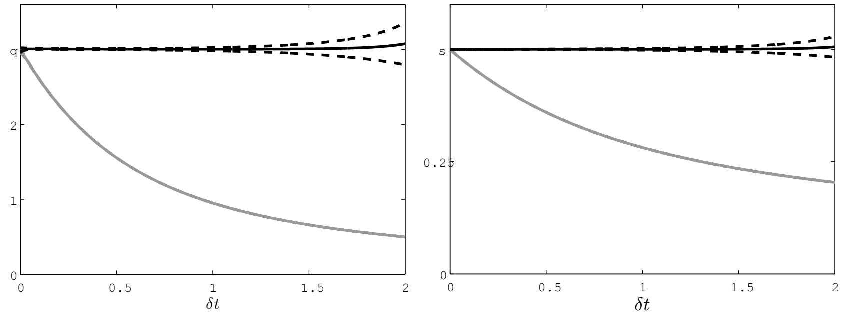


Figure 2.9: Mean (solid) and standard deviation (dashed) of the standard (gray) and exact (black) ML estimators, in (2.9) and (2.12) respectively, for a reference OU process with $(\mu, \sigma, \theta) = (1, 0.5, 3)$. The estimates plotted for each sampling interval δt are averages over 100 independent OU simulations with the given parameters. Each OU simulation stores 10^6 data points, where a data point is saved after 100 time steps of the reference process. The sampling interval of the OU simulations is 10^{-3}. We test the estimators as δt ranges from 10^{-3} to 2, in increments of 10^{-3}. This causes the growing sampling error shown as $\delta t \to 2$. Note that while the standard deviation of the standard ML estimators (gray dotted lines) is plotted in the figures, these dotted lines lie too close to the standard ML estimator mean to be visible.

mators (2.12) to the sample data points that lie in α_b. Note that we assume that r_i is only dependent on c_i, and not on $c_{i'}$ with $i' < i$. Similar to (2.11), the log likelihood function can now be written as:

$$\log \mathcal{L}(\widehat{\boldsymbol{\theta}}^{\mathrm{pc}} \mid \boldsymbol{R}, \boldsymbol{C}) \approx \sum_{i=1}^{M} \log P(r_i \mid r_{i-1}, \widehat{\boldsymbol{\theta}}_b^{\mathrm{pc}}), \quad \text{where } c_{i-1} \in \alpha_b \tag{2.13}$$

Maximizing (2.13) over the parameters ($3B$ in total) is straightforward and leads to the following estimators for each of the bins:

$$\begin{aligned}
\widehat{\mu}_b^{\mathrm{pc}} &= \frac{R_{b,c} R_{b,mm} - R_{b,m} R_{b,cm}}{|\rho_b|(R_{b,mm} - R_{b,cm}) - R_{b,m}^2 + R_{b,c} R_{b,m}}, \\
\widehat{\theta}_b^{\mathrm{pc}} &= -\delta t^{-1} \log \frac{R_{b,cm} - \widehat{\mu}_b^{\mathrm{pc}}(R_{b,c} + R_{b,m}) + |\rho_b|(\widehat{\mu}_b^{\mathrm{pc}})^2}{R_{b,mm} - 2\widehat{\mu}_b^{\mathrm{pc}} R_{b,m} + |\rho_b|(\widehat{\mu}_b^{\mathrm{pc}})^2}, \\
(\widehat{\sigma}_b^{\mathrm{pc}})^2 &= 2\widehat{\theta}_b^{\mathrm{pc}} |\rho_b|^{-1} (1 - \eta_b^2)^{-1} \; \big(\; R_{b,cc} - 2\eta_b R_{b,cm} + \eta_b^2 R_{b,mm} \\
&\qquad - 2\widehat{\mu}_b^{\mathrm{pc}}(R_{b,c} - \eta_b R_{b,m})(1 - \eta_b) + |\rho_b|(\widehat{\mu}_b^{\mathrm{pc}})^2 (1 - \eta_b)^2 \; \big),
\end{aligned} \tag{2.14}$$

where $|\rho_b|$ is the number of sample points in the bin α_b. Analogous to before, the following notations are used to restrict terms to a specific bin b: $\eta_b = \exp(-\widehat{\theta}_b^{\mathrm{pc}}\delta t)$, $R_{b,c} = \sum_{i=1}^{M} r_i \mathbb{1}(c_{i-1} \in \alpha_b)$, $R_{b,m} = \sum_{i=1}^{M} r_{i-1}\mathbb{1}(c_{i-1} \in \alpha_b)$, $R_{b,cc} = \sum_{i=1}^{M} r_i^2 \mathbb{1}(c_{i-1} \in \alpha_b)$, $R_{b,mm} = \sum_{i=1}^{M} r_{i-1}^2 \mathbb{1}(c_{i-1} \in \alpha_b)$ and $R_{b,cm} = \sum_{i=1}^{M} r_i r_{i-1}\mathbb{1}(c_{i-1} \in \alpha_b)$.

Let us illustrate this approach by calculating the bin-wise estimators for the one-dimensional conditioning $r_{i+1} \mid q_i$. The stationary distribution of an OU process with parameters $(\widehat{\mu}_b^{\mathrm{pc}}, \widehat{\theta}_b^{\mathrm{pc}}, \widehat{\sigma}_b^{\mathrm{pc}})$ is $\mathcal{N}(\widehat{\mu}_b^{\mathrm{pc}}, (\widehat{\sigma}_b^{\mathrm{pc}})^2/2\widehat{\theta}_b^{\mathrm{pc}})$; the resulting mean and standard deviation for each bin are plotted over a (q, r) scatter plot in Figure 2.10.

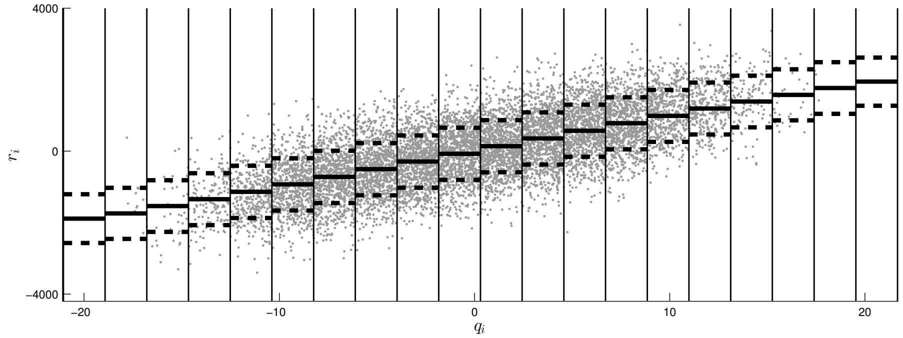


Figure 2.10: The mean (solid lines) and standard deviation (dotted lines) described by the stationary distribution of the OU estimators for each of the 20 bins approximating the distribution $r_{i+1} \mid q_i$. (Note that only 1% of the total number of data points used to obtain the estimators is shown in the plot.)

3.3.3 Conditional parameters with a linearly fitted mean.

In the specific case $\widetilde{r}_{i+1} \sim r_{i+1} \mid q_i$, the means and standard deviations of the OU processes in the different bins are approximately linear (in q) and constant, respectively, as can be seen in Figure 2.10. In fact, our experiments show that the OU parameters themselves are either (approximately) constant ($\widehat{\theta}_b^{\mathrm{pc}}$ and $\widehat{\sigma}_b^{\mathrm{pc}}$), or linear in q ($\widehat{\mu}_b^{\mathrm{pc}}$). This indicates that we can reduce the total number of parameters significantly by using constant or linear functions of q. Thus, we define $(\widehat{\mu}^{\mathrm{lf}}(q), \widehat{\theta}^{\mathrm{lf}}(q), \widehat{\sigma}^{\mathrm{lf}}(q))$ by

$$\widehat{\mu}^{\mathrm{lf}}(q_i) = \widehat{\mu}_0^{\mathrm{lf}} + \widehat{\mu}_1^{\mathrm{lf}} q_i, \quad \widehat{\theta}^{\mathrm{lf}}(q_i) = \widehat{\theta}_0^{\mathrm{lf}}, \quad \widehat{\sigma}^{\mathrm{lf}}(q_i) = \widehat{\sigma}_0^{\mathrm{lf}}, \tag{2.15}$$

where $\widehat{\boldsymbol{\theta}}^{\mathrm{lf}} := (\widehat{\mu}_0^{\mathrm{lf}}, \widehat{\mu}_1^{\mathrm{lf}}, \widehat{\theta}_0^{\mathrm{lf}}, \widehat{\sigma}_0^{\mathrm{lf}})$ is constant. When compared to the piece-wise constant OU estimators (2.14), this approach reduces the number of OU parameters from $3B$ to 4. Similar to (2.13), one can write the log likelihood function for the parameters in (2.15) as:

$$\log \mathcal{L}(\widehat{\boldsymbol{\theta}}^{\mathrm{lf}} \mid \boldsymbol{R}, \boldsymbol{Q}) \approx \sum_{i=1}^{M} P(r_i \mid r_{i-1}, q_{i-1}, \widehat{\boldsymbol{\theta}}^{\mathrm{lf}}). \tag{2.16}$$

Analogous to section 3.3.1, one obtains expressions for the estimators $\widehat{\boldsymbol{\theta}}^{\mathrm{lf}}$ by maximizing (2.16) with respect to each of the parameters $\widehat{\mu}_0^{\mathrm{lf}}$, $\widehat{\mu}_1^{\mathrm{lf}}$, $\widehat{\theta}_0^{\mathrm{lf}}$ and $\widehat{\sigma}_0^{\mathrm{lf}}$ (see Appendix B for details on notation):

$$\begin{aligned}
\widehat{\mu}_1^{\mathrm{lf}} &= -\frac{1}{3P_3}\left(P_2 + C + \frac{\Delta_0}{C}\right), \\
\widehat{\mu}_0^{\mathrm{lf}} &= \frac{(\widehat{\mu}_1^{\mathrm{lf}})^2 A_0 + \widehat{\mu}_1^{\mathrm{lf}} B_0 + C_0}{\widehat{\mu}_1^{\mathrm{lf}} D_0 + E_0} \\
\widehat{\theta}_0^{\mathrm{lf}} &= -\delta t^{-1} \log(\ (R_{mm} - 2\widehat{\mu}_0^{\mathrm{lf}} R_m - 2\widehat{\mu}_1^{\mathrm{lf}} X_{mm} + 2\widehat{\mu}_0^{\mathrm{lf}}\widehat{\mu}_1^{\mathrm{lf}} Q_m + M(\widehat{\mu}_0^{\mathrm{lf}})^2 \\
&\qquad + (\widehat{\mu}_1^{\mathrm{lf}})^2 Q_{mm})^{-1}(R_{cm} - \widehat{\mu}_0^{\mathrm{lf}}(R_c + R_m) - \widehat{\mu}_1^{\mathrm{lf}}(X_{cm} + X_{mm}) \\
&\qquad + 2\widehat{\mu}_0^{\mathrm{lf}}\widehat{\mu}_1^{\mathrm{lf}} Q_m + M(\widehat{\mu}_0^{\mathrm{lf}})^2 + (\widehat{\mu}_1^{\mathrm{lf}})^2 Q_{mm})\ ) \\
(\widehat{\sigma}_0^{\mathrm{lf}})^2 &= 2\widehat{\theta}_0^{\mathrm{lf}} M^{-1}(1 - \eta_0^2)^{-1}\ (\ R_{cc} - 2\eta_0 R_{cm} + \eta_0^2 R_{mm} \\
&\qquad + (1 - \eta_0)(2\widehat{\mu}_0^{\mathrm{lf}}(\eta_0 R_m - R_c) + 2\widehat{\mu}_1^{\mathrm{lf}}(\eta_0 X_{mm} - X_{cm})) \\
&\qquad + (1 - \eta_0)^2(M(\widehat{\mu}_0^{\mathrm{lf}})^2 + 2\widehat{\mu}_0^{\mathrm{lf}}\widehat{\mu}_1^{\mathrm{lf}} Q_m + (\widehat{\mu}_1^{\mathrm{lf}})^2 Q_{mm}\ ).
\end{aligned} \tag{2.17}$$

The stationary distribution of the OU process with q fixed is given by $\mathcal{N}(\widehat{\mu}_0^{\mathrm{lf}} + q\widehat{\mu}_1^{\mathrm{lf}}, (\widehat{\sigma}_0^{\mathrm{lf}})^2 / 2\widehat{\theta}_0^{\mathrm{lf}})$; the resulting mean and standard deviation are plotted over a (q, r) scatter plot in 2.11.

3.3.4 Numerical results.

As discussed in section 2.3, it is possible that not all bins contain samples if they are equally sized. For the empirical approach in section 3.1, this posed no serious problem. However, the accuracy of the estimated OU parameters is strongly affected if the sample size is too small. To keep the tests between methods comparable, we opt not to change the binning procedure, but instead to consider bins with less than 100 samples as empty.

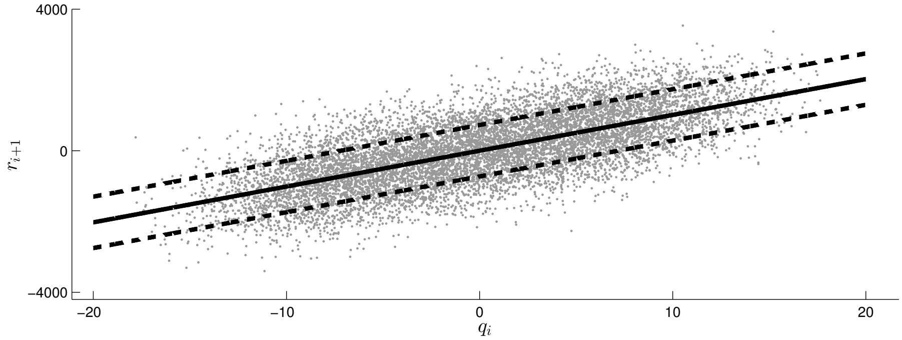


Figure 2.11: The mean and standard deviation of the stationary distribution described by linear OU parameters that approximate sampling $\widetilde{r}_{i+1}$ from the distribution of $r_{i+1} \mid q_i$. (Note that only 1% of the total number of data points used to obtain the estimators is shown in the plot.)

Simulations that sample $\widetilde{r}$ from the unconditioned distribution of r (using the constant ML estimators in (2.12)) are unstable. However, modeling $\widetilde{r}$ as an OU process that is either piece-wise constant or linear in c_i (using the bin-wise or linearly fitted ML estimators, (2.14) or (2.17)) compares favorably to the previously discussed strategies. Whereas both the empirical and CMC approaches need 2 and 3 conditioning variables, respectively, to accurately reproduce the distributions of q and p, the reduced simulations using the conditioned OU process need only $c_{i,1} = \{q_i\}$ to reproduce these distributions very accurately. These results are illustrated in Figure 2.12 and presented with more detail in Table 2.4. The accurate reconstruction of the model variables' distributions is especially impressive for the OU parameters with linearly fitted mean (referred to from now on as *linearly fitted* OU parameters), because the linearly fitted OU process only uses 4 parameters, whereas the CMC approach and bin-wise OU approach need $(N_B)^C\, K^{D+1}$ (see section 3.2.2) and $3(N_B)^C$ parameters respectively. As is the case with all other strategies, however, the autocorrelation function is reconstructed less accurately for $c_{i,1}$.

A downside of the linearly fitted OU parameters is that they are defined specifically for the case $c_{i,1} = \{q_i\}$. Generalization to other cases is nontrivial. The piece-wise constant OU parameters, however, can be easily conditioned on multiple variables. Similarly to the empirical approach, the resolved autocorrelation functions are approximated with high accuracy when the conditioning

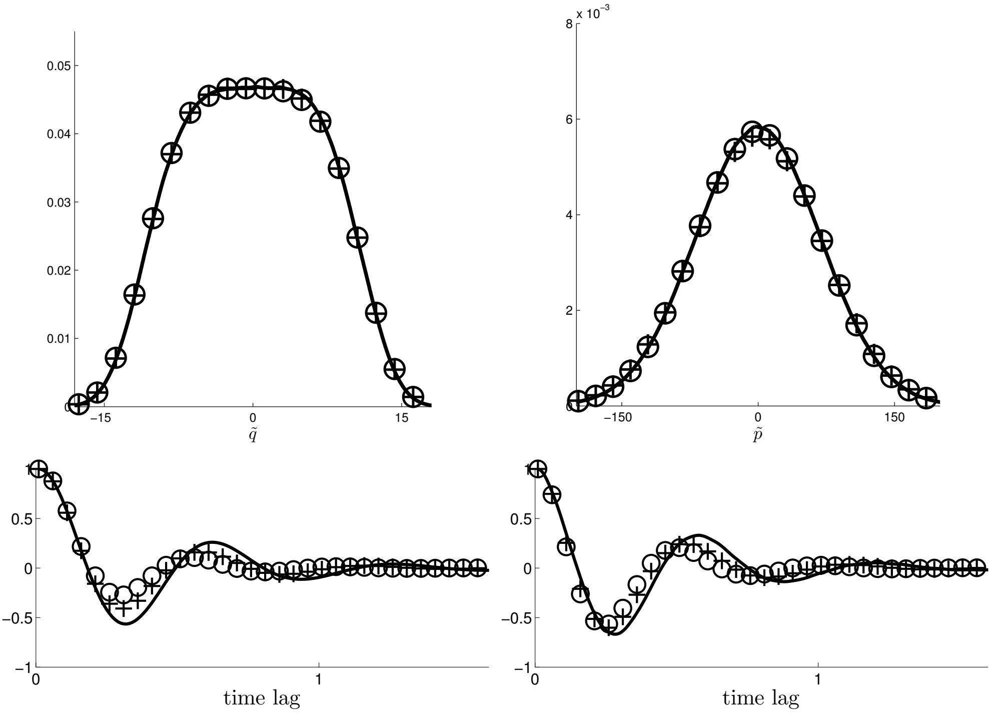


Figure 2.12: Distributions (top) and autocorrelation functions (bottom) for positions (left) and momenta (right). The applied OU approaches approximate sampling from $r_{i+1} \mid q_i$. A comparison between the distributions and autocorrelations resulting from the bin-wise (marked by $+$) and linearly fitted (marked by $\circ$) ML estimators, and from the resolved model (solid lines).

variables are extended to $c_{i,3} = \{q_i, r_i, r_{i-1}\}$, as shown in Figure 2.13.

4 Discussion

In this study we investigate how to use sample data, generated by a fully resolved multiscale model, to construct stochastic representations of unresolved processes in reduced models. We discuss three methods to model these stochastic representations, and tested the methods using the Kac–Zwanzig heat bath model. This heat bath model describes the dynamics of a distinguished particle, which is coupled linearly to a number of heat bath particles and moves over a potential. The stochastic methods aim to model the dynamical effects of the heat bath particles to drive a reduced model that only resolves the dis-

Table 2.4: Sample moments for the OU approximations

$\boldsymbol{x}_i$	#params.	mean $\gamma_1(\boldsymbol{x}_i)$	std.dev. $\gamma_2(\boldsymbol{x}_i)$	skewness $\gamma_3(\boldsymbol{x}_i)$	kurtosis $\gamma_4(\boldsymbol{x}_i)$
p_i (reference)	–	0.00	68.4	$3.7 \cdot 10^{-4}$	3.00
$\widetilde{p}_i$ ($c_{i,1} = \{q_i\}$) linearly fitted	4	0.00	69.2	$4.5 \cdot 10^{-4}$	3.01
$\widetilde{p}_i$ ($c_{i,1} = \{q_i\}$) bin-wise	30	0.00	70.3	$-1.2 \cdot 10^{-3}$	3.02
$\widetilde{p}_i$ ($c_{i,2} = \{q_i, r_i\}$)	300	0.00	72.1	$-2.6 \cdot 10^{-3}$	3.02
$\widetilde{p}_i$ ($c_{i,3} = \{q_i, r_i, r_{i-1}\}$)	3000	0.00	69.3	$-3.5 \cdot 10^{-3}$	2.99
q_i (reference)	–	0.01	6.83	$-5.5 \cdot 10^{-3}$	2.18
$\widetilde{q}_i$ ($c_{i,1} = \{q_i\}$) linearly fitted	4	-0.01	6.87	$2.8 \cdot 10^{-3}$	2.18
$\widetilde{q}_i$ ($c_{i,1} = \{q_i\}$) bin-wise	30	0.00	6.86	$3.6 \cdot 10^{-4}$	2.19
$\widetilde{q}_i$ ($c_{i,2} = \{q_i, r_i\}$)	300	-0.01	6.94	$-1.3 \cdot 10^{-4}$	2.19
$\widetilde{q}_i$ ($c_{i,3} = \{q_i, r_i, r_{i-1}\}$)	3000	0.01	6.82	$-1.8 \cdot 10^{-3}$	2.18

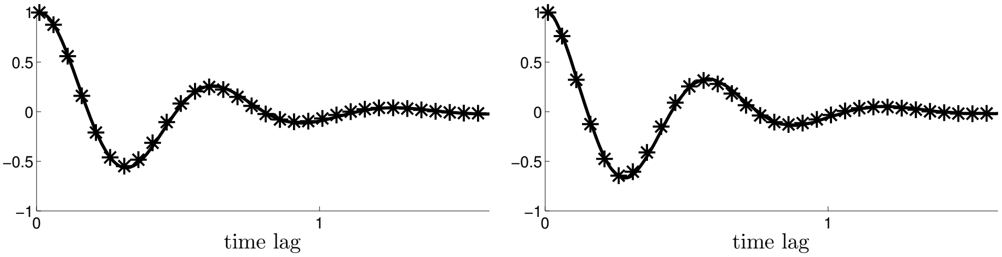


Figure 2.13: Autocorrelation functions for positions (left) and momenta (right). The piece-wise constant OU parameters are fitted to approximate sampling from $r_{i+1} \mid q_i, r_i, r_{i-1}$. A comparison between the autocorrelations resulting from the reduced model (marked by $*$) and from the resolved model (solid lines).

tinguished particle. We compared the fully resolved model and the reduced models by the probability distributions, first four statistical moments and autocorrelation functions of the position q and momentum p of the distinguished particle.

In the reduced models, the sum of the positions of the heat bath particles, denoted r, is modeled as a stochastic process. This is done in three different ways: (i) sampling from the empirical (conditional) distribution of r, (ii) using a discrete Markov chain to switch between several functions $r = f_k(q)$, and (iii) modeling r as an OU process with q-dependent parameters. As mentioned, the stochastic processes driving the reduced model were conditioned on the position of the distinguished particle. In some tests, the past state of r was also added to the set of conditioning variables. Extending the set of conditioning variables improves results, as is demonstrated most visibly in section 3.1.2. We note that extending this set typically increases the number of parameters in the stochastic model, so that more data may be needed to estimate these accurately, see section 3.2. Notwithstanding, with appropriate conditioning of the stochastic process for r, the distributions and autocorrelations of $\widetilde{q}$ and $\widetilde{p}$ in the reduced model resemble those of q and p of the fully resolved model very closely, see in particular figures 2.6 and 2.13.

The advantage of the empirical distribution approach over the other methods is that it is more robust if the available data set is rather small. The empirical distribution samples uniformly from the data, so that any nonempty sample set is, by construction, somewhat representative of the dynamics of r. However, this also restricts the empirical distribution sampling to the range of the data set, which might not be representative of the exact joint distribution of r for small data sets. In this approach, no parameters are estimated, the data only needs to be partitioned into bins. By contrast, the CMC and binned OU approaches are more sensitive to small data sets, because limited data affects the parameter estimates. These approaches involve a large number of parameters that must be estimated, most notably the CMC approach, see tables 2.3 and 2.4.

The linearly fitted OU approach reduces the number of parameters to 4, and is still able to reproduce the distributions of the resolved model variables very accurately. However, we note that extending this approach to one where the OU parameters θ and σ also have functional dependence on q (or some other conditioning variable) will be difficult, as will generalizations to nonlinear functional dependence.

As mentioned, the data needed for fitting the stochastic models for r come from a simulation of the fully resolved model. It may seem superfluous to formulate a reduced model if simulations with the full model are computationally feasible. However, if one wishes to simulate a multiscale system over a very long time interval, but fully resolved simulations are only feasible over a much shorter time interval, an efficient yet accurate reduced model can be very useful. Furthermore, in some cases it is possible to use data from observations instead of simulation data (see [40] for an example). In those situations, data-driven modeling approaches are also useful. Finally, for spatially extended systems such as atmospheric or oceanic flows, a fully resolved simulation may be only computationally feasible on part of the spatial domain of interest. The methods discussed in this study allow one to construct a spatially localized stochastic model for unresolved processes. By using identical yet independent copies of this local stochastic model, one can cover the entire spatial domain.

We emphasize that the computational gain of simulating with the reduced model instead of the fully resolved model can be very large. The Kac-Zwanzig heat bath model as used in this study has 202 degrees of freedom (positions and momenta of the distinguished particle and all 100 heat bath particles), and is integrated with time step 10^{-4}. By contrast, the reduced model has 3 degrees of freedom ($\widetilde{q}$, $\widetilde{p}, \widetilde{r}$) and integration time step 10^{-2}. An application example is [39], where the fully resolved model is a Large Eddy Simulation (LES) model for atmospheric convection. The LES model in that study used $512 \times 512 \times 80$ gridpoints for spatial discretization, and an integration time step on the order of seconds. The CMC used to represent convection as simulated by the LES model contained 10 discrete states, with random switching between the discrete states at time steps of 1 minute.

In future work we aim to use the methods presented here for ocean circulation models. For example, the strategies described in [117] propose a covariate that correlates strongly to the residual term in reduced vorticity equations. Investigating how our methods can be applied to such models is an exciting topic for future study.

Appendix

A The equivalence of the exact and standard ML estimators in appropriate limits

The unconditional maximum likelihood (ML) estimators, as described in section 3.3.1, are obtained by maximizing the log likelihood in (2.11). Here we make a distinction between two types of ML estimators: the standard estimators $\widehat{\boldsymbol{\theta}}^{\mathrm{st}} = (\widehat{\mu}^{\mathrm{st}}, \widehat{\theta}^{\mathrm{st}}, \widehat{\sigma}^{\mathrm{st}})$ in (2.9), and the exact estimators $\widehat{\boldsymbol{\theta}}^{\mathrm{ex}} = (\widehat{\mu}^{\mathrm{ex}}, \widehat{\theta}^{\mathrm{ex}}, \widehat{\sigma}^{\mathrm{ex}})$ in (2.12). The standard ML estimators are obtained by imposing the limit $\delta t \downarrow 0$ on the log likelihood equation, but the exact ML estimators make no such assumption. However, we show here that the exact estimators tend to the standard estimators as δt and the sample size M go to 0 and ∞, respectively.

Let us make the following assumptions about the model's sample data $\boldsymbol{R} = (r_0, r_1, \ldots, r_M)$:

1. $|r_i| < \infty$, for $0 \leq i \leq M$.
2. $\bar{r} := \mathbf{E}(r_i)\ \forall i$ (stationarity), and $|\bar{r}| < \infty$.
3. $\mathbb{V}\mathrm{ar}(r) = \mathbf{E}(r_i - \bar{r})^2\ \forall i$ (stationarity), and $0 < \mathbb{V}\mathrm{ar}(r) < \infty$

4. ACF $:= (\mathbf{E}(r_i r_{i-1}) - \bar{r}^2)\mathbb{V}\mathrm{ar}(r)^{-1}$, and $|\mathrm{ACF}| < 1$. Note that ACF is dependent on δt, therefore let us also assume:
    (a) $\lim_{\delta t \downarrow 0} \mathrm{ACF} = 1$.
    (b) $\lim_{\delta t \downarrow 0}(\delta t^{-1}(\mathrm{ACF}-1)) =: -a$, with $0 < a < \infty$. This essentially restricts the right derivative of the autocorrelation function from nearing infinite or zero as δt goes to 0.

To show that the standard and exact ML estimators are equivalent in the limits $M \to \infty$ and $\delta t \downarrow 0$, we will first consider each estimator in the limit $M \to \infty$. Let us therefore list the following known properties:

$$\begin{aligned}
&\lim_{M\to\infty}(M^{-1}R_c) = \bar{r},\\
&\lim_{M\to\infty}(M^{-1}R_m) = \bar{r},\\
&\lim_{M\to\infty}(M^{-1}R_{cc}) = \bar{r}^2 + \mathbb{V}\mathrm{ar}(r),\\
&\lim_{M\to\infty}(M^{-1}R_{mm}) = \bar{r}^2 + \mathbb{V}\mathrm{ar}(r),\\
&\lim_{M\to\infty}(M^{-1}R_{cm}) = \mathbf{E}(r_i r_{i-1}) = \bar{r}^2 + \mathbb{V}\mathrm{ar}(r)\,\mathrm{ACF}.
\end{aligned} \tag{2.18}$$

Equivalence for $\widehat{\mu}$

It follows directly from (2.18) that

$$\lim_{M\to\infty} \widehat{\mu}^{\mathrm{st}} = \bar{r}. \tag{2.19}$$

Now, let us consider the estimator $\widehat{\mu}^{\mathrm{ex}}$ (2.12):

$$\begin{aligned}
\widehat{\mu}^{\mathrm{ex}} &= \frac{R_c R_{mm} - R_m R_{cm}}{M(R_{mm} - R_{cm}) - R_m^2 + R_c R_m}\\
&= \frac{R_m(R_{mm} - R_{cm}) + R_{mm}(r_M - r_0)}{M(R_{mm} - R_{cm}) + R_m(r_M - r_0)}\\
&= \frac{\frac{R_m}{M}(\frac{R_{mm}}{M} - \frac{R_{cm}}{M}) + \frac{R_{mm}}{M}(\frac{r_M - r_0}{M})}{(\frac{R_{mm}}{M} - \frac{R_{cm}}{M}) + \frac{R_m}{M}(\frac{r_M - r_0}{M})}.
\end{aligned}$$

Because we analyze $\widehat{\mu}^{\mathrm{ex}}$ in the limit of $M \to \infty$, we first show that the denominator above does not go to 0 (using the properties in (2.18)):

$$\begin{aligned}
&\lim_{M\to\infty}\left(\left(\frac{R_{mm}}{M}-\frac{R_{cm}}{M}\right)+\frac{R_m}{M}\left(\frac{r_M-r_0}{M}\right)\right)\\
&\quad=\bar{r}^2+\mathbb{V}\mathrm{ar}(r)-\bar{r}^2-\mathbb{V}\mathrm{ar}(r)\,\mathrm{ACF}\\
&\quad=\mathbb{V}\mathrm{ar}(r)(1-\mathrm{ACF})>0,
\end{aligned}\tag{2.20}$$

which allows us to split the limit:

$$\begin{aligned}
\lim_{M\to\infty}\widehat{\mu}^{\mathrm{ex}}&=\frac{\lim\limits_{M\to\infty}\left(\frac{R_m}{M}\right)\lim\limits_{M\to\infty}\left(\frac{R_{mm}}{M}-\frac{R_{cm}}{M}\right)+\lim\limits_{M\to\infty}\left(\frac{R_{mm}}{M}\right)\lim\limits_{M\to\infty}\left(\frac{r_M-r_0}{M}\right)}{\lim\limits_{M\to\infty}\left(\frac{R_{mm}}{M}-\frac{R_{cm}}{M}\right)+\lim\limits_{M\to\infty}\left(\frac{R_m}{M}\right)\lim\limits_{M\to\infty}\left(\frac{r_M-r_0}{M}\right)}\\
&=\frac{\lim\limits_{M\to\infty}\left(\frac{R_m}{M}\right)\lim\limits_{M\to\infty}\left(\frac{R_{mm}}{M}-\frac{R_{cm}}{M}\right)}{\lim\limits_{M\to\infty}\left(\frac{R_{mm}}{M}-\frac{R_{cm}}{M}\right)}\\
&=\lim_{M\to\infty}\left(\frac{R_m}{M}\right)\\
&=\bar{r}.
\end{aligned}$$

And this, together with (2.19) proves:

$$\lim_{M\to\infty}\widehat{\mu}^{\mathrm{st}}=\lim_{M\to\infty}\widehat{\mu}^{\mathrm{ex}}.$$

Equivalence for θ

Directly from (2.18) we see that:

$$\lim_{M\to\infty}\delta t\widehat{\theta}^{\mathrm{st}}=1-\mathrm{ACF},$$

and because δt and M are independent, we even see that:

$$\lim_{\delta t\downarrow 0}\left(\lim_{M\to\infty}\widehat{\theta}^{\mathrm{st}}\right)=\lim_{\delta t\downarrow 0}\left(\delta t^{-1}(1-\mathrm{ACF})\right)=a.\tag{2.21}$$

Next, let us consider the following operations on the estimator $\widehat{\theta}^{\text{ex}}$ (2.12):

$$\begin{aligned} 1 - e^{-\widehat{\theta}^{\text{ex}}\delta t} &= \frac{R_{mm} - R_{cm} - \widehat{\mu}^{\text{ex}}(R_m - R_c)}{R_{mm} - 2\widehat{\mu}^{\text{ex}}R_m + M(\widehat{\mu}^{\text{ex}})^2} \\ &= \frac{\frac{R_{mm}}{M} - \frac{R_{cm}}{M} - \widehat{\mu}^{\text{ex}}(\frac{R_m}{M} - \frac{R_c}{M})}{\frac{R_{mm}}{M} - 2\widehat{\mu}^{\text{ex}}\frac{R_m}{M} + (\widehat{\mu}^{\text{ex}})^2}. \end{aligned}$$

Let us first show that the above denominator does not tend to 0 in the limit of $M \to \infty$.

$$\lim_{M\to\infty} \left(\frac{R_{mm}}{M} - 2\widehat{\mu}^{\text{ex}}\frac{R_m}{M} + (\widehat{\mu}^{\text{ex}})^2 \right) = \mathbb{V}\text{ar}(r) > 0. \tag{2.22}$$

This allows us to split the limit by using (2.20) and (2.22):

$$\begin{aligned} \lim_{M\to\infty} \left(1 - e^{-\widehat{\theta}^{\text{ex}}\delta t}\right) &= \frac{\mathbb{V}\text{ar}(r)(1 - \text{ACF})}{\mathbb{V}\text{ar}(r)} \\ &= 1 - \text{ACF}, \end{aligned} \tag{2.23}$$

which, since the log-function is continuous in $1 - \text{ACF}$ (because $-1 < \text{ACF} < 1$), results in:

$$\lim_{M\to\infty} \widehat{\theta}^{\text{ex}} = -\delta t^{-1}\log(\text{ACF}). \tag{2.24}$$

To evaluate (2.24) in the limit $\delta t \downarrow 0$, note that both $\log(\text{ACF})$ and δt go to 0 in the limit $\delta t \downarrow 0$, let us apply L'Hôspital's rule for one-sided limits:

$$\begin{aligned} \lim_{\delta t\downarrow 0} \left(\lim_{M\to\infty} \widehat{\theta}^{\text{ex}} \right) &= \lim_{\delta t\downarrow 0} \left(-\frac{d(\text{ACF})}{\text{ACF}\, d(\delta t)} \right) \\ &= \lim_{\delta t\downarrow 0} \left(-\delta t^{-1}(\text{ACF} - 1) \right) = a. \end{aligned} \tag{2.25}$$

Therefore, from (2.21) and (2.25), one concludes that:

$$\lim_{\delta t\downarrow 0} \lim_{M\to\infty} \widehat{\theta}^{\text{st}} = \lim_{\delta t\downarrow 0} \lim_{M\to\infty} \widehat{\theta}^{\text{ex}} = a. \tag{2.26}$$

Equivalence for σ

Directly from (2.18) we see that:

$$\lim_{M\to\infty} (\widehat{\sigma}^{\text{st}})^2 = 2\delta t^{-1}(1 - \text{ACF})\mathbb{V}\text{ar}(r).$$

Let us then use assumption 4(b) to arrive at:

$$\lim_{\delta t\downarrow 0}\lim_{M\to\infty}(\widehat{\sigma}^{\mathrm{st}})^2 = 2a\,\mathbb{V}\mathrm{ar}(r). \tag{2.27}$$

Now, we recall from section 3.3.1 the definition $\eta := \exp(-\widehat{\theta}^{\mathrm{ex}}\delta t)$, and rewrite the estimator $\widehat{\sigma}^{\mathrm{ex}}$ (2.12) to:

$$\begin{aligned}\lim_{M\to\infty}(\widehat{\sigma}^{\mathrm{ex}})^2 = \lim_{M\to\infty}(\frac{2\widehat{\theta}^{\mathrm{ex}}}{1-\eta^2}(\frac{R_{cc}}{M} - 2\eta\frac{R_{cm}}{M} + \eta^2\frac{R_{mm}}{M} \\ - 2\widehat{\mu}^{\mathrm{ex}}(\frac{R_c}{M} - \eta\frac{R_m}{M})(1-\eta) + (\widehat{\mu}^{\mathrm{ex}})^2(1-\eta)^2)).\end{aligned} \tag{2.28}$$

From (2.23) we know that:

$$\lim_{M\to\infty} 1-\eta^2 = 1-\mathrm{ACF}^2 > 0,$$

which shows that the denominator in (2.28) does not go to 0 in the limit $M\to\infty$, therefore we can split up the limit and look at each of the elements separately. Rigorous algebraic operations lead to the simplified equation:

$$\lim_{M\to\infty}(\widehat{\sigma}^{\mathrm{ex}})^2 = -2\delta t^{-1}\log(\mathrm{ACF})\mathbb{V}\mathrm{ar}(r),$$

to which we apply the limit $\delta t\downarrow 0$ (like in (2.25)) to finally obtain:

$$\lim_{\delta t\downarrow 0}\left(\lim_{M\to\infty}(\widehat{\sigma}^{\mathrm{ex}})^2\right) = -2a\,\mathbb{V}\mathrm{ar}(r). \tag{2.29}$$

Therefore, from (2.27) and (2.29), one concludes:

$$\lim_{\delta t\downarrow 0}\lim_{M\to\infty}(\widehat{\sigma}^{\mathrm{st}})^2 = \lim_{\delta t\downarrow 0}\lim_{M\to\infty}(\widehat{\sigma}^{\mathrm{ex}})^2. \tag{2.30}$$

B Explicit formulas for linearly fitted OU parameters

First, let us introduce the notations: $\eta_0 := \exp(-\widehat{\theta}_0^{\mathrm{lf}}\delta t)$, $Q_c = \sum_{i=1}^M q_i$, $Q_m = \sum_{i=1}^M q_{i-1}$, $Q_{cc} = \sum_{i=1}^M q_i^2$, $Q_{mm} = \sum_{i=1}^M q_{i-1}^2$, $Q_{cm} = \sum_{i=1}^M q_i q_{i-1}$, $X_{cc} = \sum_{i=1}^M r_i q_i$, $X_{mm} = \sum_{i=1}^M r_{i-1}q_{i-1}$, $X_{cm} = \sum_{i=1}^M r_i q_{i-1}$, $X_{mc} = \sum_{i=1}^M r_{i-1}q_i$ and

$$\begin{aligned}
A_0 &= Q_{mm}(R_c - R_m) - Q_m(X_{cm} - X_{mm}),\\
A_1 &= M(X_{cm} - X_{mm}) - Q_m(R_c - R_m),\\
B_0 &= R_m(X_{cm} + X_{mm}) + Q_m(R_{cm} - R_{mm}) - 2R_cX_{mm},\\
B_1 &= X_{mm}(R_c + R_m) - Q_m(R_{mm} - R_{cm}) - 2R_mX_{cm},\\
C_0 &= R_cR_{mm} - R_mR_{cm},\\
C_1 &= R_{mm}X_{cm} - R_{cm}X_{mm},\\
D_0 &= M(X_{cm} - X_{mm}) - Q_m(R_c - R_m)),\\
D_1 &= Q_{mm}(R_c - R_m) - Q_m(X_{cm} - X_{mm}),\\
E_0 &= M(R_{mm} - R_{cm}) + R_cR_m - R_m^2,\\
E_1 &= Q_{mm}(R_{mm} - R_{cm}) - X_{mm}(X_{mm} - X_{cm}).
\end{aligned}$$

An explicit formula for $\widehat{\mu}_1^{\mathrm{lf}}$ is then given by the real root of the third order polynomial $(\widehat{\mu}_1^{\mathrm{lf}})^3P_3 + (\widehat{\mu}_1^{\mathrm{lf}})^2P_2 + \widehat{\mu}_1^{\mathrm{lf}}P_1 + P_0$, where the polynomial's coefficients are given by:

$$\begin{aligned}
P_3 =&A_0A_1B_0 + A_0B_1D_0 - A_0D_1E_0 - D_0^2E_1\\
P_2 =&A_1B_0^2 + 2A_0A_1C_0 + A_0B_1E_0 + B_0B_1D_0 + C_1D_0^2 - B_0D_1E_0-\\
&C_0D_0D_1 - 2D_0E_0E_1\\
P_1 =&2A_1B_0C_0 + B_0B_1E_0 + B_1C_0D_0 + 2C_1D_0E_0 - C_0D_1E_0 - E_0^2E_1\\
P_0 =&A_1C_0 + B_1C_0E_0 + C_1E_0^2.
\end{aligned}$$

After introducing the last notations $\Delta_0 = P_2^2 - 3P_3P_1$, $\Delta_1 = 2P_2^2 - 9P_3P_2P_1 + 27P_3^2P_0$ and $C = \sqrt[3]{\frac{\Delta_1 + \sqrt{\Delta_1^2 - 4\Delta_0^3}}{2}}$, one obtains the explicit formula for $\widehat{\mu}_1^{\mathrm{lf}}$ and sequentially obtainable solutions for $\widehat{\mu}_0^{\mathrm{lf}}$, $\widehat{\theta}_0^{\mathrm{lf}}$ and $\widehat{\sigma}_0^{\mathrm{lf}}$ in (2.17).

Chapter 3

Covariate-based stochastic parameterization of baroclinic ocean eddies

In this study we investigate a covariate-based stochastic approach to parameterize unresolved turbulent processes within a standard model of the idealised, wind-driven ocean circulation. We focus on vertical instead of horizontal coarse-graining, such that we avoid the subtle difficulties of horizontal coarse-graining. The corresponding eddy forcing is uniquely defined and has a clear physical interpretation related to baroclinic instability. We propose to emulate the baroclinic eddy forcing by sampling from the conditional probability distribution functions of the eddy forcing obtained from the baroclinic reference model data. These conditional probability distribution functions are approximated here by sampling uniformly from discrete reference values. We analyze in detail the different performances of the stochastic parameterization dependent on whether the eddy forcing is conditioned on a suitable flow-dependent covariate or on a time-lagged covariate or on both. The results demonstrate that our non-Gaussian, non-linear methodology is able to accurately reproduce the first four statistical moments and spatial/temporal correlations of the stream function, energetics, and enstrophy of the reference baroclinic model.

1 Introduction

1.1 Background and motivation

The large-scale ocean circulation is strongly influenced by mesoscale turbulent eddies [71, 108]. Baroclinic instability is the primary generating mechanism for mesoscale eddies in oceanic flows [110, 136, 142]. How accurately the impact of baroclinic instability is represented in ocean models depends on the accuracy of the baroclinic eddy forcing that appears in the equations of motion. Mesoscale ocean eddies exist on spatial scales roughly between $O(10\text{ km})$ and $O(100\text{ km})$. Therefore, global climate models need grid resolutions smaller than $O(10\text{ km})$ in their ocean component in order to directly resolve these turbulent motions. Due to computational limitations, such high resolution is still infeasible in current climate models, and the effects of turbulent eddies must be parameterized. Parameterizations here are understood to be parametric mathematical models to be applied to an ocean model with a spatial resolution that leaves eddy forcing partly unresolved.

Mesoscale eddy parameterizations are commonly formulated in a deterministic way, typically based on the Gent–McWilliams (GM) parameterization [43, 47, 62, 147]. Deterministic eddy parameterizations represent an approximation of the integrated effects of the unresolved processes in terms of the resolved scale flow. These parameterizations are motivated by the idea that the properties of the unresolved scale processes can be uniquely represented by the resolved scale states. However, in practice, the resolved states are associated with many possible unresolved states [16, 54]. Therefore, deterministic parameterizations can, at best, provide an ensemble-mean representation of the unresolved scale processes. To overcome the limitations of deterministic parameterizations, atmospheric research has in recent years started to focus on stochastic parameterizations [16, 55, 104, 116]. Examples in an atmospheric context relevant to our work include Markov Chain models to represent atmospheric convection [30, 39, 89]. Stochastic eddy parameterizations are a more recent development in oceanic research. Grooms and Majda [69] developed a new approach combining elements from superparameterization and stochastic parameterizations applicable to quasi-geostrophic turbulence. Cooper and Zanna [27] posed a linear stochastic term that stochastically parameterizes transient eddies in an idealized barotropic ocean gyre model. They suggested an efficient search method along parameter space that optimizes their parameters with respect to a reference climatological mean, variance, and decorrelation time scales of the horizontal flow velocities.

In the current work we explore how to use the novel data-driven stochastic methodology posed in Verheul and Crommelin [143] for eddy parameterization. The main assumption for our parameterization is that sample data from a 'high-resolution' ocean model is available. We use this sample data to approximate conditional probability distribution functions (CPDFs) for the mesoscale eddy forcing. By conditioning on appropriately chosen covariates, i.e. model variables that correlate significantly with the mesoscale eddy forcing, we define a flow-dependent parameterization that samples directly from the CPDFs. The main goal of this parameterization is to drive a reduced ocean model in such a way that the resulting stochastic model is able to emulate the dynamics produced by the full model.

Typically, in such a reduced model the vertical discretization of the 'high-resolution' ocean model is preserved, and the horizontal grid resolution is reduced, see e.g. [14, 84, 117]. While such a horizontal coarse-graining set-up preserves some of the properties induced by the vertical stratification, a clear definition of the associated mesoscale eddy forcing is difficult both numerically and physically. Numerically, one is faced with nontrivial filtering options [117, 146] that subtly change the definition of the eddy forcing. In turn, physical interpretations of such eddy forcings are non-transparent to a certain extent because horizontally coarse-grained terms mix partly resolved processes of both barotropic and baroclinic nature. To avoid such concerns, we focus instead on a 'vertical coarse-graining' set-up which preserves horizontal grid resolution, but reduces the vertical discretization [63, 81]. This less common approach enables a clear and unambiguous definition of the eddy forcing with the clear physical interpretation related solely to the baroclinic nature of the flow. The 'vertical coarse-graining' allows us to focus fully on the development of our stochastic methodology in this spatially extended setting without being detracted by the aforementioned concerns.

In this study, we aim to drive a reduced ocean model with flow-dependent as well as spatially and temporally correlated stochastic forcing. Recent related work using flow-dependent stochastic parameterizations include the following examples. Using the stochastic approach of [56], Kitsios et al. [91] parameterizes subgrid eddy-eddy interactions as a combination of deterministic eddy viscosity and stochastic backscatter eddy viscosity. Furthermore, in [91] they formulated scaling laws for the respective coefficients dependent on the resolution, enstrophy flux, Rossby radius, and energy range. Jansen and Held [84] modeled the amplitude of the flow-dependent energy source due to backscatter forcing with simple spatially uncorrelated Gaussian white noise. By combining a standard hyperviscous closure with this stochastic term they successfully returned the energy otherwise spuriously dissipated, as is typical for hyperviscous closures. Finally, some recent studies have focused on increasing the efficiency of superparameterization, an extremely expensive computational approach to parameterization that embeds high-resolution simulations in grid cells of low-resolution large scale simulations [66, 67, 87]. Majda and Grote [102] proposed a framework that models the small-scale dynamics with quasilinear stochastic partial differential equations, which was later implemented with success in Grooms and Majda [70] for a one-dimensional turbulent system. However, the feedback to the large scales was effectively non-stochastic in this implementation. Grooms and Majda [69] successfully used unidirectional plane waves in random directions for efficient computation of the flow-dependent Fourier integrals that determine the stochastic feedback to the large scales.

Most relevant to our proposed methodology are the studies by Berloff [11, 14], and by Zanna and colleagues [117, 160]. The goal in these studies is to model the spatio-temporal correlations of the ocean flow, a goal that we share here. Moreover, Zanna and colleagues employ a stochastic methodology based on the use of a covariate, again similar to what is proposed here. Regarding the former, Berloff [14] showed that the temporal correlations of a diagnosed eddy forcing can be reproduced by forcing a 'non-eddy-resolving' stationary double gyre ocean model with a simple flow-independent but spatially varying autoregressive process. This approach showed good results in reproducing the desired statistics in the stochastic model, however it required many parameters to be estimated. In Berloff et al. [11] this methodology was extended to model spatio-temporal correlations in a coupled ocean-atmosphere model. The low-frequency coupled variability in this system gave novel non-stationary statistical properties of the reference ocean eddy forcing. These properties were modeled in the stochastic forcing by introducing flow-dependency in the variance of the forcing (dependent on the baroclinicity of the ocean flow).

In Porta Mana and Zanna [117] it was proposed to reproduce spatio-temporal correlations by sampling from reference values of the eddy forcing. To achieve this, the CPDFs for the eddy forcing were approximated with Gaussian distributions, conditioned on a suitable covariate. The stochastic and deterministic feedback to a double gyre quasi-geostrophic ocean model using this covariate were explored in Zanna et al. [160]. This parameterization drastically improved the mean state and variability of the ocean state. While similar in design philosophy to our work here, we note some important differences.

Firstly, Zanna and colleagues [117, 160] develop their methodology in a set-up of horizontal coarse-graining, with the related difficulties discussed earlier in this introduction. Secondly, the covariate specified in [117, 160] is based on temporal tendencies of the vorticity, however these tendencies are in turn dependent on the unresolved eddy forcing. To close the system, the temporal tendencies must be approximated. By contrast, in the approach developed here we use resolved model variables and lagged self-conditioning to formulate our parameterization. Thirdly, this lagged self-conditioning allows us to explicitly represent temporal correlations in the parameterization. This feature is implicitly included only with respect to the sampling interval in [117, 160]. Fourthly, whereas in [117, 160] the CPDFs for the eddy forcing are approximated with Gaussian distributions, we assume no underlying distribution. Instead, we sample directly from the CPDFs as described by the sample data. Fifthly, and lastly, while the covariate used in [117, 160] is motivated physically as well as justified numerically, the parameterization concerns a single covariate. Therefore, all dynamical effects on the ocean flow are attributed to that one covariate. We aim to make the dynamical effects of our covariates intuitive and transparent by using multiple simple covariates. This allows us to perform sensitivity analyses, as well as compare between two-way coupled simulations with different configurations to illustrate the differences between covariates.

The presentation of this work is as follows: in the remainder of this section we present a formal problem description. In Section 2 we define the physical ocean model. The stochastic model and methodology are detailed in Section 3. Finally, different choices for stochastic models and the accompanying results are discussed in Sections 4 and 5, respectively.

1.2 Problem description.

An ocean-climate model consists of coupled partial differential equations (PDEs) resulting from a set of conservation laws [36, 110]. In an abstract description of an ocean model a state vector $\boldsymbol{u}$ is evolved over time in response to some external forcing $\mathcal{F}$, a linear operator $\mathcal{L}\boldsymbol{u}$ and some non-linear operator $\mathcal{N}(\boldsymbol{u})$. Without loss of generality, we consider the state vector to consist of two orthogonal modes $\boldsymbol{u} := (u_0, u_1)$. The coupled PDEs with quadratic $\mathcal{N}$ can then be written as:

$$\begin{aligned}\partial_t u_0 &= \mathcal{F}_0 + \mathcal{L}_0 u_0 + \mathcal{N}_{00}(u_0, u_0) + \mathcal{N}_{01}(u_0, u_1)\\ \partial_t u_1 &= \mathcal{F}_1 + \mathcal{L}_1 u_1 + \mathcal{N}_{11}(u_1, u_1) + \mathcal{N}_{10}(u_0, u_1),\end{aligned}$$

where $\mathcal{N}_{00}, \mathcal{N}_{11}$ indicate the nonlinear self-interaction of each mode, and $\mathcal{N}_{01}, \mathcal{N}_{10}$ represent the nonlinear coupling of the different modes.

Next, we consider a reduced ocean model where only the variable u_0 is evolved. To distinguish it from the u_0 in the coupled model above, we denote it as $\widetilde{u}_0$ in the reduced model in the following. Without parameterization to compensate for the missing term $\mathcal{N}_{01}$, the dynamics of the reduced model can differ significantly from the dynamics of u_0 in the full, coupled model. The stochastic approach explored in this study aims to remedy this shortcoming by driving the reduced model with a stochastic process $\widetilde{\boldsymbol{R}}$, i.e. to define a stochastic reduced model:

$$\partial_t \widetilde{u}_0 = \mathcal{F}_0 + \mathcal{L}_0 \widetilde{u}_0 + \mathcal{N}_{00}(\widetilde{u}_0, \widetilde{u}_0) + \widetilde{R}.$$

The main objective of our work is to choose the stochastic process $\widetilde{\boldsymbol{R}}$, the so-called stochastic eddy forcing, in such a way that both the long-term statistical behavior and the physical properties of $\widetilde{u}_0$ resemble those of u_0. Hence, the criteria that we use to assess the accuracy of our stochastic reproduction are the field's first four statistical sample moments, the autocorrelations, and spatial covariances and correlations, as well as the energetics and enstrophy (see Section D for formal definitions of these quantities).

2 Physical model

In this study, we consider a standard model of idealised ocean dynamics, namely, quasi-geostrophic (QG), potential-vorticity (PV) equations in a classical double-gyre configuration (see e.g. Vallis [142]). The fluid-dynamic model describes idealised, wind-driven midlatitude ocean circulation with prescribed density stratification in a flat-bottom square basin with north-south and east-west boundaries. We employ a set-up in which the vertical discretisation is done by projection onto the two leading vertical eigenmodes (see e.g. [52, 63, 81]), i.e. the barotropic mode and the first baroclinic mode. The potential vorticity conservation for the barotropic (baroclinic) mode stream function ψ_0 (ψ_1) with rigid lid vertical boundary conditions then reads:

$$\begin{aligned}\partial_t q_0 + J(\psi_0, q_0) + R + \beta\partial_x\psi_0 &= A_H\nabla^4\psi_0 + \frac{\partial_x\tau^y - \partial_y\tau^x}{\overline{\rho}H}\ , \\ \partial_t q_1 + J(\psi_1, q_0) + J(\psi_0, q_1) + \epsilon_{111}J(\psi_1, q_1) + \beta\partial_x\psi_1 &= \\ A_H\nabla^4\psi_1 + \frac{\gamma(\partial_x\tau^y - \partial_y\tau^x)}{\overline{\rho}H}\ , \end{aligned} \tag{2M}$$

where $J(f,g) := (\partial_x f)(\partial_y g) - (\partial_y f)(\partial_x g)$, the relative PVs are given by $q_0 = \nabla^2\psi_0$ and $q_1 = \nabla^2\psi_1 - \lambda^{-2}\psi_1$, respectively, and R is defined as:

$$R(x, y, t) := J(\psi_1(x, y, t), q_1(x, y, t)). \tag{3.1}$$

The term R acts as the feedback of the baroclinic mode on the barotropic mode, and is interpreted as baroclinic eddy forcing term in this study. In the following, we denote the 2-mode model with 2M, and we refer to the first equation of 2M with a stochastic representation of R as the stochastic 1-mode model (S1M, see Section 3), and to the first equation of 2M with R set to zero as the deterministic 1-mode model (D1M):

$$\partial_t q_0 + J(\psi_0, q_0) + \beta\partial_x\psi_0 = A_H\nabla^4\psi_0 + \frac{\partial_x\tau^y - \partial_y\tau^x}{\overline{\rho}H}\ . \tag{D1M}$$

The code for above deterministic models is part of OMUSE [115] and is available at the OMUSE project website: `https://bitbucket.org/omuse`. In our numerical model simulations, the flow is driven at the surface by the asymmetric double-gyre zonal wind stress (as e.g. in [14]),

$$\tau^x(y) = \tau_0 \left[\cos\left(\frac{2\pi(y - L/2)}{L} \right) + 2 \sin\left(\frac{\pi(y - L/2)}{L} \right) \right] , \qquad \tau^y = 0 \ ,$$

where $\tau_0 = 0.05$ Nm^{-2}, and $L = 4000$ km is the size of the square basin with $0 \leq x, y \leq L$. The first internal Rossby radius of deformation, λ, represents a length scale of baroclinic eddies and is set to $\lambda = 50$ km, a typical value for the midlatitude ocean circulation. We use an eddy-resolving horizontal resolution of 10 km in our numerical simulations with a correspondingly small lateral viscosity coefficient, $A_H = 100$ m^{2}s^{-1}, as well as free-slip boundary conditions. Furthermore, we use typical values for the mean ocean depth, $H = 4000$ m, the mean density of sea water, $\overline{\rho} = 1000$ kgm^{-3}, and the meridional variation of the Coriolis parameter, $\beta = 1.8616 \times 10^{-11}$m^{-1}s^{-1}. Finally, we consider the idealized case of constant stratification such that $\epsilon_{111} = 0$ and $\gamma = \sqrt{2}$ (see e.g. Hua and Haidvogel [81]). All numerical model simulations in this work have a spin-up time of 30 years and an integration length of another 30 years. See Table 3.1 for an overview of parameter values used in this study.

Figure 3.1 shows snapshots as well as the temporal averages $\mu(H\psi_0)$ and standard deviations std$(H\psi_0)$ of the barotropic stream function ψ_0 in Sverdrup (1 Sv $= 10^6$ m^{3}s^{-1}) for 2M and D1M, respectively. The statistical quantities are calculated from simulation results stored on a weekly basis. For both models the time-mean flow (see Figures 3.1b and 3.1e) consists of the southern (subtropical) and northern (subpolar) gyres that fill about 2/3 and 1/3 of the basin, respectively, which is consistent with the wind stress pattern. In the eastern part of the basin the flow is characterized by the linear Sverdrup balance which leads to essentially identical time-mean flow fields for both 2M and D1M. Near the western boundary, on the other hand, narrow boundary currents close the circulation and nonlinear terms are significant. The magnitude and the meridional extension of the time-mean western boundary currents are significantly larger for 2M than for D1M. In terms of fluctuations, for both models the basin can be partitioned into the more energetic western boundary region, characterized by strong vortices, and the less energetic eastern part, dominated by the planetary waves (see [15] for details). However, in addition to the strengthened time-mean flow, the variability is significantly more pronounced as well for 2M, as visible in both snapshots (Figures 3.1a and 3.1d) and standard deviation fields (Figures 3.1c and 3.1f). In particular, significantly larger and stronger vortices are present at the western boundary. Variability is also dominant in the rest of the basin, whereas for D1M the instantaneous flow pattern in the eastern part largely resembles the time-mean flow pattern. Finally, Figures 3.2a and 3.2b show corresponding time series of kinetic energy in joules (1 J = 1 kgm^{2}s^{-2}) and enstrophy in kgs^{-2} which again demonstrate larger mean values and stronger variability for 2M than for D1M.

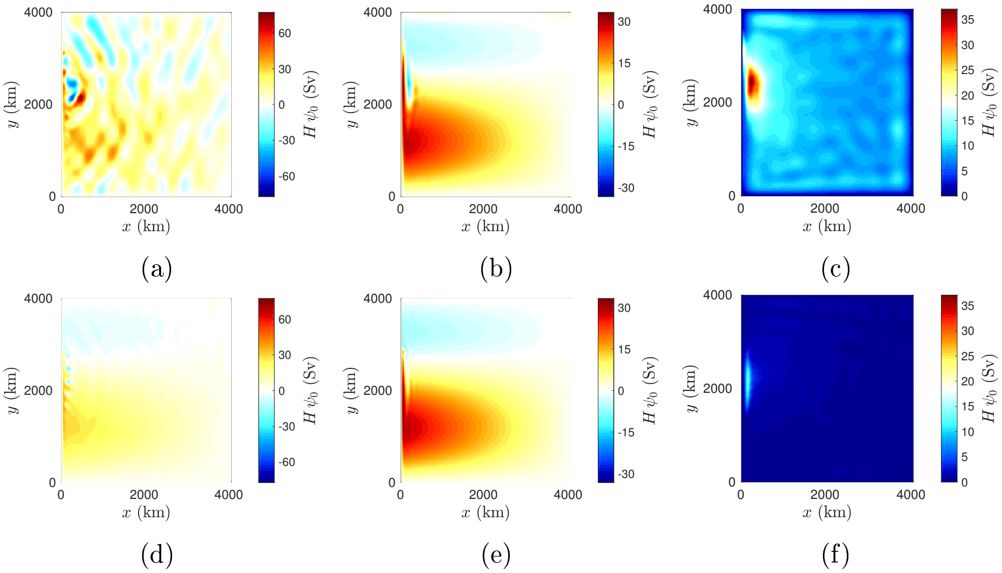


Figure 3.1: Properties of ψ_0 in 2M/D1M: (a)/(d) snapshot, (b)/(e) pointwise temporal mean, (c)/(f) pointwise temporal standard deviation.

We consider 2M as a minimal model that captures the main barotropic and baroclinic processes of interest, as well as the interactions between these dynamical processes. Notwithstanding, it is clear that 2M is strongly idealized, because of e.g. the assumption of QG dynamics, the idealized square basin geometry and the coarse vertical discretization with only two vertical modes. The model could be made more realistic, e.g. by increasing the number of vertical modes, by including a vertically dependent stratification, or by applying different boundary conditions. In particular, the relatively small eastward jet extension (related to the boundary conditions and stratification, see Figure 3.1) is not very realistic. However, the model allows for straightforward implementation of our stochastic modeling approach, as we discuss in the next section. Therefore, we consider it an appropriate test model for the purposes of developing the stochastic methodology from Verheul and Crommelin [143] in the setting of a spatially extended model.

As already mentioned in the introduction, in this study we focus on model reduction by vertical coarse-graining instead of horizontal coarse-graining. We aim to formulate a reduced model for the barotropic stream function and vorticity, with a stochastic representation of the baroclinic feedback R. Here, it amounts to reducing the number of degrees of freedom by a factor of 2, a modest reduction compared to what can be achieved with the more commonly pursued horizontal coarse-graining. We point out that it is straightforward to generalize to a higher number of vertical modes, by changing the definition of R in (3.1) to the sum of nonlinear feedback terms on the barotropic mode, $\sum_i J(\psi_i, q_i)$. For a recent investigation into the roles of the different individual baroclinic modes and their interaction, see Shevchenko and Berloff [131]. In our context the entire effect of baroclinicity is reduced to the one baroclinic eddy forcing of the barotropic mode (for any number of baroclinic modes in the reference model). Consequently, our approach is formally unaffected by more baroclinic modes in the reference model. Finally, Shevchenko and Berloff [131] reports mainly quantitative changes due to the inclusion of more baroclinic modes with the overall eddy effects remaining qualitatively similar. We speculate that the same will hold for the baroclinic eddy forcing and its stochastic parameterization.

More importantly, with vertical coarse-graining the full (baroclinic) model and the reduced model are both discretized on the same horizontal grid, so that R purely represents the effects of baroclinic instability. In this way, we avoid the conceptual difficulties of filtering (coarse-graining) fields that are discretized on a lattice, as also discussed in the introduction. R has a clear physical interpretation and we do not have to disentangle physical effects of unresolved processes from, e.g, grid transfer effects and reduced accuracy of discretized differential operators.

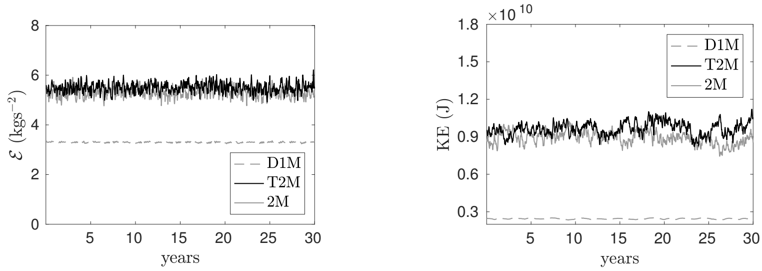
Table 3.1: Parameter settings for all modal ocean numerical models (2M, D1M, S1M)

Parameter	Explanation	Value
α	Robert–Asselin filter parameter	0.1
β	rate of Coriolis change	1.8616×10^{-11} m^{-1}s^{-1}
A_H	lateral viscosity coefficient	10^2 m^{2}s^{-1}
$\overline{\rho}$	mean density	10^3 kgm^{-3}
H	ocean depth	4×10^3 m
τ_0	magnitude of wind-forcing	5×10^{-2} Nm^{-2}
ϵ_{111}	triple interaction coefficient	0
$\varphi_1(z = 0)$	surface eigenfunction	$\sqrt{2}$
λ	first Rossby radius of deformation	5×10^4 m
Δx	horizontal grid spacing x-direction	10^4 m
Δy	horizontal grid spacing y-direction	10^4 m
L_x	horizontal domain size x-direction	4×10^6 m
L_y	horizontal domain size y-direction	4×10^6 m
N_x	number of grid points x-direction	401
N_y	number of grid points y-direction	401
Δt	integration time step	1.8×10^3 s
δt	sampling interval	1.8×10^3 s
T_c	conditioning interval	9.43488×10^8 s
T_s	spin-up time	9.43488×10^8 s
T	integration time	9.43488×10^8 s
N_B	number of bins per conditioning variable	5

2.1 Spatial structure and restriction of the eddy forcing

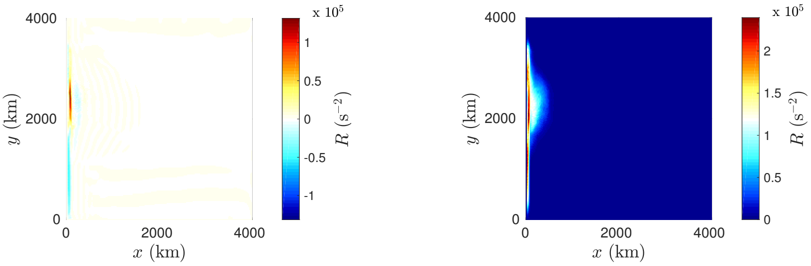
Figures 3.3a and 3.3b show the temporal average $\mu(R)$ and standard deviation $\mathrm{std}(R)$ of the eddy forcing R as diagnosed from 2M. Both fields exhibit the same order of magnitude and are essentially confined to a narrow band at the western boundary. Within this region, $\mu(R)$ exhibits a dipole structure with negative values in the southern half and positive values towards the north. The two local extrema in $\mu(R)$ correspond to two local maxima in $\mathrm{std}(R)$.

The spatial structure of R suggests that it might be sufficient to model R within only a subdomain at the western boundary instead of the entire basin. In order to test this, we performed a ‘truncated’ 2-mode model simulation, T2M, which is identical to 2M except that R is set to zero outside the western



(a) (b)

Figure 3.2: Comparison of scalar physical properties between the deterministic models: (a) enstrophy, (b) kinetic energy.



(a) (b)

Figure 3.3: Properties of R in the reference 2M: (a) pointwise temporal mean, (b) pointwise temporal standard deviation.

boundary region WB $= [10, 490] \times [500, 3490]$ km. Figure 3.2 also shows the corresponding time series of enstrophy and the kinetic energy for T2M. The mean levels as well as both the short and long-term variability of enstrophy and kinetic energy are similar for T2M and 2M (the same holds when comparing the spatial fields shown in Figures 3.1a–3.1c with corresponding T2M fields, not shown) which indicates that T2M and 2M essentially produce the same flow dynamics for our model set-up. Consequently, we will focus on modeling R restricted to WB in the following; this has the practical advantage of reducing the volume of data that must be handled in our stochastic modeling of R (see Section 3.4).

3 Stochastic model

The goal of this work is to formulate a stochastic process $\widetilde{\boldsymbol{R}}$ that emulates the eddy forcing $\boldsymbol{R}$ (see (3.1)). Adding this stochastic eddy forcing to D1M results in the stochastic 1-mode model (S1M):

$$\partial_t \widetilde{q}_0 + J(\widetilde{\psi}_0, \widetilde{q}_0) + \widetilde{R} + \beta \partial_x \widetilde{\psi}_0 = A_H \nabla^4 \widetilde{\psi}_0 + \frac{\partial_x \tau^y - \partial_y \tau^x}{\overline{\rho} H}. \tag{S1M}$$

Recall that throughout this work we compare variables from deterministic models (e.g. ψ_0) with their counterparts in a stochastic model (e.g. $\widetilde{\psi}_0$), and we use the same symbols for both but emphasize the difference with a tilde.

3.1 Conditioning procedure

To close the system S1M, a model that describes the temporal evolution of $\widetilde{R}$ is needed. We model $\widetilde{R}$ as a stochastic process, following the approach discussed in Verheul and Crommelin [143]. This approach is a form of resampling, in which $\widetilde{R}$ is sampled uniformly from conditioned observed values of R. However, whereas in [143] we considered a situation in which R was a scalar quantity, here we are dealing with a spatially extended system in which R is a two-dimensional field. Therefore, we extend the method from [143] to a multidimensional setting, and apply it pointwise to sample the field $\widetilde{R}$. In this extension, we preserve the modular design philosophy behind the stochastic methodology, as well as the ability to represent non-linear and non-Gaussian behavior.

Our stochastic methodology makes use of sample data $(\boldsymbol{\Psi}, \boldsymbol{Q}, \boldsymbol{R})$ from the full model 2M. As follows from (3.1), let us write $R^n := J(\psi_1(x, y, n\Delta t), q_1(x, y, n\Delta t))$ and $\boldsymbol{R} = (R^0, \ldots, R^N)$ to denote the time series of R resulting from a 2M simulation. Let $R^n(i,j)$ denote the eddy forcing value in grid point (i,j) at time n. The stochastic forcing $\widetilde{R}$ in S1M is then sampled pointwise from sample data R conditioned on two types of covariates: time-lagged R-values $\widetilde{R}^{n-l_\theta(i,j)}(i,j)$ and a flow-dependent model variable $C(\widetilde{\psi}_0^n)(i,j)$:

$$\begin{aligned} \widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid (R^{n-l_\theta(i,j)}(i,j) = \widetilde{R}^{n-l_\theta(i,j)}(i,j), \\ C(\psi_0^n)(i,j) = C(\widetilde{\psi}_0^n)(i,j)), \quad \forall i,j \in \mathrm{WB}, \end{aligned}$$

or for short:

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid (\widetilde{R}^{n-l_\theta(i,j)}(i,j), C(\widetilde{\psi}_0^n)(i,j)), \quad \forall i,j \in \mathrm{WB}, \qquad (3.2)$$

where $l_\theta(i,j)+1$ denotes the conditioning time lag for grid point (i,j) (see Section 3.2), and $C(\widetilde{\psi}_0^n)(i,j)$ is a function of $\widetilde{\psi}_0^n$ (see Section 3.3). Also, the stochastic forcing is generated only in the region WB and considered zero outside of this region (see Section 2.1).

Intuitively, the formulation in (3.2) says that the stochastic values are sampled from precisely those reference R-values that occurred in 2M when certain model variables in 2M matched the values of the corresponding variables in S1M. The distributions $R^{n+1}(i,j) \mid (\widetilde{R}^{n-l_\theta(i,j)}(i,j), C(\widetilde{\psi}_0^n)(i,j))$ are usually not known, therefore we approximate them with a simple binning method (see Section 3.4).

An important detail that we highlight is that the sampling interval used in the conditioning procedure (3.2) equals the integration time step (half an hour, see Table 3.1). This allows our parameterization a level of detail not usually seen in stochastic climate models, but this does come at a cost in the form of considerable memory requirements.

Crucially, the temporal evolution of $\widetilde{\boldsymbol{R}}$ is governed by sampling from the CPDFs in (3.2). Obviously, the set of conditioning model variables could be chosen arbitrarily large, making this methodology easily extendable. But for the sake of simplicity, and the test cases we discuss here, we consider at most one of each covariates: a single function $C(\psi_0)$ and a single lagged value of R. The set of conditioning variables is denoted $\{R^{n-l_\theta}, C(\psi_0))\}$. Different choices for $l_\theta(i,j)$ and $C(\psi_0^n)(i,j)$ give different sampling methods, and, in turn, different discrete stochastic processes $\widetilde{R}$. These stochastic processes together with S1M define several stochastic models. In the following we discuss several sets of conditioning model variables, and we denote each stochastic model by their variable choices, i.e. S1M-$R[l_\theta]\,C$. For example, if one chooses $l_\theta(i,j) = 0, \{C(\psi^n)(i,j)\} = \emptyset$, then S1M-$R[0]$ describes the 1-mode model S1M forced by a simple time-correlated stochastic process (see (3.2)).

3.2 Time-lagged covariate

An important criterion for our stochastic simulations is the reproduction of the autocorrelations of ψ_0 exhibited in the full model 2M. To reproduce the temporal correlations of R we condition the CPDFs in (3.2) on temporally lagged values of R, i.e. $R^{n-l_\theta(i,j)}$ for some $l_\theta(i,j) \geq 0$ (see (3.2)). The choice for lag times relates to an interesting, but different, question entirely: if one could sample the stochastic term $\widetilde{R}^{n+1}$ from the conditioned probability distribution $R^{n+1} \mid \widetilde{R}^n, \ldots, \widetilde{R}^{n-L}$, how large does L need to be to accurately reproduce temporal correlations shown in $(\boldsymbol{\Psi}, \boldsymbol{Q}, \boldsymbol{R})$? This question can be phrased intuitively as: how much information of the history of the stochastic process is sufficient for our conditioning procedure. While this is an interesting problem, we consider this investigation outside of the scope of this paper, and take a heuristic approach.

We condition the CPDFs in (3.2) on a single lagged $\widetilde{R}$. We consider temporal decorrelation for each grid point to be the time at which the autocorrelation function first drops below the threshold $\theta = e^{-1}$. Figure 3.4 shows that the decorrelation time of R varies widely over the grid, i.e. anywhere from a day near the western boundary to 10 weeks around the eastward jet. Therefore, we expect the need to choose a pointwise lagged time l_θ in the conditioning procedure (3.2), i.e. to define a spatially dependent lag time $l_\theta(i,j)$. In Sections 4.1 and 4.3 we show results from stochastic simulations conditioning with lag times constant over the grid, as well as with spatially varying $l_\theta(i,j)$.

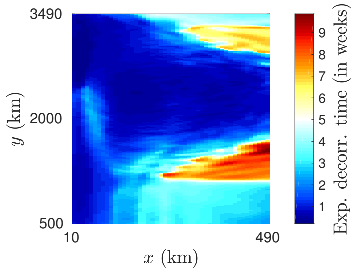


Figure 3.4: Pointwise decorrelation time (in days) of R over the region WB.

3.3 Flow-dependent covariate

Ideally, one conditions the CPDFs in (3.2) on resolved covariates, i.e. resolved model variables (in both D1M and 2M) that correlate strongly with the eddy forcing. The existence of such model variables in any multiscale model is not guaranteed. However, several studies have investigated and proposed candidates in related climate models (e.g. Porta Mana and Zanna [117]). In this section we choose a set of covariate candidates from the different terms constituting the PV budget. Expressing the eddy forcing R in 2M in terms of the other model variables gives:

$$R = -\partial_t \nabla^2 \psi_0 - J(\psi_0, q_0) - \beta \partial_x \psi_0 + A_H \nabla^4 \psi_0 + \frac{\partial_x \tau^y - \partial_y \tau^x}{\bar{\rho} H}. \tag{3.3}$$

This means that the eddy forcing R is expressed directly as a linear combination of each of the terms in the right-hand-side equation (3.3). However, unlike [117], we don't consider $\partial_t \nabla^2 \psi_0$ as a covariate candidate, because this is an unresolved term dependent on R itself. Each of the resolved terms in (3.3) are natural candidates for our covariate selection procedure. Therefore, we define the set of covariate candidates V as:

$$V := \{\psi_0, q_0, \beta\partial_x\psi_0, J(\psi_0, q_0), A_H \nabla^4 \psi_0, \frac{\partial_x \tau^y - \partial_y \tau^x}{\bar{\rho} H}\}.$$

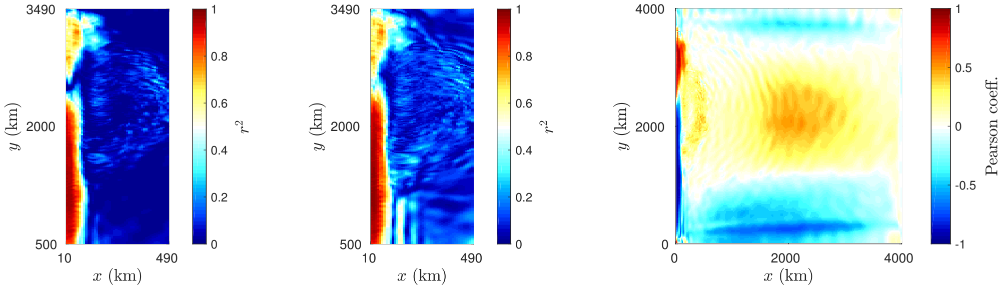
We use linear regression analysis and standard Pearson coefficient plots to assess these candidate covariates. For the regression analysis all model candidate variables in V are considered the regressors, and R the response variable, respectively. The r^2-value (not to be confused with the eddy forcing R) for a given linear regression is a statistical quantity for the percentage of the response variable's variability that is 'explained' by the covariates. While a high r^2-value indicates a good regression fit, it by no means guarantees the best covariate. Therefore, we make the following observations associated with r^2-values only to compare different sets of covariates, and not to 'prove' quality.

Our pointwise regression analysis shows that from all variables in the set V, it is the Jacobian $J(\psi_0, q_0)$ that explains most of the variability of R. This can be seen by comparing two different regressions, first between the entire set V and R, second between $J(\psi_0, q_0)$ and R. Figures 3.5a and 3.5b show the pointwise

r^2-values for these two regressions. Comparing the two different plots, one sees near identical r^2-values. This strongly indicates that the other candidates provide hardly any additional information for the regression. However, while the Jacobian shows the highest r^2-values, we note that it is far from a perfect predictor, as the central and eastern regions of WB remain badly represented.

Let us next compare the point wise Pearson correlation coefficients $p_{X,Y} = \mathbb{C}ov(X,Y)(\mathrm{std}_X \mathrm{std}_Y)^{-1}$, where $\mathbb{C}ov(X,Y)$ denotes the covariance between X and Y, and std_X and std_Y the standard deviations of X and Y respectively. Comparing each candidate in V coupled with R, we confirm the previous assessments that $J(\psi_0^n, q_0^n)$ correlates significantly with R, see Figure 3.5c. The highest correlation found between the Jacobian and R is located within the region WB described in Section 2.1. Other candidates in V show either significantly lower correlation to R or significantly lower r^2-values (not shown).

In addition to the statistical analysis above, our intuitive understanding for why the Jacobian is the most suitable covariate is that the Jacobian is the only resolved scale-coupling term in (3.3), i.e. term that is dependent on both small and large scale vortices. Therefore, despite the relatively low r^2-values in large parts of WB, $J(\psi_0, q_0)$ is our choice for covariate out of the tested candidates, and we choose $C(\widetilde{\psi}_0^n)(i,j) := J(\widetilde{\psi}_0^n(i,j), \widetilde{q}_0^n(i,j))$ in the following sections.



(a) (b) (c)

Figure 3.5: Covariate selection criteria. r^2-values for each grid point in $[10, 490] \times [500, 3490]$ km for pointwise regression analysis: (a) with regressor $J(\psi_0, q_0)$, (b) with regressors V. (c) Pearson coefficient $J(\psi_0, q_0)$ per grid point.

3.4 Sampling from empirical distribution

We choose the set of conditioning variables to contain either one or both of the time-lagged R-values $\widetilde{R}^{n-l_\theta(i,j)}(i,j)$ (see Section 3.2) and the Jacobian $J(\widetilde{\psi}_0^n(i,j), \widetilde{q}_0^n(i,j))$ (see Section 3.3). Following the set-up from Verheul and Crommelin [143], we apply an equidistant binning procedure to approximate the CPDFs in conditioning procedure (3.2). We refer to uniformly drawing samples from these bins as sampling from the empirical distribution. To establish the binning associated with the chosen covariates, the range between the minimum and maximum of each covariate is independently partitioned in N_B equidistant intervals, with the outer intervals considered half-open. This partitioning results in disjoint bins, each of which describes a set of $R^{n+1}(i,j)$-values. Through this discretization, one obtains a mapping from conditioning variables to sets of $R^{n+1}(i,j)$-values. See Appendix B for further technical details on the binning procedure.

Because our stochastic sampling procedure (3.2) acts pointwise, excessive artificial spatial roughness can arise in the generated fields $\widetilde{R}$. To prevent this phenomenon, we apply Gaussian spatial smoothing to the stochastic fields, as detailed in Appendix C. This is an ad hoc way to promote spatial smoothness, but one we consider adequate for the scope of this work. We leave a more systematic way to generate smooth fields for a future study.

A limitation of discrete sampling methods is that there is no predetermined way of handling situations in which the values of the conditioning variables are outside of the ranges exhibited in the sample data. We refer to bins outside of the ranges of the sample data as empty bins. When conditioning on both covariates, the bins are projected onto the linear trend $\widehat{J}$ between $R^{n-l_\theta(i,j)}(i,j)$ and $J(\psi_0^n(i,j), q_0^n(i,j))$ to more efficiently use the available bins (see Appendix A).

Furthermore, during simulations of the S1M, the Jacobian is removed from the conditioning variables whenever it goes outside the range in the sample data. The same is done at grid points where the correlation between J and R is low in the sample data (see Figure 3.5c).

We point out the computational efficiency of this sampling procedure. To evolve our stochastic model for the eddy forcing over time, we only need to calculate from which bin to sample for each grid point, and then draw a uniform random sample from that bin from memory. For comparison, in, for example, the approach from Cooper and Zanna [27], also relying on availability of high-resolution data, a system of linear stochastic ordinary differential equations (SDE) must be evolved at each model time step, involving six parameters and two variables per model grid point (for the model grid used here, this would amount to an SDE with 320000 variables). Besides the computational cost of integrating the model in time, the cost of constructing the stochastic eddy forcing model (i.e. the 'training phase') can be substantial. For our approach, it involves simple binning of data, with negligible computational cost. By contrast, the approach from [27] requires an expensive optimization procedure involving many reduced model runs.

3.5 Emulated stochastic eddy forcing

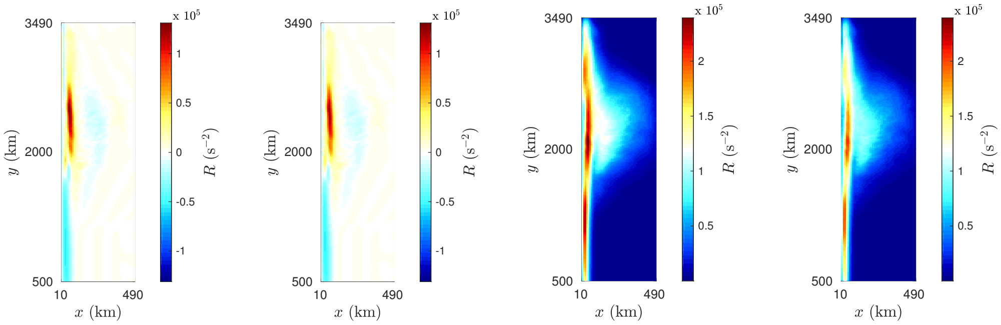
Similar to the investigations in [117], let us verify our stochastic methodology before coupling the stochastic process for R back to the reduced barotropic model. We use the output from a 2M simulation (i.e. sample data $\boldsymbol{J}$ and $\boldsymbol{R}$) to generate a so-called offline 'emulation' of the stochastic process. For the purposes of such an emulated process, let us choose $l = 0$:

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid (\widetilde{R}^{n}(i,j), J(\widetilde{\psi}_0^{n}(i,j), \widetilde{q}_0^{n}(i,j))), \quad \forall i,j \in \mathrm{WB}.$$

This stochastic term is then compared to the same reference eddy forcing sample data. While not directly testing our ultimate goal of driving a reduced model with stochastic forcing, this poses an interesting question in itself: can our methodology reproduce the statistical properties of the reference eddy forcing when the input to our conditional sampling method is known to be 'correct'? Thus, this test should be considered a verification of the consistency of our procedure, rather than a validation. As intuition would suggest, even this simple verification can fail if, e.g, the number of bins is chosen too small or if the conditioning variables are not effective predictors.

Individual snapshots over time of these emulated stochastic fields show very little error. The long-term statistics are shown in Figure 3.6, where one can see an accurately reproduced mean and standard deviation for the spatially filtered emulated stochastic forcing. We note that the spatial smoothing (see Section

C) decreases the standard deviation of $\widetilde{R}$ somewhat, compared to R. This should not be surprising, given that the variability of $\widetilde{R}$ is artificially smoothed out. However, based on our experiments, we consider the benefit from the spatial cohesion and smoothness of $\widetilde{R}$ as a result of this spatial smoothing more significant than the disadvantage of its decreased standard deviation.



(a) (b) (c) (d)

Figure 3.6: Comparison between pointwise temporal mean (a)-(b) and standard deviation (c)-(d) for the reference R (a) and (c) versus the filtered emulated $\widetilde{R}$ (b) and (d).

4 Results

The natural point of comparison between S1M and 2M is the evolution of the barotropic modes $\widetilde{\psi}_0$ and ψ_0. Therefore, we assess our stochastic parameterization by inspecting how well $\widetilde{\psi}_0$ reproduces the physical and statistical properties of ψ_0 in the reference 2M. The specific quantities we compare are: the enstrophy, kinetic energy, and energy transfer related to R (physical), and the statistical moments, and spatial and temporal correlations (statistical). See Appendix D for a reference of formal definitions for each of these.

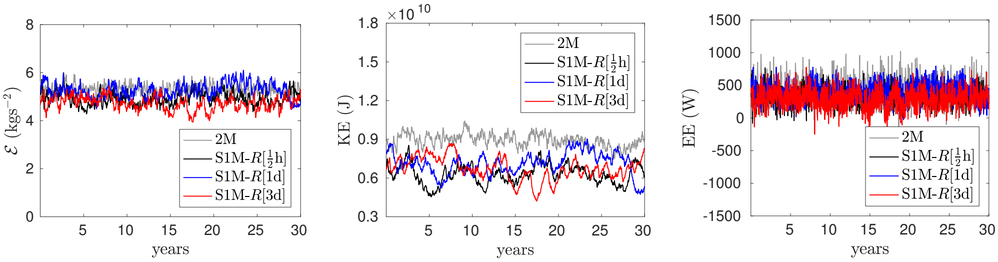
4.1 One-way coupling

Here, we consider a flow-independent sampling method, i.e. we consider only the time-lagged eddy forcing $R^{n-l_\theta(i,j)}$ as covariate and condition one-dimensionally: $R^{n+1} \mid \widetilde{R}^{n-l_\theta(i,j)}$. Let us start with a spatially constant $l_\theta(i,j) = l$:

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid \widetilde{R}^{n-l}(i,j) \quad \forall i,j \in \mathrm{WB}. \qquad (R[l])$$

While we will use such simulations to highlight the reasons that spatially dependent lag times are desired, they also result in some interesting observations. Let us discuss these results for three different choices for l (recall the choice for half an hour integration time-step, see Section 2): half an hour (S1M-$R[\frac{1}{2}\text{h}]$), a day (S1M-R[1d]), and three days (S1M-R[3d]).

These three reduced model simulations reproduce the mean barotropic stream function very well (not shown), with somewhat better results for longer time lags. The standard deviation of these barotropic stream functions (not shown) are a major improvement over D1M (Figure 3.1f), but are still significantly smaller than in 2M (Figure 3.1c). Figure 3.7a shows that the mean enstrophy (3.14) is also reproduced accurately for each of the lag time choices with a maximal error of 11% for S1M-R[3d] and an error of only 0.9% for S1M-R[1d]. For each of the lag options the enstrophy's variability is slightly overestimated compared to the 2M reference diagnostics. Both the kinetic energy (3.15) and the energy exchange term in watts (1 W = 1 Js^{-1}, 3.16) are significantly underestimated compared to 2M, as shown in Figures 3.7b and 3.7c, respectively. The mean kinetic energies in Figure 3.7b deviate between 20% and 30% from 2M, in addition to showing long excursions from their means, unlike what the reference diagnostic exhibits. The energy transfer means deviate between 2% (for S1M-R[1d]) and 29% (for S1M-R[3d]), but show standard deviations that are between 18% and 26.7% off from the reference values. However, we do consider these results to be remarkably good, considering the straightforward methodology that generated them.



(a) (b) (c)

Figure 3.7: Comparison of scalar physical properties between 2M, and S1M-$R[l]$ for different lags l: (a) enstrophy, (b) kinetic energy, (c) energy exchange term.

Figure 3.8 shows that similar plots result from spatially dependent lag times

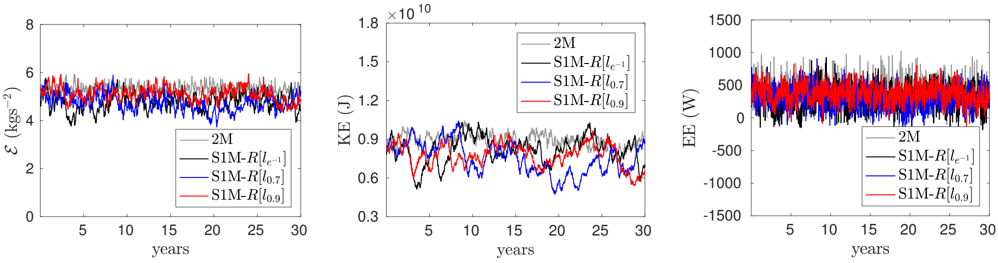
$l_\theta(i,j)$ (chosen equal to the time at which the autocorrelation function (ACF, 3.13) first crosses some threshold θ):

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid \widetilde{R}^{n-l_\theta(i,j)}(i,j) \quad \forall i,j \in \mathrm{WB}. \tag{$R[l_\theta]$}$$

All means and deviations of the stochastic barotropic stream function improve significantly, a trend illustrated in Table 3.2. The temporal means of the kinetic energy plots in Figure 3.8b improve compared to Figure 3.7b to an error between 9% and 16%. Aside from this result, Figure 3.8 indicates that spatially dependent lag times in S1M-$R[l_\theta]$ fail to significantly improve the diagnostic physical results. However, these results instead illustrate the limitations of flow-independent conditioning methods $R[l]$ and $R[l_\theta]$, as supported by later results in Section 4.3.

To illustrate the reproduction of temporal correlations with a scalar quantity we first consider the ACFs in the two grid points $[200, 2390]$ km and $[440, 3190]$ km in two different dynamical areas in region WB. The ACF plots in Figures 3.11 and 3.12 show that, unlike the energetics and enstrophy, a significant improvement can be seen when comparing the S1M-$R[l]$ simulations (Figure 3.11) with the S1M-$R[l_\theta]$ simulations (Figure 3.12). This should not be surprising given that the constant lag times chosen are shorter than the $l_\theta(i,j)$-values, even for $\theta = 0.9$. This means that the information added to the stochastic process by the process history is relatively insignificant, i.e. the lagged R-values are not significantly decorrelated.

The spatial covariances here are represented by the covariances between $\widetilde{\psi}_0$ in a central grid point and its surrounding grid points (we again choose both $[200, 2390]$ km and $[440, 3190]$ km as the two example points). Figures 3.10c and 3.10d show that the S1M-$R[\frac{1}{2}\mathrm{h}]$ very accurately reproduces the spatial structure of the reference covariances in Figures 3.10g and 3.10h, unlike the D1M covariances in Figures 3.10a and 3.10b. Note the significantly smaller magnitude for the barotropic references in Figures 3.10a and 3.10b, illustrating the strong improvement by each of the stochastic simulations over D1M. Quantitatively, however, the spatial covariances in these grid points (Figures 3.10c and 3.10d) are significantly underestimated compared to Figures 3.10g and 3.10h.



(a) (b) (c)

Figure 3.8: Comparison of scalar physical properties between 2M, and S1M-$R[l_\theta]$ for different spatial lag patterns l_θ: (a) enstrophy, (b) kinetic energy, (c) energy exchange term.

4.2 Two-way coupling: single conditioning variable

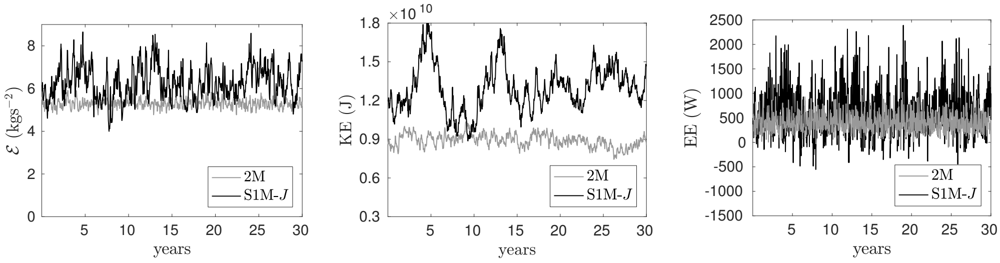
The results from Section 4.1 already show promise for the suggested methodology, but could use improvement when it comes to the scalar physical quantities and spatial covariances, see Figures 3.10c and 3.10d. We expect both aspects to improve by conditioning on the Jacobian, which simultaneously adds flow-dependency and implicitly represented spatial correlations to neighboring grid points (as discussed in Section 3.3):

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid J(\widetilde{\psi}_0^n(i,j), \widetilde{q}_0^n(i,j)) \quad \forall i,j \in \text{WB}. \tag{J}$$

Similar to the results in the previous section, S1M-J reproduces the mean barotropic stream function very well (not shown). However, it also significantly overestimates the standard deviation (not shown). Figure 3.9 shows that the comparisons to the flow-independent methods in Section 4.1 are unfavorable. The S1M-J simulation severely overestimates all of the previously considered physical quantities.

Comparing the ACF plots of $\widetilde{\psi}_0$ between S1M-J and S1M-$R[l]/R[l_\theta]$ simulations tells a similar story. Figures 3.11 and 3.12 show that the ACFs are reproduced much more accurately when conditioning on time lagged values for R. This difference is to be expected, given that the flow-dependent conditioning (J) does not involve the process' history.

In contrast to the covariance plots for D1M (Figures 3.10a and 3.10b), the covariance plots for S1M-J (Figures 3.10e and 3.10f) show the same spatial



(a) (b) (c)

Figure 3.9: Comparison of scalar physical properties between 2M, and S1M-J: (a) enstrophy, (b) kinetic energy, (c) energy exchange term (with wider range on y-axis).

structure of the covariances shown for 2M (Figures 3.10g and 3.10h). However, whereas the covariances for S1M-$R[\frac{1}{2}\text{h}]$ are significantly underestimated, the covariances for S1M-J are significantly overestimated.

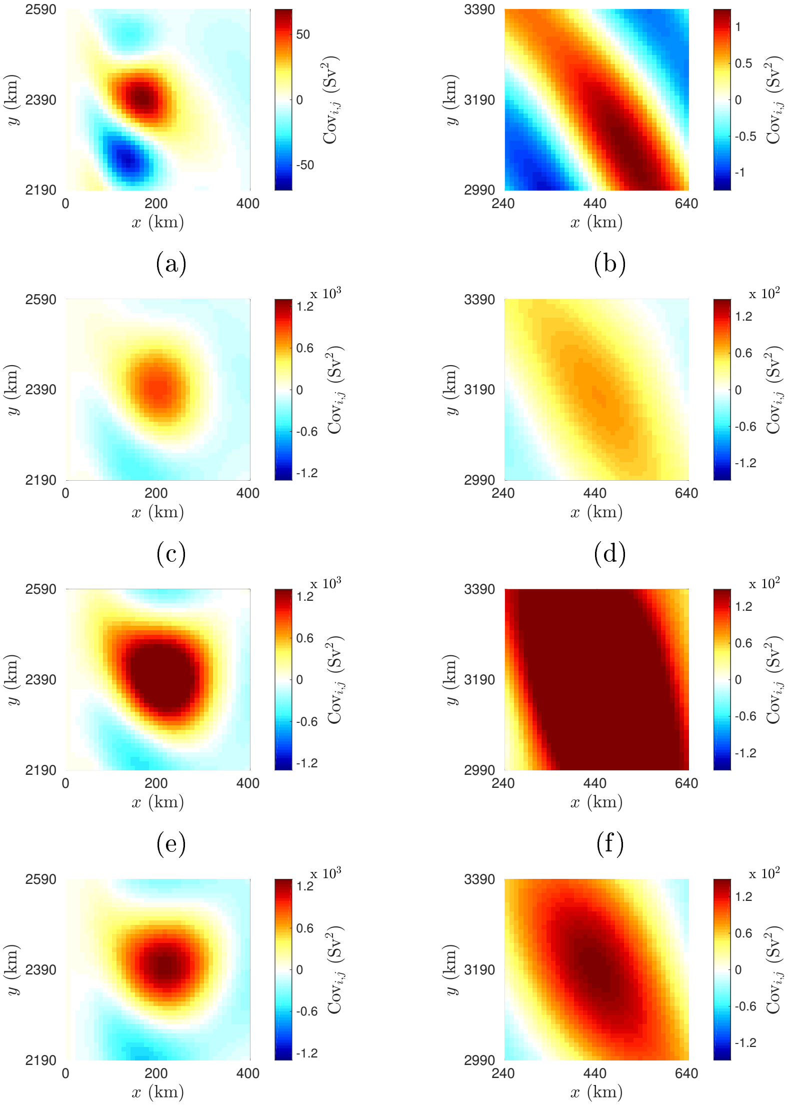
The overall conclusion from these tests is then that the Jacobian leads to overestimated amplitudes for most considered diagnostic criteria, i.e. enstrophy, energetics, spatial covariances, and standard deviation of ψ_0 (see Table 3.2), whereas the autocorrelations are underestimated. This further emphasizes our assessment that the Jacobian is far from a perfect predictor (as briefly discussed in Section 3.3). However, this flow-dependent spatially correlated driving force, albeit too erratic as sole conditioning variable, can improve the previously discussed flow-independent results.

4.3 Two-way coupling: double conditioning variables

By combining the aspects of the tests described in Sections 4.1 and 4.2, one arrives at the two-fold conditioning procedure, as described in (3.2), where $C(\psi_0^n)(i,j) = \widehat{J}(\psi_0^n(i,j), q_0^n(i,j))$ ($\widehat{J}$ denotes the linearly fitted J, see Section A).

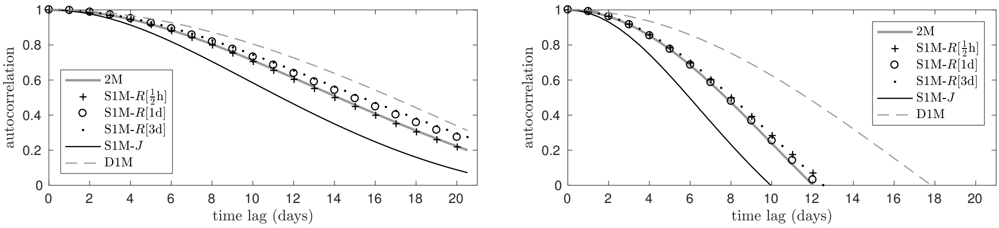
Similar to the tests in Section 4.1, let us first consider the simulations that condition on lagged R-values with constant lag l:

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid (\widetilde{R}^{n-l}(i,j), \widehat{J}(\widetilde{\psi}_0^n(i,j),\widetilde{q}_0^n(i,j))), \quad \forall i,j \in \text{WB}. \qquad (R[l]\,\widehat{J})$$



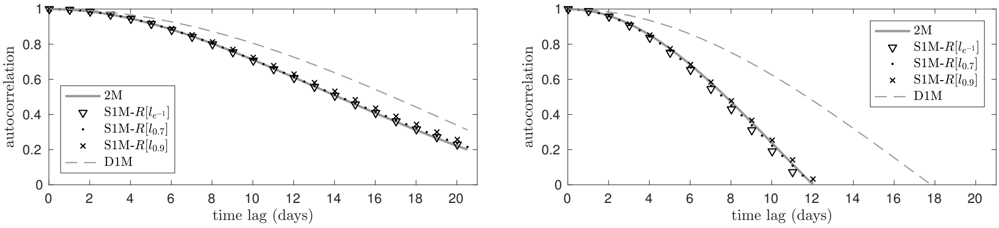
(a) (b) (c) (d) (e) (f) (g) (h)

Figure 3.10: Covariance plots for ψ_0 between grid point $[200, 2390]$ km/$[440, 3190]$ km and neighbouring grid points (see (3.17)) for each of the following models: (a)/(e) D1M, (b)/(f) S1M-$R[\frac{1}{2}\mathrm{h}]$, (c)/(g) S1M-J, (d)/(h) 2M.



(a) (b)

Figure 3.11: Comparison of ACFs of ψ_0 between D1M, 2M, S1M-J, and S1M-$R[l]$ for different lags l for grid point: (a) $[200, 2390]$ km, (b) $[440, 3190]$ km.

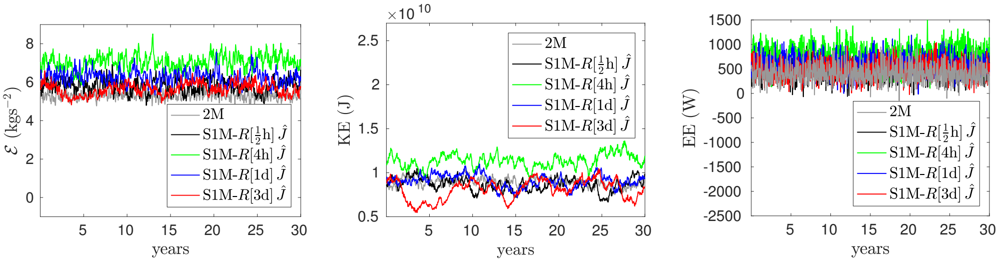


(a) (b)

Figure 3.12: Comparison of ACFs of ψ_0 between D1M, 2M, and S1M-$R[l_\theta]$ for different lag patterns l_θ for grid point: (a) $[200, 2390]$ km, (b) $[440, 3190]$ km.

The motivation for the sampling method ($R[l]\,\widehat{J}$) is to combine the benefits of both conditioning variables, i.e. the spatial structure of the Jacobian, and temporal correlations from lagged R-values, respectively.

Similar to previous simulations, all the tested simulations reproduce the mean barotropic stream function excellently (not shown), but the S1M-R[4h] $\widehat{J}$ and S1M-R[3d] $\widehat{J}$ simulations overestimate the standard deviation (not shown). While an immediate improvement over the S1M-$R[l]$ simulations in Section 4.1 can be seen, the results with the S1M-$R[l]J$ model are quite sensitive to the choice of l. This is illustrated in Figure 3.13, where physical results from a S1M-R[4h] $\widehat{J}$ simulation are shown in addition to the same lag-time choices discussed in Section 4.1. In Figure 3.13 the enstrophy and energetics are plotted for the various spatially constant lag times. On the one hand, the simulations S1M-$R[\frac{1}{2}\mathrm{h}]$ $\widehat{J}$ and S1M-R[1d] $\widehat{J}$ result in major improvements over the physical diagnostics resulting from S1M-$R[l]$ (Figure 3.7) and S1M-J (Figure 3.9). Specifically, the mean of the kinetic energy only deviates 2.9% and 1.7% from 2M's reference for S1M-$R[\frac{1}{2}\mathrm{h}]$ $\widehat{J}$ and S1M-R[1d] $\widehat{J}$, respectively. On the other hand, the S1M-R[4h] $\widehat{J}$ model performs overall worse than S1M-$R[l]$ (Figure 3.8).



(a) (b) (c)

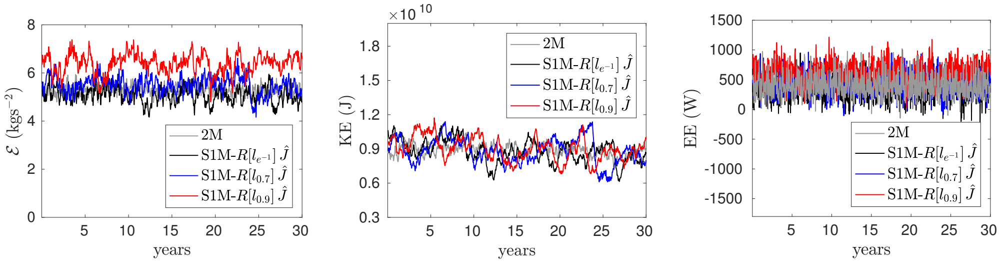
Figure 3.13: Comparison of scalar physical properties between 2M, and S1M-$R[l]\,\widehat{J}$ for different lags l: (a) enstrophy, (b) kinetic energy (with wider range on y-axis), (c) energy exchange term.

The sensitivity discussed above stems from the choice for a constant lag l because, as discussed in Section 3.2, decorrelation times of R vary widely between grid points. Instead of using the spatially constant lag times, we use spatially variable lag times based on the decorrelation time scales of the eddy forcing $R(i,j)$. As in Section 4.1, this spatially variable $l_\theta(i,j)$ is chosen equal to the time lag at which the ACF for $R(i,j)$ first crosses some threshold θ:

$$\widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid (\widetilde{R}^{n-l_\theta(i,j)}(i,j), \widehat{J}(\widetilde{\psi}_0^n(i,j),\widetilde{q}_0^n(i,j))), \quad \forall i,j \in \mathrm{WB}. \tag{$R[l_\theta]\,\widehat{J}$}$$

The results for S1M-$R[l_\theta]\,\widehat{J}$ are shown in Figure 3.14, for several values of θ. With $\theta = 0.9$, enstrophy and energy exchange are too high. We hypothesize that with $\theta = 0.9$, the lagged R-values are still very strongly correlated, so that they add little information and the conditioning is dominated by the Jacobian. As a result, S1M-$R[l_{0.9}]\,\widehat{J}$ suffers from similar errors as S1M-J (see Section 4.2).

The results with $\theta = 0.7$ and $\theta = e^{-1}$ are overall very good, with diagnostics in Figure 3.14 close to those of the reference model 2M. We focus here on S1M-$R[l_{0.7}]\,\widehat{J}$, however results for S1M-$R[l_{e^{-1}}]\,\widehat{J}$ are highly comparable. For $\theta = 0.7$ the mean of the enstrophy, kinetic energy, and energy exchange terms are all reproduced excellently, with an error of 3.5, 0.9%, and 9.2%, respectively. Additionally, the standard deviation of the energy exchange term is also within 4.7% of the reference, proving another significant improvement over the previously tested approaches.



(a) (b) (c)

Figure 3.14: Comparison of scalar physical properties between 2M, and S1M-$R[l_\theta]\,\widehat{J}$ for different spatial lag patterns l_θ: (a) enstrophy, (b) kinetic energy, (c) energy exchange term.

We note that the standard deviation of the kinetic energy is too high for all S1M-$R[l_\theta]\,\widehat{J}$. This is caused by the limited spatial dependency in our sampling method ($R[l_\theta]\,\widehat{J}$), which can lead to forcing fields that are spatially less smooth than in the 2M reference model. This increased spatial roughness affects local gradients and thereby the kinetic energy (see (3.15)). Despite this shortcoming, S1M-$R[l_{0.7}]\,\widehat{J}$ performs well by all other criteria.

By preserving the temporal information provided by the lagged R-values, the ACFs for 2M in $[200, 2390]$ km and $[440, 3190]$ km are reproduced almost exactly, as shown in Figure 3.15. This drastically improves on the autocorrelations reproduced by S1M-J and S1M-$R[l]$ (Figure 3.11), and equals the best results obtained with pure temporal stochastic parameterizations S1M-$R[l_\theta]$ (Figure 3.12).

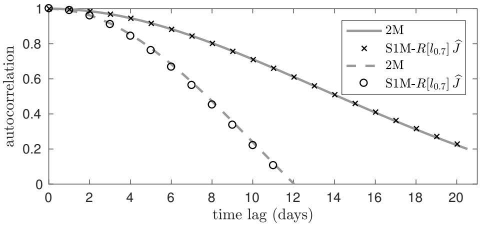
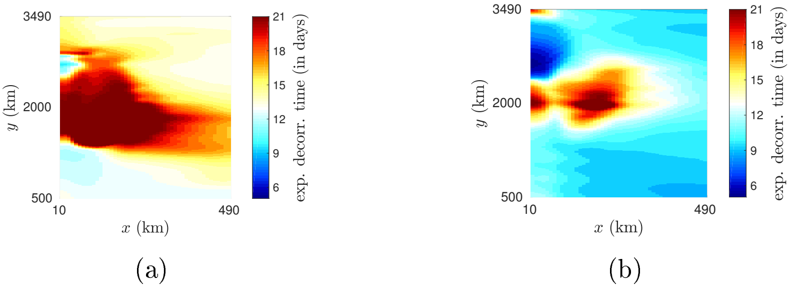


Figure 3.15: Comparison of the ACFs of ψ_0 between the two grid points $[200, 2390]$ km (line and crosses) and $[440, 3190]$ km (dashed and circles) between 2M and S1M-$R[l_{0.7}]\,\widehat{J}$

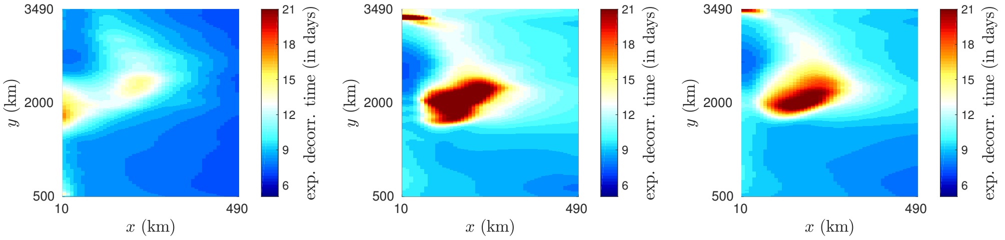
Let us next consider ACFs more comprehensively by focusing on the entire WB region. Consider the exponential decorrelation time scales, i.e. the time lag at which the ACF for the barotropic stream function first dips below e^{-1}, pointwise over the whole western boundary region (WB). Figure 3.16 shows the drastic differences of exponential decorrelation time scales between D1M and 2M; the significantly weaker vortices near the western boundary present in D1M cause much longer decorrelation time scales across the WB region.



(a) (b)

Figure 3.16: Pointwise exponential decorrelation time (in days) of ψ_0 in each of the following models: (a) D1M, (b) 2M.

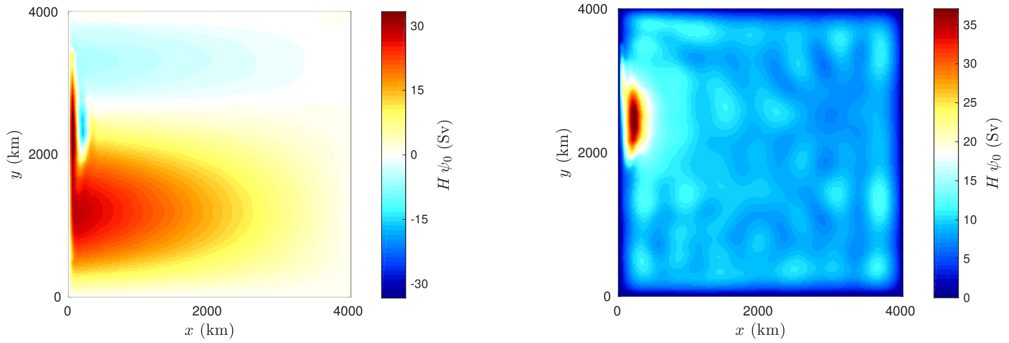
Consistent with the ACFs in Figures 3.11a and 3.11b, S1M-J preserves little of the temporal history (Figure 3.17a), as its ACFs decorrelate much faster over the whole WB region. However, this result already presents a strong improvement over decorrelation time scales exhibited in D1M (Figure 3.16a). Significant improvements come once again when we observe the results for simulations that condition on lagged R-values: qualitatively similar patterns for both S1M-$R[\frac{1}{2}\text{h}]$ (Figure 3.17b) and S1M-$R[l_{0.7}]\,\widehat{J}$ (Figure 3.17c). These patterns approximate the reference 2M (Figure 3.16b) very well. None of the tested simulations are able to adequately reproduce the high decorrelation time scales in the $[10, 40] \times [1800, 2300]$ km region. This may be caused by boundary effects, or by the dynamic complexities of the gyre's detachment point at this location. We do, however, emphasize that the main outcome from these results is that our goal of improving the stochastic model for $\widetilde{R}$ by combining spatial and temporal information (through conditioning on J and $R[l_\theta]$, respectively) is accomplished with the spatially dependent sampling method ($R[l_\theta]\,\widehat{J}$).



(a) (b) (c)

Figure 3.17: Pointwise exponential decorrelation time (in days) of $\widetilde{\psi}_0$ in each of the following models: (a) S1M-J, (b) S1M-$R[\frac{1}{2}\text{h}]$, (c) S1M-$R[l_{0.7}]\,\widehat{J}$.

Table 3.2 shows the expectation and maximum of the absolute errors ϵ_i over the grid for the first four statistical moments, e.g. $\epsilon_1 = \mu(H\psi_0) - \mu(H\widetilde{\psi}_0)$. Multiple errors for S1M-$R[\frac{1}{2}\text{h}]\,\widehat{J}$ are higher than their corresponding values for either S1M-$R[\frac{1}{2}\text{h}]$ or S1M-J, illustrating the previously discussed difficulties with spatially constant lag times. However, the S1M-$R[l_{0.7}]\,\widehat{J}$ simulation does not solve this completely, as the error ϵ_1 is shown to be worse than for S1M-$R[l_{0.7}]$. One sees that adding conditioning variables for our discrete sampling method does not guarantee a universal improvement to the statistical quantities, because the added conditioning variable J does correlate strongly with the eddy forcing in only part of the WB region. Aside from the first statistical moment, the S1M-$R[l_{0.7}]\,\widehat{J}$ simulation gives the best overall results for the statistical moments, consistent with our motivations and results, described earlier in this section. The most drastic improvements with respect to the D1M are found in the errors of the standard deviation, reducing the error by an order of magnitude. Crucially, these results are further emboldened by observing the spatial fields of the statistical moments of $\widetilde{\psi}_0$. Figures 3.18a and 3.18b show that the reference mean and standard deviation of ψ_0 (Figures 3.1b and 3.1c) are indeed extremely well reproduced by S1M-$R[l_{0.7}]\,\widehat{J}$.



(a) (b)

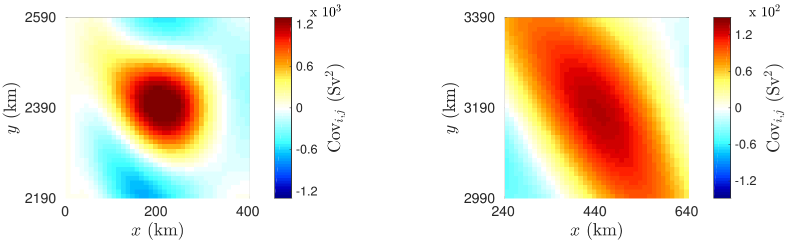
Figure 3.18: Pointwise temporal statistical moments of $\widetilde{\psi}_0$ in S1M-$R[l_{0.7}]\,\widehat{J}$: (a) mean, (b) standard deviation.

Table 3.2: Mean and maximum absolute errors of the first four statistical moments ϵ_s of the barotropic stream functions of stochastic simulations with different sets of conditioning variables.

	E	max	**E**	max	**E**	max	**E**	max
conditioning	ϵ_1	ϵ_1	ϵ_2	ϵ_2	ϵ_3	ϵ_3	ϵ_4	ϵ_4
magnitude	10^{-1}	10^{1}	10^{0}	10^{1}	10^{-2}	10^{0}	10^{-1}	10^{1}
unit	Sv	Sv	Sv	Sv	–	–	–	–
D1M	3.33	2.82	8.19	3.33	7.16	3.31	3.81	3.00
S1M-$R[\frac{1}{2}\mathrm{h}]$	1.95	1.18	2.80	1.20	5.56	3.15	2.12	3.04
S1M-$R[l_{0.7}]$	1.42	0.71	1.00	0.97	5.24	3.22	2.73	3.02
S1M-J	2.05	1.03	5.50	1.81	6.13	4.38	2.40	3.01
S1M-$R[\frac{1}{2}\mathrm{h}]\,\widehat{J}$	1.75	0.95	0.77	1.39	6.02	2.83	2.02	3.01
S1M-$R[l_{0.7}]\,\widehat{J}$	1.64	0.83	0.72	0.77	5.45	2.94	1.92	2.97

Figure 3.19 shows that, besides improving the energetics and statistical moments of the system, S1M-$R[l_{0.7}]\,\widehat{J}$ is also able to reproduce spatial covariances present in 2M, as can be seen by comparing Figure 3.19 with Figures 3.10g and

3.10h. This is a significant improvement over both S1M-$R[\frac{1}{2}\mathrm{h}]$ (Figures 3.10c and 3.10d) and S1M-J (Figures 3.10e and 3.10f). Given that both S1M-$R[\frac{1}{2}\mathrm{h}]$ and S1M-J reproduced the spatial patterns of the covariances qualitatively well, this quantitative improvement is most likely due to the more accurately reproduced standard deviation of the barotropic stream field (see Table 3.2).



(a) (b)

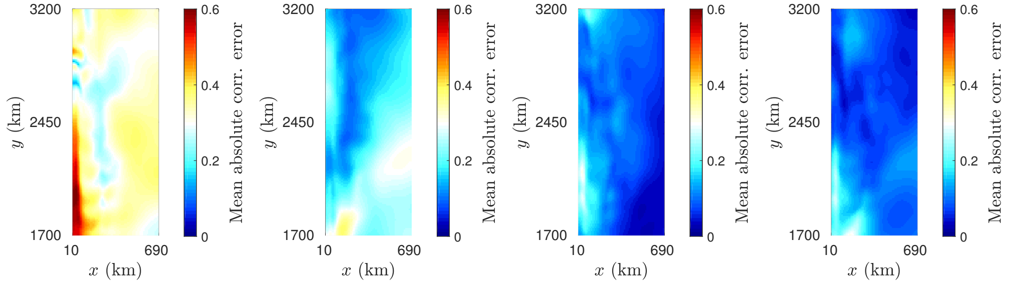
Figure 3.19: Covariance plots for ψ_0 for S1M-$R[l_{0.7}]\widehat{J}$ between grid point $[200, 2390]$ km/$[440, 3190]$ km) in (a)/(b) and neighbouring grid points.

To support these claims, let us investigate spatial correlations over the region $[10, 690] \times [1700, 3200]$ km, i.e. the region of the domain where the standard deviation of the barotropic stream is significant (see Figure 3.1c). Similar to the spatial covariances, for each grid point in the region we compute the correlations between ψ_0 in this central grid point and its surrounding grid points, see (3.18). Contrasted with the reference correlations in 2M, the mean absolute correlation errors in grid point (i, j) for a stochastic model are given by:

$$g(i,j) = P(i,j)^{-1} \cdot \sum_{\substack{i',j'>0 \\ i-20\leq i'\leq i+20 \\ j-20\leq j'\leq j+20}} |\,\mathrm{Corr}(\psi_0(i,j), \psi_0(i',j')) - \mathrm{Corr}(\widetilde{\psi}_0(i,j), \widetilde{\psi}_0(i',j'))\,|, \tag{3.4}$$

where $P(i, j)$ denotes the number of grid points (i', j') over which the summation in (3.4) runs (the summation cannot run over grid points that exceed the boundaries of the full domain), and ψ_0 and $\widetilde{\psi}_0$ are the barotropic stream functions of 2M and of the stochastic model, respectively.

These mean absolute correlation errors are shown in Figure 3.20 for each of the highlighted test models. The figure shows that the two grid points chosen to illustrate the covariances in Figure 3.10 are representative for the globally reproduced correlations. Importantly, Figure 3.20 shows that all stochastic models improve significantly on the spatial correlations as reproduced by the barotropic reference D1M. Additionally, one sees that, besides overestimating the magnitude of the covariances (Figures 3.10e and 3.10f), S1M-J (Figure 3.20b) reproduces the spatial patterns of the correlations with less accuracy than both S1M-$R[\frac{1}{2}\mathrm{h}]$ (Figure 3.20c) and S1M-$R[l_{0.7}]\,\widehat{J}$ (Figure 3.20d). These latter two reproduce the spatial correlations of the reference 2M with a similarly high accuracy. This indeed indicates that the improvements to the spatial covariances by S1M-$R[l_{0.7}]\,\widehat{J}$ (Figures 3.19a and 3.19b) are due to the standard deviation of the flow being better resolved, and is less likely attributed to the spatial correlations.



(a) (b) (c) (d)

Figure 3.20: Mean absolute correlation error (3.4) of $\widetilde{\psi}_0$ in each of the following models: (a) D1M, (b) S1M-J, (c) S1M-$R[\frac{1}{2}\mathrm{h}]$, (d) S1M-$R[l_{0.7}]\,\widehat{J}$.

Concluding, we have successfully introduced flow-dependency into the stochastic parameterization. In many respects the flow-independent parameterization S1M-$R[l_{0.7}]$, discussed in Section 4.1, already shows promising results. By introducing the Jacobian into the conditioning S1M-$R[l_{0.7}]\,\widehat{J}$ we have further improved almost all of the considered physical and statistical criteria posed in Section 1.2.

5 Summary and discussion

In this study we investigated a covariate-based stochastic approach to parameterize unresolved processes within a standard model of the idealised, wind-driven ocean circulation. We considered the reduction of the reference 2-mode baroclinic model (2M) to a 1-mode barotropic model (D1M). The reduced D1M lacks the baroclinic feedback given by the baroclinic eddy forcing R. We developed a stochastic model for R, and coupled it to the 1-mode model to obtain a stochastic 1-mode barotropic model (S1M). With a suitable stochastic model for R, S1M is able to mimic the behavior of the barotropic mode in the reference 2M closely.

We focused on vertical instead of horizontal coarse-graining, such that all considered models (2M, D1M, S1M) are discretized on the same high-resolution horizontal grid. Hereby, we avoid the subtle difficulties of horizontal coarse-graining (e.g. choices of filter and grid transformation), and can fully focus on the stochastic model formulation. The corresponding eddy forcing R is uniquely defined and has a clear physical interpretation solely related to the baroclinic nature of the flow.

The stochastic parameterization of the eddy forcing R is based on a covariate-approach recently developed in Verheul and Crommelin [143] within a scalar set-up. Here we construct a pointwise spatial extension of the covariate-approach such that it can be applied to a spatially extended ocean model. More precisely, in S1M the eddy forcing R is modeled as a spatially extended stochastic process $\widetilde{R}$. Sample data from a 2M reference simulation for both the eddy forcing R and the resolved model variables is assumed to be given for our approach. The stochastic term $\widetilde{R}$ is sampled uniformly from conditional probability distribution functions (CPDFs) approximated over the available sample data, i.e. sampled from the so-called conditional empirical distributions. The CPDFs are approximated with a simple binning procedure. $\widetilde{R}$ is then conditioned on both the most suitable flow-dependent covariate, which turned out to be the resolved nonlinear advection field, and past states of $\widetilde{R}$ itself, inducing adequate temporal correlations. For our specific ocean model, the volume of employed data could be significantly reduced by limiting the stochastic forcing to the western boundary sub-region of the grid, which in diagnostic tests proved to reproduce the flow dynamics of 2M.

In order to evaluate the performance of the stochastic parameterization, we compared the reference and reduced stochastic models (i.e. 2M and S1M) with respect to a range of physical and statistical criteria. These criteria are the first four statistical moments, the autocorrelation function, spatial covariances and correlations of the stream function, kinetic energy, energy conversion, and the enstrophy.

The results show that the flow-dependent covariate and the past states of $\widetilde{R}$ each contribute in their own way to the stochastic model. Conditioning on the past states of $\widetilde{R}$ introduces temporal consistency (Section 4.1). The flow-dependent covariate introduced an energetic driving force and improved the spatial cohesion of the system, but by itself is an insufficient predictor for the eddy forcing (Section 4.2).

Conditioning both on the flow-dependent covariate and on past $\widetilde{R}$-values resulted in a good model for R, so that S1M reproduced many characteristics of the 2M reference model very well (Section 4.3). In order to achieve this it turned out to be crucial to account for the spatial dependence of the decorrelation time scale of R.

Overall our final results with S1M show a large improvement over D1M, as well as very close similarity to the reference 2M. Additionally, the results provide further proof of concept of our methodology, extending the approach from Verheul and Crommelin [143] to a spatially extended setting. Because the empirical distribution is discrete and needs only one sample point per bin to sample from, it is a method robust to small sample sets. However, practical limitations can pose challenges, because of large amount of data involved for spatially extended systems. This led us to develop straightforward methods to use sample data more effectively, as well as to rethink ways to handle sparsely populated bins (Section 3.4). With a large amount of data, efficient storage and access of the data becomes more important for our methodology. In this study, optimization of memory usage and data storage was not our priority, however we point out that several straightforward optimizations are possible for our approach, e.g. reduced sampling interval, fast database-based lookup tables, and interpolated sample data. We leave these for future study. Furthermore, we emphasize the computational efficiency of the methodology developed here, as our methodology requires only straightforward calculations to determine what piece of memory to read out, i.e. what bin to sample from.

In future work we intend to explore methods to remedy these limitations. The approach proposed here can be further extended to improve the representation of spatial structures and correlations in the parameterized eddy forcing fields by, e.g, exploring additional types of flow-dependent covariates (e.g. based on energetics). A more thorough sensitivity analysis of the conditioning time lags of $\widetilde{R}$ would also be helpful and insightful for the covariate selection. Furthermore, we are interested in exploring continuous approximation methods for the CPDFs, as well as other continuous stochastic parameterizations.

Appendix

A Linear fit details

The conditioning procedure as defined in Section 3.1 is defined pointwise over the grid, therefore the number of sample points for each approximated CPDF in (3.2) is limited by the length of the time series ($T_c/\delta t$, see Table 3.1). This proved somewhat problematic in our experiments, because a straightforward equidistant binning (see Appendix B) did not always result in sufficiently filled bins in the case where (3.2) is conditioned on both the covariate $J(\widetilde{\psi}_0^n, \widetilde{q}_0^n)$ and the time lagged $\widetilde{R}^{n-l_\theta(i,j)}$. Therefore, in the case where we condition on both, we project the binning on the linear trend between $J(\widetilde{\psi}_0^n, \widetilde{q}_0^n)$ and $\widetilde{R}^{n-l_\theta(i,j)}$. This linear fit allows us to make more efficient use of the number of bins, i.e. to more evenly partition the sample data over the number of bins.

Let us define:

$$\overline{J}(\psi_0(i,j), q_0(i,j)) = k(i,j) + m(i,j)R(i,j), \tag{3.5}$$

to be the linear trend between $J(\psi_0(i,j), q_0(i,j))$ and $R(i,j)$ (see (3.1)), where $[k(i,j), m(i,j)] = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$, and $\boldsymbol{X}(i,j) = [1, (\boldsymbol{R})^T]$ a $(T/\delta t \times 2)$-tensor, and $\boldsymbol{y}(i,j) = (\boldsymbol{J})^T$ a $(T/\delta t \times 1)$-columnvector. Then, by subtracting this linear trend from the Jacobian values we get:

$$\widehat{J}(\psi_0(i,j), q_0(i,j)) = J(\psi_0(i,j), q_0(i,j)) - \overline{J}(\psi_0(i,j), q_0(i,j)),$$

and using the conditioning set $\{\widetilde{R}^{n-l_\theta(i,j)}(i,j), \widehat{J}(\widetilde{\psi}_0^n(i,j), \widetilde{q}_0^n(i,j))\}$ results in the linearly fitted sampling procedure:

$$\begin{aligned} \widetilde{R}^{n+1}(i,j) \sim R^{n+1}(i,j) \mid \widetilde{R}^{n-l_\theta(i,j)}(i,j), (\widehat{J}(\widetilde{\psi}_0^n(i,j), \widetilde{q}_0^n(i,j)) \\ \text{for } i \in \{1,\dots,N_x\},\, j \in \{1,\dots,N_y\} \end{aligned} \tag{3.6}$$

B Empirical distribution details

Here we elaborate on the general description of the empirical distribution that is found in Section 3.4. The equidistant binning method independently partitions the range between the minimum and maximum of each of the covariates into N_B intervals α_b. Each of the intervals α_b describes a set ρ_b of R^{n+1}-values. Let $\mathcal{U}(A)$ denote a uniform distribution over the elements of the set A, i.e. if $a' \sim \mathcal{U}(A)$ then $\forall a \in A:\ P(a' = a) = |A|^{-1}$. The conditional sampling method (3.2) is then approximated by the empirical approach:

$$\begin{aligned} &\widetilde{R}^{n+1}(i,j) \sim \mathcal{U}(\rho_b(i,j)), \\ &\qquad \text{where } b:\ (\widetilde{R}^n(i,j), \widehat{J}(\widetilde{\psi}_0^n(i,j), \widetilde{q}_0^n(i,j)) \in \alpha_b(i,j). \end{aligned} \tag{3.7}$$

This empirical approach is a prototypical example of a discrete sampling method. An obvious limitation of such methods is that they can exclusively sample from observed sample values. Therefore, the stochastic model has no predetermined way of handling situations in which the values of the conditioning variables are outside of the ranges exhibited in the sample data. With respect to the equidistant binning procedure, see Section 3.4, this situation manifests as empty bins, i.e. a bin $\alpha_b(i,j)$ for which $\rho_b(i,j)$ is empty. The likelihood of the conditioning variable's values to arrive at an empty bin is predominantly determined by three factors: the number of conditioning variables D, the number of bins per conditioning variable N_B, and the limitations on the amount of sample data. With each of the aforementioned factors, this is a non-trivial issue. We solve this problem by linking each of the empty bins to a nonempty bin in the training phase of the simulation. Because the covariate $\widehat{J}(\psi_0, q_0)$ is least likely to hold predictive qualities for the sampling procedure (as explained in Section 3.4), the predictive quality of this covariate is considered negligible in empty bins. And thus, the conditioning in empty bins is considered 1-dimensional, i.e. in empty bins we condition only on $R^{n-l(i,j)}$, which is, by construction, guaranteed to be in range of sample data.

C Spatial smoothing

With the pointwise conditioning procedure (see Section 3.1) the spatial correlations of R are only modeled implicitly by conditioning on the Jacobian term $J(\psi_0, q_0)$. As discussed in Section 1, finding more rigorous ways to explicitly reproduce spatial correlations will stay a topic for an immediate future project. We use a more heuristic method to introduce spatial smoothness and remedy the spatially uncorrelated fields here. Namely, we use a truncated Gaussian smoothing filter $G(x, y)$:

$$G'(x,y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2+(y-\mu)^2}{2\sigma^2}}, \qquad G(x,y) = \frac{G'(x,y)}{\sum_{x,y} G'(x,y)}, \tag{3.8}$$

where we choose a basic 3×3 filter $G(x, y)$ with $\mu = 0$ and $\sigma = 0.5$, where $x, y \in \{-1, 0, 1\}$. Ilustrating snapshots are shown in Figure 3.21, where one can see the spatially smoothed snapshot more closely resembling the resolved R, both in structure and spatial smoothness.

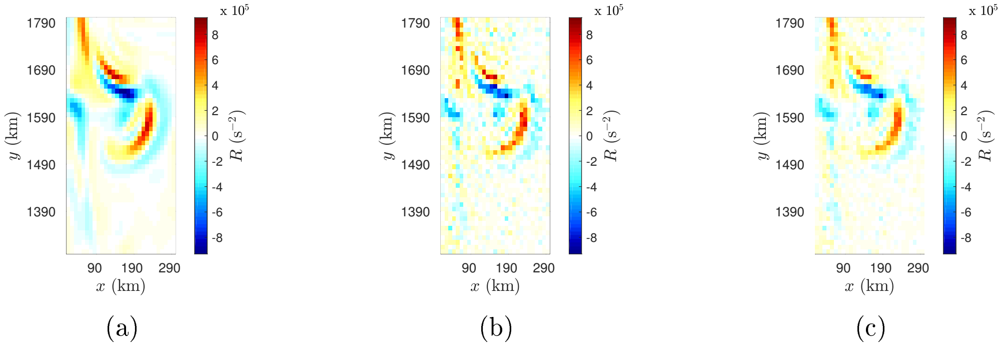


Figure 3.21: A comparison between snapshots of the resolved R (left), the unfiltered emulated stochastic forcing (middle), and the filtered emulated stochastic forcing (right).

As previously mentioned, this smoothing is an added heuristic method to strengthen the spatial correlations between neighboring grid points. But more importantly, because the stochastic procedure, as defined in (3.2), works completely pointwise over the grid, this Gaussian filter will smooth out unwanted spatial roughness. Otherwise, the spatial roughness would artificially 'add energy' into the system, spuriously exciting the system beyond realistic goals.

D Definitions of diagnostic criteria

- The first four statistical sample moments: mean, standard deviation, skewness, and kurtosis over the model variables ψ_0 (pointwise in physical space):

$$\mu(\psi_0) = \mathbf{E}(\psi_0^i) \tag{3.9}$$

$$\mathrm{std}(\psi_0) = \left(\mathbf{E}((\psi_0^i)^2) - (\mathbf{E}(\psi_0^i))^2\right)^{1/2} \tag{3.10}$$

$$\gamma(\psi_0) = \mathbf{E}\left[(\psi_0^i - \mathbf{E}(\psi_0^i))^3\right](\mathbb{V}\mathrm{ar}(\psi_0^i))^{-3/2} \tag{3.11}$$

$$\mathrm{Kurt}(\psi_0) = \mathbf{E}\left[(\psi_0^i - \mathbf{E}(\psi_0^i))^4\right](\mathbb{V}\mathrm{ar}(\psi_0^i))^{-2} \tag{3.12}$$

- The autocorrelation function (ACF) with time lag l over the model variable ψ_0:

$$\mathrm{ACF}_l(\psi_0) = \mathbf{E}\Big[(\psi_0^i - \mathbf{E}(\psi_0^i))(\psi_0^{i+l} - \mathbf{E}(\psi_0^i))\Big](\mathbb{V}\mathrm{ar}(\psi_0^i))^{-1} \tag{3.13}$$

- The mean, standard deviation, and the variability (both short-term and long-term variability) of the time-dependent scalar quantities enstrophy $\mathcal{E}$, kinetic energy KE, and energy exchange/transfer EE as horizontally integrated over grid A:

$$\mathcal{E} = \frac{\overline{\rho}H}{2}\int_A q_0^2, \tag{3.14}$$

$$\mathrm{KE} = -\frac{\overline{\rho}H}{2}\int_A (\nabla\psi_0)^2, \tag{3.15}$$

$$\mathrm{EE} = \overline{\rho}H\int_A \psi_0\, J(\psi_1, q_1). \tag{3.16}$$

- The covariance $\mathbb{C}ov(\psi_0)$ between grid points (i,j) and (i',j') for model variable ψ_0:

$$\begin{aligned}&\mathbb{C}ov(\psi_0(i,j),\psi_0(i',j')) =\\ &\qquad \mathbf{E}([\psi_0(i,j)-\mathbf{E}(\psi_0(i,j))][\psi_0(i',j')-\mathbf{E}(\psi_0(i',j'))])\end{aligned} \tag{3.17}$$

- Similar to the covariance above, the correlation $\mathrm{Corr}(\psi_0)$ between grid points (i,j) and (i',j') for model variable ψ_0:

$$\mathrm{Corr}(\psi_0(i,j),\psi_0(i',j')) = \frac{\mathbb{C}ov(\psi_0(i,j),\psi_0(i',j'))}{\mathrm{std}(\psi_0(i,j))\mathrm{std}(\psi_0(i',j'))} \tag{3.18}$$

E Numerical integration details

The numerical implementation uses the standard centered difference scheme $(\Delta x)^{-n}\,\delta_x^n f(x,y,t)$ of order $O((\Delta x)^2)$ to approximate the n-th order spatial derivatives [140] in 2M, where:

$$\delta_x^n f(x,y,t) = \begin{cases} \displaystyle\sum_{i=0}^{n}(-1)^i\binom{n}{i}f(x+\left(\frac{n}{2}-i\right)\Delta x,y,t) & \text{if } n \text{ even,}\\ \displaystyle\sum_{i=0}^{n}(-1)^i\binom{n}{i}\frac{1}{2}\left[f(x+\left(\frac{n+1}{2}-i\right)\Delta x,y,t)+\right. & \\ \quad \left. f(x+\left(\frac{n-1}{2}-i\right)\Delta x,y,t)\right] & \text{if } n \text{ odd,}\end{cases} \tag{3.19}$$

and analogously for partial derivatives to y.

We choose the Arakawa stencil $J_1 = \frac{1}{3}(J^{++}+J^{+\times}+J^{\times+})$ to discretize the Jacobian terms for its conservation properties, this Arakawa scheme satisfies finite difference analogs of energy and mean squared vorticity conservation laws [3].

To integrate D1M, 2M, and S1M over time, let us consider the second-order centered-difference leapfrog scheme [42, 72, 85], which is used widely in weather and climate models. The leapfrog time stepping method admits a well-documented spurious computational mode that manifests as spurious oscillations between even and odd time steps that amplify during nonlinear simulations [42, 95, 157]. This phenomenon is referred to as time-splitting. The

Robert–Asselin filter [5, 119] is an oft-used time-filtering solution (for a list of examples in climate models see, e.g, Williams [155]) that dampens this computational mode and thus controls the time-splitting instability. The combination of the leapfrog scheme with the Robert–Asselin filter results in the Asselin-leapfrog scheme:

$$\psi^{n+1} = \overline{\psi}^{n-1} + 2\Delta t\, F(\psi^n), \tag{3.20}$$

where $F(\psi)$ denotes the right-hand side terms in the PDE governing the temporal evolution of ψ, and the bar notation in (3.20) denotes the Robert–Asselin time-filter:

$$\overline{\psi}^{n} = \psi^n + \alpha(\overline{\psi}^{n-1} - 2\psi^n + \psi^{n+1}), \tag{3.21}$$

where the parameter α denotes the Robert–Asselin filter strength.

We further impose the free-slip condition $n_\perp \cdot (\nabla \circ \nabla)\psi_m = 0$ along the lateral boundaries, where $n_\perp$ denotes the horizontal unit vector normal to the boundary, and $\circ$ is the Hadamard product, defined for matrices A and B with same dimensions by $(A \circ B)_{ij} = (A)_{ij}(B)_{ij}$.

All our choices for model parameters are shown in Table 3.1. The discussed discretization methods result in the following discretized integration scheme for the 2-mode referemce model 2M:

$$\begin{aligned}
&\psi_0^{n+1} = \overline{\psi}_0^{n-1} + 2\Delta t\, S_0(\psi_0^n, \psi_1^n), \qquad \psi_1^{n+1} = \overline{\psi}_1^{n-1} + 2\Delta t\, S_1(\psi_0^n, \psi_1^n),\\
&\quad \text{where } \nabla^2 S_0(\psi_0^n, \psi_1^n) = -\, J(\psi_0^n, q_0^n) - R^n - \beta\partial_x\psi_0^n + A_H\nabla^4\psi_0^{n-1} +\\
&\qquad\qquad \frac{\partial_x\tau^y - \partial_y\tau^x}{\overline{\rho}H}\\
&\quad \text{where } \nabla^2 S_1(\psi_0^n, \psi_1^n) = -\, J(\psi_1^n, q_0^n) - J(\psi_0^n, q_1^n) - \epsilon_{111} J(\psi_1^n, q_1^n) -\\
&\qquad\qquad \beta\partial_x\psi_1^n + A_H\nabla^4\psi_1^{n-1} +\\
&\qquad\qquad \frac{\varphi_1^n(z=0)(\partial_x\tau^y - \partial_y\tau^x)}{\overline{\rho}H},
\end{aligned} \tag{3.22}$$

where numerical solutions to the Poisson's equations for $\nabla^2 S_m$ are found with the Intel MKL Poisson solver. The viscosity terms above are taken from the previous time-step for numerical stability purposes.

The discretized 2M (3.22) is initialized with the conditions $\overline{\psi}_0^{n=-1} = \overline{\psi}_1^{n=-1} = \psi_0^{n=0} = \psi_1^{n=0} = 0$ and $q_0^0 = q_1^0 = 0$. To equilibrate any baroclinic simulation starting from rest, a sufficient spin-up period needs to be performed. For all our experiments, we choose a 30 year spin-up time. The integration time-step Δt is chosen to be half an hour for both the barotropic and baroclinic models, these choices are made to find comparison with similar set ups (e.g. [10, 13, 117]).

Applying the pointwise sampling method (3.2) to evolve $\widetilde{R}$ over time, and numerically integrating S1M results in the following numerical integration scheme for the discretized stochastic 1-mode model:

$$
\begin{aligned}
\widetilde{\psi}_0^{n+1} &= \overline{\widetilde{\psi}}_0^{n-1} + 2\Delta t\, S(\widetilde{\psi}_0^n), \\
\widetilde{R}^{n+1} &\sim R^{n+1} \mid (R^{n-l_\theta}, C(\psi_0))) = (\widetilde{R}^{n-l_\theta}, C(\widetilde{\psi}_0)) \\
\text{with } \nabla^2 S(\widetilde{\psi}_0^n) &= - J(\widetilde{\psi}_0^n, \widetilde{q}_0^n) - \widetilde{R}^n - \beta \partial_x \widetilde{\psi}_0^n + A_H \nabla^4 \widetilde{\psi}_0^{n-1} + \\
&\quad \frac{\partial_x \tau^y - \partial_y \tau^x}{\overline{\rho} H},
\end{aligned}
\tag{3.23}
$$

where the initial conditions are chosen to start from an equilibrated baroclinic state before starting the 30 years spin-up for the stochastic simulation: $\overline{\widetilde{\psi}}_0^{-1} = \overline{\psi}_0^{M-1}$, $\overline{\widetilde{\psi}}_1^{-1} = \overline{\psi}_1^{M-1}$, $\widetilde{\psi}_0^0 = \psi_0^M$, $\widetilde{\psi}_1^0 = \psi_1^M$ and $\widetilde{q}_0^0 = q_0^M$, $\widetilde{q}_1^0 = q_1^M$.

Chapter 4

Stochastic parameterization with VARX processes

In this study we investigate a data-driven stochastic methodology to parameterize small-scale features in a prototype multiscale dynamical system, the Lorenz '96 (L96) model. We propose to model the small-scale features using a vector autoregressive process with exogenous variable (VARX), estimated from given sample data. To reduce the number of parameters of the VARX we impose a diagonal structure on its coefficient matrices. We apply the VARX to two different configurations of the 2-layer L96 model, one with common parameter choices giving unimodal invariant probability distributions for the L96 model variables, and one with non-standard parameters giving trimodal distributions. We show through various statistical criteria that the proposed VARX performs very well for the unimodal configuration, while keeping the number of parameters linear in the number of model variables. We also show that the parameterization performs accurately for the very challenging trimodal L96 configuration by allowing for a dense (non-diagonal) VARX covariance matrix.

1 Introduction

1.1 Background

For many spatially extended dynamical systems the equations of motion cannot be solved on sufficiently fine scales because of unfeasible computational costs. The typical approach for dealing with this problem is to formulate a reduced system that describes the variables of interest, usually the large-scale degrees of freedom. To compensate for the missing dynamical effects (feedback) that arise from the small scales, some dynamical term that represents or approximates these missing effects needs to enter the reduced system. Following common terminology in ocean-atmosphere science where this is an important problem, we call such terms parameterizations. In a previous study, we considered discrete resampling-based methods [143, 145]. These methods were successful in reproducing various important statistical and physical aspects in the reduced models. These promising results notwithstanding, their capability to model spatial correlations in the dynamical feedback from the small scales is limited. Here, instead, we investigate parameterizations that are better able to reproduce the spatio-temporal correlations explicitly, without significant computational cost.

Specifically, we propose to use a vector autoregressive process with exogenous parameters (VARX) for parameterization. We include endogenous and exogenous variables in the VARX process with coefficient matrices that have sparse structure, e.g. (tri)-diagonal. The main aim is that the reduced model with the parameterization accurately reproduces the statistical properties of the reference (fully resolving, non-reduced) model, including its spatial correlations. Moreover, our stochastic parameterization assumes no knowledge of the underlying physical structure of the system. We use available sample data from the fully resolving reference model to infer the VARX model, similar in spirit to the data-driven approaches in Crommelin and Vanden-Eijnden [30], Porta Mana and Zanna [117], Verheul and Crommelin [143], Verheul et al. [145]. In the context of ocean-atmosphere modeling, various other forms of stochastic parameterizations have been considered, e.g. stochastic cellular automata [9, 29, 133], and Markov chain approaches [30, 41, 89, 103], see also [16] for a recent overview.

We consider multiscale models wherein the state vector $\boldsymbol{z} := (\boldsymbol{x}, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_J)$ evolves over time according to a set of coupled ordinary differential equations

(ODEs) that include a constant forcing $\mathcal{F}$, a linear operator $\mathcal{L}\boldsymbol{z}$, and some nonlinear operator $\mathcal{B}(\boldsymbol{z})$. This set of ODEs can result from the spatial discretization of a partial differential equation; in this study we focus on the ODE formulation in which the elements of the state vector are associated with, e.g, values on a spatial grid. We consider nonlinear ODEs of the following form (occurring in e.g. ocean models [13, 81]):

$$\frac{d\boldsymbol{x}}{dt} = \mathcal{F} + \mathcal{L}_x\boldsymbol{x} + \mathcal{B}_{xx}(\boldsymbol{x}) + \mathcal{B}_{xy}(\boldsymbol{x}, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_J) \tag{4.1}$$

$$\frac{d\boldsymbol{y}_j}{dt} = \mathcal{L}'_x\boldsymbol{x} + \mathcal{L}_y\boldsymbol{y}_j + \mathcal{B}_{yy}(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_J) + \mathcal{B}_{yx}(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_J, \boldsymbol{x}), \tag{4.2}$$

where the vector $\boldsymbol{x} := (x_1, \ldots, x_K)$ represents the large-scale processes, the vectors $\boldsymbol{y}_j := (y_{j,1}, \ldots, y_{j,k})$ represent the small-scale processes, where $1 \leq j \leq J$, $1 \leq k \leq K$ are spatial grid indices, and each x_k is coupled to J small-scale $y_{j,k}$. Thus, K is the total number of gridpoints on which the large-scale processes are defined. This number can be very large for spatially extended systems (e.g. $K = 10^6$ for a system with 2 spatial dimensions, specified on a 1000×1000 grid). The $y_{j,k}$ can be thought of as being defined on a micro-grid (with J gridpoints) associated with each macro-gridpoint k.

The operator $\mathcal{B}_{yy}$ denotes the nonlinear self-interaction of the $\boldsymbol{y}_j$ variables, and $\mathcal{B}_{xy}$ denotes the nonlinear feedback of the $\boldsymbol{y}_j$ variables on the $\boldsymbol{x}$ variables. The operators $\mathcal{B}_{xx}$ and $\mathcal{B}_{yx}$ have analogous interpretations. We assume that an analytic solution to (4.1)-(4.2) is not available, so that we have to resort to numerical integration. The computational bottleneck for numerical integration of Equations (4.1)–(4.2) is evolving all $\boldsymbol{y}_j$ variables for each x_j. Therefore, we construct a reduced model involving only the variables of interest $\boldsymbol{x}$. This reduced model consists of (4.1) with $\mathcal{B}_{xy}$ replaced by a stochastic (VARX) parameterization $\widetilde{\boldsymbol{b}} := \widetilde{\boldsymbol{b}}(\widetilde{\boldsymbol{x}})$ that is meant to emulate $\boldsymbol{b} := \mathcal{B}_{xy}(\boldsymbol{x}, \boldsymbol{y}_1, \ldots, \boldsymbol{y}_J)$. To distinguish between variables in the original deterministic model (e.g. $\boldsymbol{x}$), and their analogues in the reduced stochastic model (e.g. $\widetilde{\boldsymbol{x}}$) we use the tilde-notation for all variables in the stochastic model. Thus, the reduced model is

$$\frac{d\widetilde{\boldsymbol{x}}}{dt} = \mathcal{F} + \mathcal{L}_x\widetilde{\boldsymbol{x}} + \mathcal{B}_{xx}(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{x}}) + \widetilde{\boldsymbol{b}}(\widetilde{\boldsymbol{x}})\,. \tag{4.3}$$

The state-dependence of $\widetilde{\boldsymbol{b}}(\widetilde{\boldsymbol{x}})$ allows the properties of the stochastic process to evolve together with the resolved variables ($\widetilde{\boldsymbol{x}}$). The parameters of the process $\widetilde{\boldsymbol{b}}(\widetilde{\boldsymbol{x}})$ are inferred from reference simulation data ($\boldsymbol{X}, \boldsymbol{B}$), obtained by numerical integration of Equations (4.1)–(4.2). Here $\boldsymbol{X} := (\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N)$, $\boldsymbol{B} := (\boldsymbol{b}^1, \ldots, \boldsymbol{b}^N)$ in which $\boldsymbol{x}^n := \boldsymbol{x}(t_n) := \boldsymbol{x}(n\Delta t)$ denotes the n-th time-instance of $\boldsymbol{x}$ and $\boldsymbol{b}^n := \mathcal{B}_{xy}(\boldsymbol{x}^n, \boldsymbol{y}_1^n, \ldots, \boldsymbol{y}_J^n)$ denotes the n-th time-instance of $\boldsymbol{b}$. Finally, N denotes the number of sample points (or time steps).

In section 2 we present a simple and straightforward VARX framework that uses sparse coefficient matrices. Then in Section 3 we apply our parameterization to the Lorenz '96 (L96) model [99], a frequently used test bed for developing parameterization methods [25, 30, 111, 153]. Next, we discuss technical details of our parameterization in Section 4 and present numerical results in Section 5.

2 VARX representation

We model the stochastic term $\widetilde{\boldsymbol{b}}$ in (4.3) as a VARX process (see, e.g, Lütkepohl [100]). Numerical implementation of such a process is straightforward (e.g. Pavliotis [114]). We make no assumptions about the underlying physics of $\boldsymbol{b}$, instead we infer the VARX process from the second-order statistics of $\boldsymbol{b}$ estimated from the available sample data ($\boldsymbol{X}, \boldsymbol{B}$).

2.1 Mean: linear combination of covariates

A complete characterization of a VARX(p), i.e. VARX of order p, is given by its drift matrices A_i, $i = 1, \ldots, p$ and D and covariance matrix $\Sigma\Sigma^T$:

$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_1 \widetilde{\boldsymbol{b}}^{n-1} + \cdots + A_p \widetilde{\boldsymbol{b}}^{n-p} + D\boldsymbol{x}^n + \Sigma \boldsymbol{\xi}^n \tag{4.4}$$

where $\boldsymbol{a}_0$ is the linear offset, $A_1, \ldots, A_p$ represent the *endogenous* drift matrices, D is the *exogenous* drift matrix, $\Sigma\Sigma^T$ is the covariance matrix, and $\boldsymbol{\xi}^n$ is a vector of independent normally distributed random variables, $\boldsymbol{\xi}^n \sim \mathcal{N}(0, I)$. The matrices A_i, D, and Σ all have size $K \times K$.

Borrowing some terminology from statistics, the variable $\widetilde{\boldsymbol{b}}^n$ is known as the *regressand* and the variables $\widetilde{\boldsymbol{b}}^{n-1}, \ldots, \widetilde{\boldsymbol{b}}^{n-p}, \boldsymbol{x}^n$ are known as the *regressors*. By choosing regression coefficient matrices A_i or D to be nonzero, the variable $\widetilde{\boldsymbol{b}}^n$ becomes dependent on those regressors. By imposing certain sparsity patterns on the drift matrices in (4.4) we can choose to make $\widetilde{\boldsymbol{b}}^n$ conditional on $\widetilde{\boldsymbol{b}}$ or $\boldsymbol{x}$ at specific space or (past) time points. For example, $\widetilde{b}_k^n$ can be made conditionally dependent on its previous state $(n-1)$ at neighboring gridpoints $(k \pm 1)$ by letting the matrix elements $(A_1)_{l,m}$ be nonzero if $(l,m) = (k \pm 1, k)$ or if $(l,m) = (k, k \pm 1)$. Similarly, if D is diagonal, $\widetilde{b}_k^n$ is conditionally dependent on x_k^n (i.e., at the same spatial grid point with index k) but not on $x_{k'}^n$ at grid points $k' \neq k$.

Typically, the matrices A_i and D in (4.4) are obtained through maximum likelihood estimation. We apply the weighted least squares procedure [137] to obtain accurate estimators. The training phase of our proposed algorithm consists primarily of calculating the regression coefficients (i.e., the elements of the matrices A_i, D and $\Sigma\Sigma^T$). Since the weighted least squares procedure is highly optimized, this training phase is very cost-efficient. Generalizations of this approach are possible by modeling $\widetilde{\boldsymbol{b}}$ as realizations of a Gaussian process where the drift components are represented as generalized linear models (GLIMs) (see e.g. McCullagh and Nelder [107] for a detailed description on GLIMs).

With data-driven approaches, the number of parameters that must be estimated and/or the amount of sample data needed, can grow exponentially in the number of conditioning variables, see, e.g, the binning approach in Verheul and Crommelin [143], Verheul et al. [145] or the conditional Markov chain setups in Crommelin and Vanden-Eijnden [30], Dorrestijn et al. [38], Gottwald et al. [65]. Here we mitigate this problem by imposing the structure of VARX, wherein, even if all matrices in (4.4) are fully dense (and thus $\widetilde{b}_k^n$ is conditional on the entire vector $\boldsymbol{x}^n$ as well as on all vectors $\widetilde{\boldsymbol{b}}^{n'}$ with $n - p \leq n' \leq n-1$), we still only have $K + K^2(p+2)$ parameters. By restricting the drift matrices in (4.4) to be sparse, the number of parameters reduces further. For example, if we choose all A_i, D and $\Sigma\Sigma^T$ to be diagonal matrices, then the number of parameters grows linearly in p and K.

With this approach, estimation and order selection are nontrivial issues. For example, order selection is difficult because $\boldsymbol{b}$ will be very strongly correlated with itself at short lag times and effectively decorrelated at long lag times. For a more detailed discussion of the order selection problem, see Section 3.2.1. Another difficulty for estimation is that the model needs to satisfy the stationarity constraints, otherwise the trajectory of the model variables can diverge to infinity. In order for the VAR(p) to be stationary, the matrix elements of the A_i

must satisfy the VAR(p) stability constraint [101]:

$$\forall \lambda : \quad \left| I^n \lambda^p - A_1 \lambda^{p-1} - A_2 \lambda^{p-2} \cdots - A_p \right| = 0 \quad \Rightarrow \quad |\lambda| < 1. \tag{4.5}$$

Equivalently, the VAR(p) is stationary if the eigenvalues of the *companion matrix* F have modulus less than one, where the companion matrix of (4.4) is defined as:

$$\begin{bmatrix} A_1 & A_2 & \dots & A_n \\ \mathbf{1}_n & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} & \vdots \\ \mathbf{0} & \dots & \mathbf{1}_n & \mathbf{0} \end{bmatrix}. \tag{4.6}$$

The development of regression methods that explicitly enforce this stability constraint is beyond the scope of this study. We only verify that our models satisfy the stability constraint *a posteriori*.

2.2 Covariance and resulting VARX

We consider two different forms of the covariance $\Sigma\Sigma^T$ of the VARX process (4.4). In one, the covariance matrix is a multiple of the identity matrix, i.e. all cross-covariances are ignored and auto-covariances do not depend on k. In the other, the covariance matrix is fully dense, allowing for nonzero cross-covariances and k-dependent auto-covariances.

Given the constant offset $\boldsymbol{a}_0$ and the matrices $\{A_i\}_{1\leq i\leq p}$ and D, we calculate the residuals $\boldsymbol{b}^n - \boldsymbol{a}_0 - A_1\boldsymbol{b}^{n-1} - \dots - A_p\boldsymbol{b}^{n-p} - D\boldsymbol{x}^n$ of the regression fit from the sample time series $(\boldsymbol{X}, \boldsymbol{B})$. For the first form of the covariance we set $\Sigma_D := \sigma I$, where σ is the averaged standard deviation over the residuals over all k. Although this form is extremely simple, it has only a single parameter (σ) so that it can easily be used even when K is very large. For a dense covariance matrix we compute all the pairwise sample covariances from residuals. Σ_L is then obtained from the Cholesky decomposition of the sample covariance matrix. This is straightforward and general but becomes unfeasible for large K (we recall that the covariance matrix and hence also Σ_L is of size $K \times K$). However, we include this covariance structure as an “optimal” reference for the current study.

Applying this VARX(p) model as forcing $\widetilde{\boldsymbol{b}}$ to the reduced model (4.3) results in the following stochastic model:

$$\begin{aligned} \widetilde{\boldsymbol{x}}^{n+1} &= T\left(\Delta t,\, \mathcal{F} + \mathcal{L}_x \widetilde{\boldsymbol{x}}^n + \mathcal{B}_{xx}(\widetilde{\boldsymbol{x}}^n, \widetilde{\boldsymbol{x}}^n) + \widetilde{\boldsymbol{b}}^n\right), \\ \widetilde{\boldsymbol{b}}^n &= \boldsymbol{a}_0 + A_1 \widetilde{\boldsymbol{b}}^{n-1} + \cdots + A_p \widetilde{\boldsymbol{b}}^{n-p} + D\widetilde{\boldsymbol{x}}^n + \Sigma \boldsymbol{\xi}^n, \end{aligned} \tag{4.7}$$

where T represents a numerical integration scheme of choice (see Section 4), and Σ can be either Σ_D or Σ_L. We emphasize the coupling between $\widetilde{\boldsymbol{x}}$ and $\widetilde{\boldsymbol{b}}$ goes in both directions: $\widetilde{\boldsymbol{b}}$ enters as a forcing term in the time integration of $\widetilde{\boldsymbol{x}}$, whereas the time evolution of $\widetilde{\boldsymbol{b}}$ depends on $\widetilde{\boldsymbol{x}}$ through the dependence of the VARX process on $\widetilde{\boldsymbol{x}}$. Such a state-dependence allows for the modeling of different dynamical regimes of $\widetilde{\boldsymbol{b}}$. If $\boldsymbol{x}$ and the chosen lagged $\boldsymbol{b}$ are adequate predictors, such regimes can occur in a similar fashion as in the sample data $(\boldsymbol{X}, \boldsymbol{B})$.

We note that while the VARX process allows for a spatially varying (i.e., k-dependent) mean and covariance, only the mean is able to vary temporally. Therefore, we expect our parameterization to be less suitable for cases where the small-scale processes have multiple variance regimes under the same large-scale state $\boldsymbol{x}$.

2.3 Computational complexity

The methodology we propose here requires very little computational cost in the training stage. First, the regression matrices A_i and D in (4.7) are calculated with a single least squares call. The least squares algorithm is very efficient with computational complexity $O(K^2N)$, and a well-optimized routine on many computational platforms. Second, the covariance $\Sigma\Sigma^T$ is estimated straightforwardly with the sample (co)variances calculated from the residuals, also with complexity $O(K^2N)$. In the case of VARX models with diagonal covariance, the matrix root Σ_D of $\Sigma_D\Sigma_D^T$ is computed directly with sample standard deviations. In the alternate case of fully dense covariance, the matrix root Σ_L is computed with a Cholesky decomposition. For $N > K$, the Cholesky decomposition is a less costly operation with $O(K^3)$ complexity that only needs to be calculated once in the initialization phase because our covariance is constant over time.

The motivation for restricting the regression matrices A_i, D and Σ (by imposing sparsity, e.g. a diagonal form) has two origins: first, the memory usage. Many ocean-atmosphere studies consider models with very large grids, e.g. $K = 512^2$ gridpoints. Full covariance matrices for such grids would contain upward of 512^4 nonzeroes. Such matrices typically are too large to fit in the computing platform's work memory, making efficient online computations unfeasible. Second, the cost of numerically integrating $\widetilde{\boldsymbol{x}}$ over time, i.e. the online costs of the stochastic methodology. The online cost of our stochastic methodology is dominated by the matrix vector products (MVPs) required to simulate $\widetilde{\boldsymbol{b}}$ (4.7). If we restrict the number of nonzero conditioning variables, the drift matrices A_i and D become sparse, e.g. $(K \times K)$-band matrices. This reduces the complexity of the drift MVPs in (4.7) to linear in K. The structure of the covariance has a different impact on the computational complexity of (4.7). The diagonal $(K \times K)$-matrix Σ_D gives linear (in K) complexity of the MVPs in (4.7). By contrast, the lower-triangular Σ_L gives $O(K^2)$ complexity of the MVPs in (4.7), causing a computational bottleneck for large K. Imposing sparsity (other than diagonality) on Σ in a statistically and dynamically consistent way is nontrivial yet important for systems with large K; we leave this topic for future study.

2.4 Comparison to other stochastic parameterizations

In this study we compare different stochastic parameterizations in terms of their effect on the long-term statistical behavior of the resolved model variables (see Section 5). Besides the VARX model proposed here, this comparison includes parameterizations based on AR(1) and on NARMAX processes that have been proposed before in the literature. For clarity, we label the different parameterizations with short descriptive names (e.g. (VARX(30) Σ_L)) instead of referring to equation numbers. We compare the following parameterizations:

- WN: white noise process. This is an "unconditioned" parameterization (no conditioning on $\widetilde{\boldsymbol{x}}$ or on past values of $\widetilde{\boldsymbol{b}}$). It is included as it represents the simplest stochastic model, and enables us to assess the merit of more complicated stochastic models.

- AR(1): autoregressive process, independently applied to each of the grid points k. Discussions on AR processes can be found in standard text books on time series analysis. In Arnold et al. [4], parameterization with AR(1) is proposed and discussed in more detail. They consider a parameterization consisting of both a deterministic and stochastic part: a regressed polynomial dependent on x (deterministic) and a one-step autoregression (stochastic) with varying options for noise models (we compare to their "additive" noise model). They show both that the stochastic parameterizations improve significantly over deterministic parameterizations and that the autoregression models are a major improvement over WN. We include this parameterization as it is a special case of the VARX models proposed here.

- VARX(p) Σ_D: vector autoregressive process with exogenous variable. We choose all matrices $A_i = 0$ for $i \neq p$ (see (4.4)) and we enforce sparsity by requiring the drift matrices A_p, D and the noise matrix Σ_D all to be diagonal. We choose a single nonzero drift matrix A_i to circumvent parameter estimation issues, as resolving these would require a study of itself, see Section 3.2.1 for a detailed discussion. As discussed above, imposing sparsity on the regression coefficient matrices is intended to limit the number of parameters and to make this parameterization approach more tractable for high-dimensional ocean and atmosphere models.

- VARX(p) Σ_L: similar as VARX(p) Σ_D, however with a lower triangular (non-diagonal) root covariance matrix Σ_L instead of a diagonal one (Σ_D). This allows us to explicitly model the cross-correlations between spatial points. Σ_L is not sparse; we leave the case of a non-diagonal but sparse covariance matrix for a follow-up study (nearing completion).

- NARMAX$_{1,2,0,1}$ and NARMAX$_{1,1,1,0}$: nonlinear autoregression moving average with exogenous input models, proposed for parameterization by Chorin and Lu [25]. The subscripts denote the values of parameters (p, r, s, q) that define the specific NARMAX structure (e.g. the number of endogenous variables, or the number of moving average terms). The NARMAX parameterization in Chorin and Lu [25] is applied independently to each grid point. Thus, NARMAX is scalar-valued, whereas VARX is vector-valued. When the matrices A_p, D and Σ are all multiples of the identity matrix, VARX can be seen as a specific case of NARMAX: in addition to the VARX description, NARMAX includes moving average noise and nonlinearities in the regressed terms.

While model selection for NARMAX (selecting the structure of nonzero model variables in its general form) is a nontrivial problem, we compare to the specific two NARMAX models proposed in Chorin and Lu [25]. These models were selected for the exact same test configuration as the unimodal configuration in this study (see Table 4.1) and the configuation in Crommelin and Vanden-Eijnden [30]. Here we test how these NARMAX models perform in case of the trimodal configuration. We refer to Chorin and Lu [25] for the extensive algorithmic details of the NARMAX parameterizations and model choices.

3 Lorenz '96 model

The 2-layer Lorenz '96 (L96) model [99] is frequently used to test and develop stochastic parameterizations. It was formulated as an idealized representation of atmospheric flow, but has similarities to various multiscale models. The L96 model equations from Lorenz [99] were reformulated in Fatkullin and Vanden-Eijnden [48] to explicitly express the time scale gap ϵ between the variables x_k and variables $y_{j,k}$:

$$\frac{d\,x_k}{dt} = x_{k-1}(x_{k+1} - x_{k-2}) - x_k + F + b_k \tag{4.8}$$

$$\frac{d\,y_{j,k}}{dt} = \frac{1}{\epsilon}\left[y_{j+1,k}(y_{j-1,k} - y_{j+2,k}) - y_{j,k} + h_y x_k\right] \tag{4.9}$$

$$\text{with } b_k := \frac{h_x}{J}\sum_{j=1}^{J} y_{j,k}, \tag{4.10}$$

where $k = 1, \ldots, K$ and $j = 1, \ldots J$ can be interpreted as spatial indices for the variables x_k and $y_{j,k}$ on a circle with constant latitude. Because of the circle's periodicity, the following periodic boundary conditions hold:

$$x_k = x_{k+K}, \quad y_{j,k} = y_{j,k+K}, \quad y_{j+J,k} = y_{j,k+1}. \tag{4.11}$$

3.1 Model parameter configurations

Generally, the variables x_k and $y_{j,k}$ are referred to as the "large-scale" and "small-scale" variables. When setting $\epsilon \ll 1$ there is clear time scale separation, with x_k and $y_{j,k}$ the fast and slow variables, respectively. Instead, we choose $\epsilon = 0.5$, so that no clear temporal scale gap exists, as is more realistic for oceanic and atmospheric flows (see also Crommelin and Vanden-Eijnden [30]). This choice also provides a more challenging setup for parameterizations because it does not allow for parameterization by averaging of the fast variables. We test two L96 model configurations, with different parameters, as detailed below. The parameter choices for these two configurations are also listed in Table 4.1 for clarity.

For the first configuration we follow the setup from Crommelin and Vanden-Eijnden [30] and Chorin and Lu [25], with parameters $(\epsilon, K, J, F, h_x, h_y) = (0.5, 18, 20, 10, -1, 1)$. This configuration of the L96 model results in a reference distribution for x_k that is unimodal and not too far from Gaussian (see, for example, Figure 4.5a). We refer to this as the unimodal configuration.

To put our suggested parameterization approach further to the test we also use a nonstandard configuration of the L96 model that we call the trimodal configuration. By increasing the forcing F, the number of spatial points K, and the feedback parameter from the fast to the slow scales h_x, the stationary distribution and dynamics of the L96 model become significantly more difficult to reproduce with the reduced model with stochastic parameterization (see Figure 4.5b for the stationary distribution of x_k). The model parameters that define the unimodal and trimodal configurations are listed in Table 4.1.

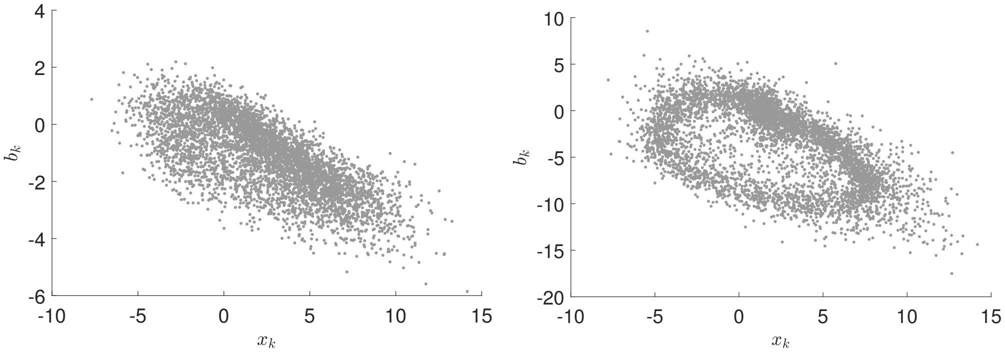
3.2 Stochastic model

The L96 system is ergodic [48] and invariant under spatial translations. The statistical properties of each x_k are identical. As a direct consequence, the cross-correlations are the same for each spatial point k, this satisfies the assumptions of the simple covariance form discussed in Section 2.2.

The sample data $(\boldsymbol{X}, \boldsymbol{B})$ of the two different deterministic L96 reference simulations reveal strong correlations between x_k and b_k, as illustrated for one such k in Figures 4.1a and 4.1b. Because the statistical properties of x_k are identical for all k, this figure is equivalent to that for any other k. Additionally, Figures

4.2a and 4.2b show that the conditional probability density function (CPDF) $P(b_k|x_k)$ can change significantly for different ranges of x_k-values. Therefore, because x_k is a resolved variable in both the deterministic and stochastic L96 models, x_k is a valuable predictor variable for (the distribution of) b_k. Clearly, the presence of a good predictor is not guaranteed, and identifying one may be nontrivial for some problems or application fields. For ocean modeling, this was explored in [13, 117, 160].

Because the (conditional) distributions of b_k in Figure 4.2a resemble normal distributions, we assume the underlying distribution of b_k to be Gaussian. This starting point will test the robustness of our parameterization, because the trimodal configuration exhibits distinct multi-modality in b_k. While Figure 4.1b does suggest a clear correlation between x_k and b_k for the trimodal configuration, there is a distinct circular pattern present in the scatter plot. The marginal distributions of b_k and x_k are also clearly trimodal, see Figures 4.2b and 4.5b, respectively.



(a) (b)

Figure 4.1: Strong correlation between x_k and b_k shown by scatter plots for the reference deterministic L96 eqs. (4.8)–(4.10): (a) unimodal, and (b) trimodal configurations.

The stochastic L96 model is obtained by forcing a reduced version (without b_k) of (4.8) with the VARX $\widetilde{\boldsymbol{b}}$ (4.7) that aims to approximate b_k in each k. Following the model reduction approach as described in Section 2, the stochastic L96 model then becomes:

$$\widetilde{x}_k^{n+1} = T\left(\Delta t,\, \widetilde{x}_{k-1}^n(\widetilde{x}_{k+1}^n - \widetilde{x}_{k-2}^n) - \widetilde{x}_k^n + F + \widetilde{b}_k^n\right) \tag{4.12}$$

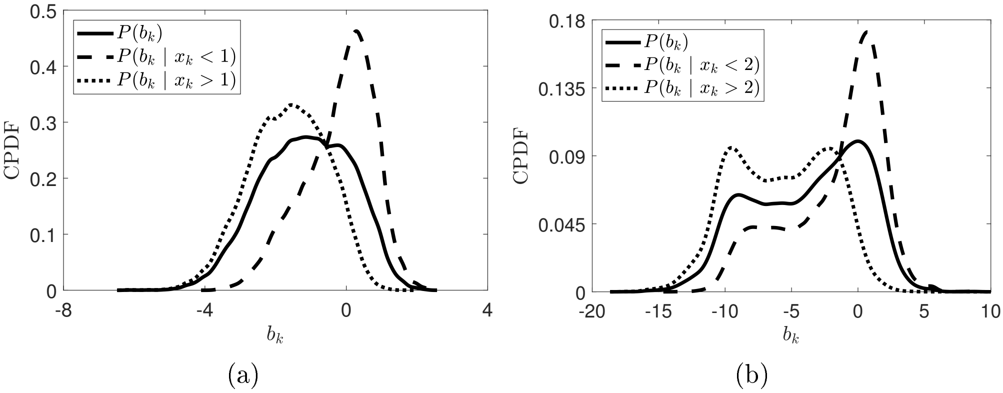


Figure 4.2: CPDFs for b_k dependent on x_k for the reference deterministic reference L96 eqs. (4.8)–(4.10): (a) unimodal, and (b) trimodal configurations

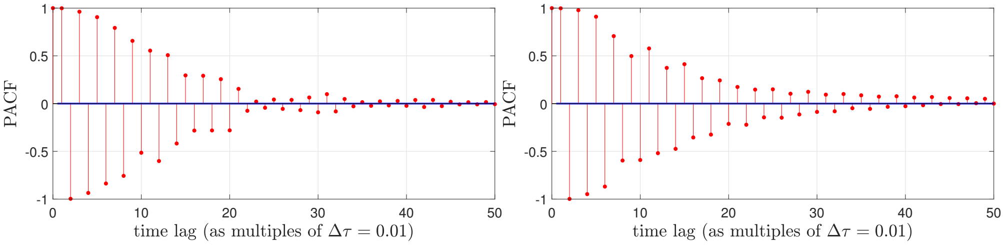
$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_1\widetilde{\boldsymbol{b}}^{n-1} + \cdots + A_p\widetilde{\boldsymbol{b}}^{n-p} + D\widetilde{\boldsymbol{x}}^n + \Sigma\boldsymbol{\xi}^n, \tag{4.13}$$

where T is the numerical integration scheme of choice (see Section 4).

3.2.1 Order selection - lag time choice

The order selection of the VARX(p) $\widetilde{\boldsymbol{b}}$ in (4.4) determines the temporal decorrelation of the VARX. By choosing the order p appropriately, one can match the stochastic model with the decorrelation timescale of the reference model. In this section, let us consider the choice for nonzero lag times p, i.e. the order selection for the VARX(p). The reference x_k has strongly oscillating, slowly decaying correlations (see also Figures 4.6a and 4.6b later on). To model this behavior perfectly one would need a high order VARX process. However, estimating a VARX(1) that is numerically stable is rather straightforward whereas estimating a stable VAR(p) of arbitrary order p is difficult due to the constraint (4.5). This constraint can only be verified a posteriori; we are not aware of estimation methods that guarantee (4.5) is satisfied a priori. Therefore, we opt for a single nonzero A_p and, in doing so, interpret the process as VARX(1) over an interval p times larger than the sampling interval. This leaves us with the choice for the nonzero lag contribution p.

Because we choose the coefficient matrices A_p and D diagonal (see Section 2.3), connections to univariate autoregressive (AR) models are easily made, particularly in the case of Σ_D (as it is also diagonal). In univariate time series



(a)

(b)

Figure 4.3: The partial autocorrelation functions (PACFs) for b_k for **(a)** the unimodal and **(b)** the trimodal "resolving" deterministic reference L96 simulations (see Table 4.1).

analyses, the order p of AR(p) models is often determined with the Box–Jenkins method [21]. Both the ACF and partial autocorrelation function (PACFs) are used to determine the order of an AR model for approximating timeseries data. Here the partial autocorrelation of lag n' is the autocorrelation between b_k^n and $b_k^{n+n'}$ that is not accounted for by lags 1 through $n'-1$, i.e. the partial autocorrelation is a conditional correlation that controls for all shorter lags:

$$\mathrm{PACF}(b_k, l) = \frac{\mathbb{C}ov(b_k^n, b_k^{n-l} \mid b_k^{n-1}, \ldots, b_k^{n-l+1})}{\sqrt{\mathbb{V}\mathrm{ar}(b_k^n \mid b_k^{n-1}, \ldots, b_k^{n-l+1})\mathbb{V}\mathrm{ar}(b_k^{n-l} \mid b_k^{n-1}, \ldots, b_k^{n-l+1})}} \tag{4.14}$$

Although the *sample* (P)ACFs do not necessarily describe the same autoregressive properties as the *analytical* (P)ACFs, they typically are used in model selection. It is common practice that when the ACF shows sinusoidal behavior with no clear decay to 0 (as is the case for both the unimodal and trimodal deterministic reference simulations, see Figures 4.6a–4.6b), the order p of the modeling AR(p) is chosen at the last spike in the PACF after which the PACF no longer returns to this same level [83]. The PACFs of the "resolving" unimodal and trimodal b_k are plotted in Figure 4.3. The PACF in Figure 4.3a shows such a spike around $\Delta\tau = 0.14$, hence we pick $p = 14$ for the unimodal case. The PACF in the trimodal case, Figure 4.3b, decays very gradually, showing no clear steep monotonous decline. This necessitates, according the Box–Jenkins method, the choice for a relatively long time scale of approximately $\Delta\tau = 0.3$, i.e. $p = 30$.

4 Practical implementation of stochastic parameterization

The stochastic L96 model Equations (4.12)–(4.13) is forced by the VARX model $\widetilde{\boldsymbol{b}}$ in (4.13) dependent on exogenous ($\widetilde{\boldsymbol{x}}$) and endogenous (past $\widetilde{\boldsymbol{b}}$) states. The reference data $(\boldsymbol{X}, \boldsymbol{B})$ is used to approximate appropriate CPDFs from which $\widetilde{\boldsymbol{b}}$ is sampled, determined by the selection of endogenous and exogenous variables. For example, if we select D and A_1 to be scalar matrices and $A_i = 0$ for $i > 1$, then by (4.13) $\widetilde{\boldsymbol{b}}^n$ is sampled from the Gaussian approximation of the CPDF $P(\boldsymbol{b}^n \mid \boldsymbol{x}^n = \widetilde{\boldsymbol{x}}^n, \boldsymbol{b}^{n-1} = \widetilde{\boldsymbol{b}}^{n-1})$, or $P(\boldsymbol{b}^n \mid \widetilde{\boldsymbol{x}}^n, \widetilde{\boldsymbol{b}}^{n-1})$ for short.

We solve the L96 system directly using a classical second-order Runge–Kutta integration scheme [48]. The regression coefficients and covariance matrix of the VARX (4.13) are precomputed with least squares. The VARX is integrated over time together with the L96 system and applied to the timestepping of $\widetilde{x}_k$ (4.12). The pseudo-code for our stochastic L96 model is shown in Figure 4.4. All parameters used in our deterministic and stochastic simulations are listed in Table 4.1.

Table 4.1: Parameter settings for all deterministic and stochastic L96 models

Parameter	Explanation	unimodal L96	trimodal L96
ϵ	scale separation	0.5	0.5
K	# discretized large-scale spatial points / # resolved x-variables	18	32
J	# discretized small-scale spatial points / # unresolved y-variables per x-variable	20	16
F	forcing on the x variables	10	18
h_x	scale coupling constant	-1	-3.2
h_y	scale coupling constant	1	1
Δt	integration time step full L96 model	10^{-3}	10^{-3}
$\Delta\tau$	integration time step reduced L96 model	10^{-2}	10^{-2}
δt	sampling interval	10^{-2}	10^{-2}
N	Number of integration time steps in a simulation	$10^6 + p$	$10^6 + p$

We choose the sampling interval δt of the reference data $(\boldsymbol{X}, \boldsymbol{B})$ to be larger than the integration time step Δt of the full L96 model. We pick $\delta t = 10\Delta t$, same as in Crommelin and Vanden-Eijnden [30]. This reduces the amount of data that must be handled, at the price of loosing some high-frequency (short timescale) information. However, as we set the integration time step of the reduced model equal to the sampling interval, i.e. $\Delta\tau = \delta t$ (see Table 4.1),

```
input:   $\boldsymbol{X}$   : concatenated vector of sample data for $x_k^n$, size $NK \times 1$.
         $\boldsymbol{B}$   : concatenated vector of sample data for $b_k^n$, size $NK \times 1$.

/* Precompute the VARX coefficients $\boldsymbol{a}_0, A_i$, and $D$ */
$\boldsymbol{Z} = (\boldsymbol{1}, \boldsymbol{X}(\text{lag } 0, \boldsymbol{B}(\text{lag } 1), \dots, \boldsymbol{B}(\text{lag } p))$, the regressor variable matrix, size $K(N-p) \times (p+2)$
$(\boldsymbol{a}_0, D, A_1, \dots, A_p) = (\boldsymbol{Z}^T \boldsymbol{Z})^{-1} \boldsymbol{Z}^T \boldsymbol{B}$

/* Either compute $\Sigma_D$ */
$\boldsymbol{R} = \text{VEC}(\boldsymbol{b}^n - \boldsymbol{a}_0 - A_1 \boldsymbol{b}^{n-1} - \dots - A_p \boldsymbol{b}^{n-p} - D\boldsymbol{x}^n)$, where $\boldsymbol{R}$ the
                                            concatenated residual vector $(K(N-p) \times 1)$
$\Sigma_D = \sqrt{\mathbb{V}\text{ar}(\boldsymbol{R})}\, I_{K \times K}$

/* Or compute $\Sigma_L$ */
/* Let $[\boldsymbol{R}]$ denote the reshaped $((N-p) \times K)$-matrix corresponding to $\boldsymbol{R}$ */
$\Sigma_L = \text{Chol}(\mathbb{C}ov([\boldsymbol{R}]))$, where $\mathbb{C}ov([\boldsymbol{R}])$ a $(K \times K)$-matrix

$(\widetilde{\boldsymbol{x}}^{-p+1,\dots,0}, \widetilde{\boldsymbol{b}}^{-p+1,\dots,0}) = (\boldsymbol{x}^{-p+1,\dots,0}, \boldsymbol{b}^{-p+1,\dots,0})$
for $i := 0$ to $N-1$ do
    /* Sample $\widetilde{\boldsymbol{b}}$ */
    $\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_1 \widetilde{\boldsymbol{b}}^{n-1} + \dots + A_p \widetilde{\boldsymbol{b}}^{n-p} + D\widetilde{\boldsymbol{x}}^n + \Sigma \boldsymbol{\xi}^n$, where $\boldsymbol{\xi}^n \sim \mathcal{N}(0, \boldsymbol{I})$

    /* Update $\widetilde{x}$ with second order Runge–Kutta, notation: let $\boldsymbol{x}_{+d}$ denote the
        module rotation of points $x_k$, e.g. $\boldsymbol{x}_{+1} := (x_2, \dots, x_K, x_1)$) */
    $\widetilde{\boldsymbol{x}}' = \widetilde{\boldsymbol{x}}^n + \frac{\Delta\tau}{2}(\widetilde{\boldsymbol{x}}^n_{-1}(\widetilde{\boldsymbol{x}}^n_{+1} - \widetilde{\boldsymbol{x}}^n_{-2}) - \widetilde{\boldsymbol{x}}^n + F + \widetilde{\boldsymbol{b}}^n)$
    $\widetilde{\boldsymbol{x}}^{n+1} = \widetilde{\boldsymbol{x}}^n + \Delta\tau(\widetilde{\boldsymbol{x}}'_{-1}(\widetilde{\boldsymbol{x}}'_{+1} - \widetilde{\boldsymbol{x}}'_{-2}) - \widetilde{\boldsymbol{x}}' + F + \widetilde{\boldsymbol{b}}^n)$
endfor
```

Figure 4.4: Algorithm for the time integration of the stochastic L96 model. Notation: $\boldsymbol{B}$(lag i) is the vector of concatenated sample data $\boldsymbol{b}^j$ for all $p+1-i \leq j \leq N-i$, i.e. the sample data of b_k at i time steps in the past (where the first $p-i$ vectors $\boldsymbol{b}$ of $\boldsymbol{B}$ are skipped to make each $\boldsymbol{B}$(lag i) equal in length). Similar notation is used for $\boldsymbol{X}$(lag 0) to denote the vector of concatenated sample data $\boldsymbol{x}^j$ for all $p+1 \leq j \leq N$.

these are very short timescales that are not resolved by the reduced model anyway.

5 Numerical results

In this section we compare the statistical behavior of the reduced model with VARX stochastic parameterization Equations (4.12)–(4.13) with the reference model Equations (4.8)–(4.9). Recall from Section 3 that the statistics of x_k are identical for all k. Therefore, the statistical properties determined for x_k describe the full statistics of $\boldsymbol{x}$, i.e. equal for all k. Let $\mu := \mathbf{E}(x_k)$ and $\sigma := \sqrt{\mathbf{E}((x_k)^2) - \mathbf{E}(x_k)^2}$ denote the mean and standard deviation of x_k, respectively, where $\mathbf{E}$ denotes the average over time. We assess the following statistical criteria of the variable of interest x of the models:

- The probability density function (PDF) of x_k.
- The autocorrelation coefficient (ACF) of x_k: $\mathrm{ACF}(\tau) := \sigma^{-2}\mathbf{E}\left[(x_k^t - \mu)(x_k^{t+\tau} - \mu)\right]$.
- The cross-correlation coefficient (CCF) between x_k and x_{k+1}: $\mathrm{CCF} := \sigma^{-2}\mathbf{E}\left[(x_k^t - \mu)(x_{k+1}^t - \mu)\right]$.
- The mean wave amplitude $\mathbf{E}(|u_m|)$ for each wave number $0 \leq m \leq K/2$, where a time series for the wavenumber vector $\boldsymbol{u} := \widehat{\boldsymbol{x}}$ is obtained by calculating the Fourier transform of $\boldsymbol{x}$ at every time step.
- The wave variance $\mathbf{E}(|u_m - \mathbf{E}(u_m)|^2)$,

For the VARX model in (4.13) we use several different settings, each described explicitly in the following subsections. We show a representative selection of the results. All VARX models have a single nonzero A_p for chosen lag time p to circumvent VARX estimation stability issues (as discussed earlier). Furthermore, all VARX models until Section 5.3.2 have a diagonal covariance structure, i.e. a diagonal matrix Σ_D. To reduce the number of parameters we choose the coefficient matrices A_p and D to be diagonal in all cases.

First, in Section 5.1, we illustrate for completeness the contrast between the deterministic reference L96 simulations and simulations with the simplest possible stochastic model, denoted (*WN*), in which the $\widetilde{b}_k$ are independent white noise terms. Then, in Section 5.2 we discuss results for stochastic model simulations with a single regressor: either only endogenous (*Multi* AR(1)) or exogenous (*WND*), respectively. Next, we demonstrate that with both regressors (VARX(14) Σ_D) (called "double regressor") the unimodal L96 reference statistics are reproduced very accurately in Section 5.3.1. However, we also show that (VARX(30) Σ_D) does not perform well in case of the trimodal L96 model configuration. In Section 5.3.2, we therefore compare the (VARX(30) Σ_D) and (VARX(30) Σ_L) simulations, and show that by allowing for a non-diagonal structure of the covariance we also succeed at reproducing the statistics of the trimodal L96 model accurately. In section 5.4, we compare the results for our

VARX models to those for the NARMAX models proposed in Chorin and Lu [25]. While the NARMAX models perform very accurately for the unimodel L96 test case, we show that the NARMAX models do not perform well for the trimodal L96 configuration. Neither the trimodal distribution of x_k nor the wave statistics were reproduced accurately. All our simulations here use the parameter configurations as listed in Table 4.1.

5.1 White noise parameterizations

We start with an 'unconditioned' stochastic parameterization, that is to say a parameterization in which $\widetilde{b}_k^n$ is not conditioned on its own past state(s) nor on $\widetilde{\boldsymbol{x}}^n$:

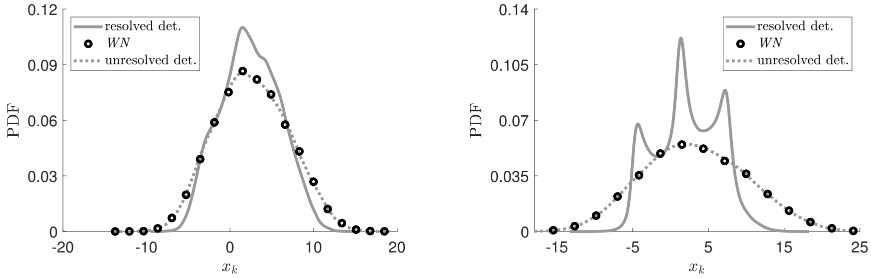
$$\widetilde{\boldsymbol{b}}^n = \sigma I \, \boldsymbol{\xi}^n, \tag{WN}$$

where $\boldsymbol{\xi}^n$ is a vector of independent normally distributed random variables. Note that this model is equivalent to choosing $A_0, \ldots, A_p, D = 0$ and $\Sigma = \sigma I$ in (4.13). In this simplest possible stochastic parameterization, the time evolution of each $\widetilde{b}_k$ is a series of Brownian motion increments, i.e. a white noise process, therefore we denote it (WN). We include it here to verify the added value of conditioning in the more complicated parameterizations discussed later on.

In Figures 4.5a and 4.5b we plot the distributions of x_k for the two L96 model configurations. First, the "resolved" reference simulation obtained with the full L96 model Equations (4.8)–(4.9), second, the "unresolved" reference simulation, i.e. (4.8) with $b_k = 0$. The former is what we aim to reproduce with our reduced models. The latter of the two we include as a worst-case reference, the result of a reduced model with no parameterization at all to account for the missing unresolved scales.

The overall shape of the distribution of x_k in the unimodal L96 model is reproduced (although the details are not well captured), both with the (WN) parameterization and without any parameterization (the "unresolved" case), see Figure 4.5a. This result is in line with, e.g, Crommelin and Vanden-Eijnden [30] and Chorin and Lu [25], where it was also found that the distribution of x_k is not very difficult to reproduce with a reduced model, in case of the L96 unimodal configuration. However, Figure 4.5b shows that the distribution of x_k for the trimodal L96 configuration is not reproduced at all by (WN), nor by the "unresolved" case.

Interestingly, Figures 4.5 and 4.6 show that the (WN) parameterization introduces no significant changes to the long-term statistics of the "unresolved" model without any parameterization. Thus, the perturbations of the white noise are not able to alter the dynamics of the "unresolved" model.



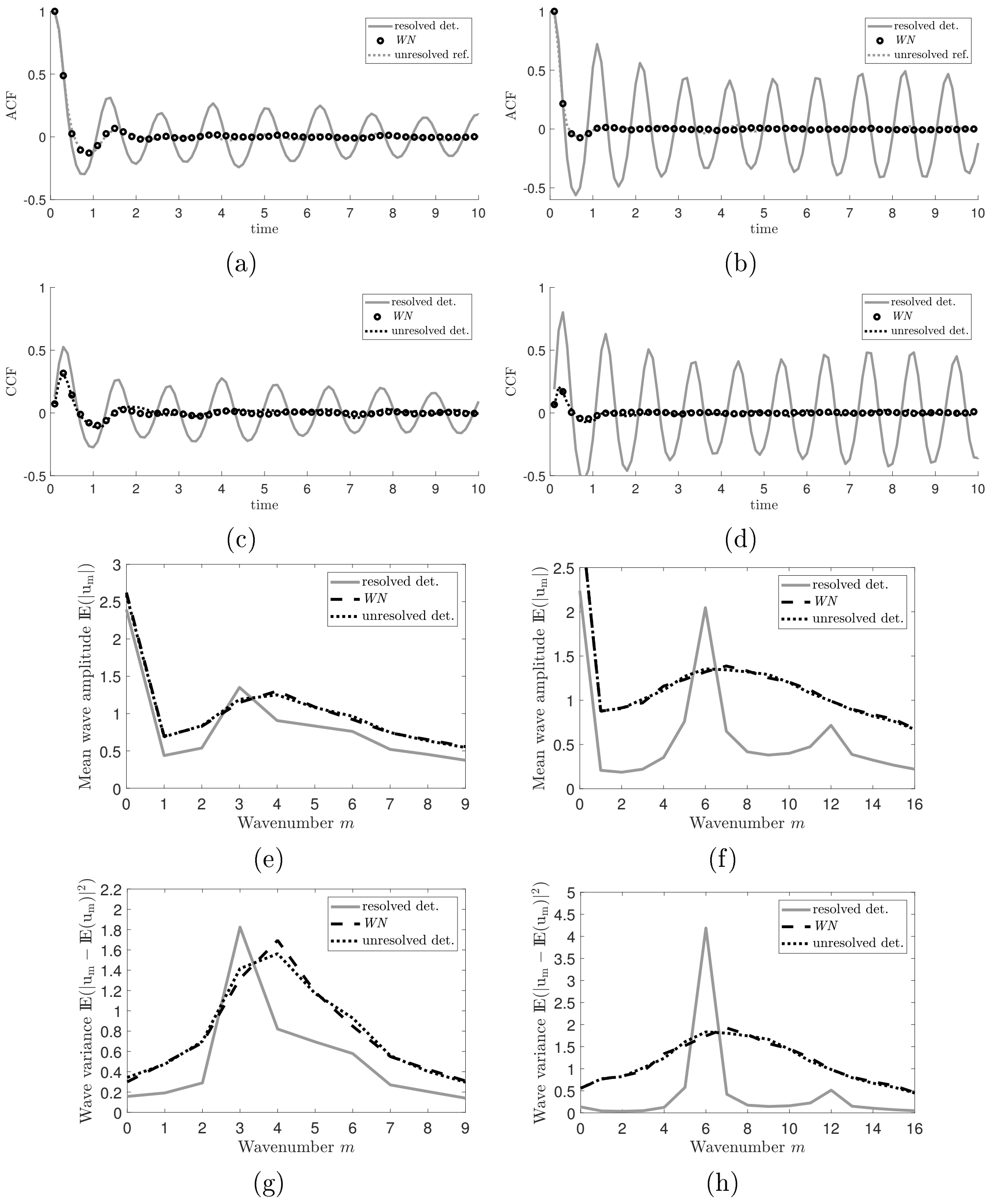
(a) (b)

Figure 4.5: Comparison between PDFs of interest for **(a)** the unimodal L96 and **(b)** the trimodal L96 configuration (see Table 4.1) resulting from unconditioned stochastic simulations using (WN), as well as "resolving" and "unresolving" ($b_k = 0$) deterministic reference simulations.

Similarly, the ACF, CCF, and wave criteria are not reproduced to any satisfactory degree with (WN), see Figures 4.6a-4.6h. In Figures 4.6a and 4.6b one sees that the reduced model with (WN) exhibits ACFs that are very similar to those of the unresolved deterministic model; neither the amplitudes nor the long decorrelation scales shown by the resolved deterministic simulation are reproduced. The CCFs in Figures 4.6c and 4.6d show the same problems. The mean wave amplitudes and wave variances of (WN) in Figures 4.6e-4.6f and Figures 4.6g-4.6h, respectively, show that the reduced models have more uniform spread over the larger wave numbers and do not peak at the correct wavenumbers, compared to the resolved deterministic model.

5.2 Single regressor parameterizations

Next, we consider reduced model simulations with single regressors for the VARX models. We expect that the state-dependency, temporally correlated mean, and exogenous predictor variable of the VARX model will improve the performance, capturing more of the features from the resolving L96 reference simulation.



(a) (b)

(c) (d)

(e) (f)

(g) (h)

Figure 4.6: Comparison between ACFs **(a)**-**(b)**, CCFs **(c)**-**(d)**, wave mean amplitude **(e)**-**(f)**, and wave variance **(g)**-**(h)** for the unconditioned stochastic simulations (*WN*) as well as the “resolving” and “unresolving” deterministic reference simulations for unimodal **(a)**, **(c)**, **(e)**, and **(g)** and trimodal **(b)**, **(d)**, **(f)**, **(h)** configurations.

First, let us consider the autoregressive model (*Multi* AR(1)) consisting of multiple independent AR(1) processes due to its diagonal drift coefficient matrix (A_1):

$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_1\widetilde{\boldsymbol{b}}^{n-1} + \Sigma_D\boldsymbol{\xi}^n. \tag{\textit{Multi} AR(1)}$$

Second, we consider a vector of independent white noise processes with drift:

$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + D\widetilde{\boldsymbol{x}}^n + \Sigma_D\boldsymbol{\xi}^n, \tag{\textit{WND}}$$

As can be seen from the criteria plotted in Figure 4.7, the (*Multi* AR(1)) model does not significantly improve over (*WN*) (cf. Figures 4.5a–4.6b). By contrast, Figure 4.7a shows that the (*WND*) model reproduces the unimodal distribution of x_k significantly better than the (*WN*) model. This is due to the x_k dependence of (*WND*). It suggests that the exogenous variable $\boldsymbol{x}$ indeed holds predictive value for $\mathcal{B}_{xy}$ (as suggested in Section 1). Also, the (*Multi* AR(1)) model is independent of $\widetilde{\boldsymbol{x}}$, unlike (*WND*). However, while (*WND*) reproduces the distribution of x_k accurately in the unimodal case (Figure 4.7a), it fails to do so in the trimodal case (Figure 4.7b). Furthermore, (*WND*) improves only slightly on the ACFs of x_k when compared to (*WN*) (see Figure 4.7c). These same conclusions are reached for the CCFs and wave criteria (not shown). To introduce more spatio-temporal consistency in the VARX we test combinations of endogenous and exogenous regressors in the next section.

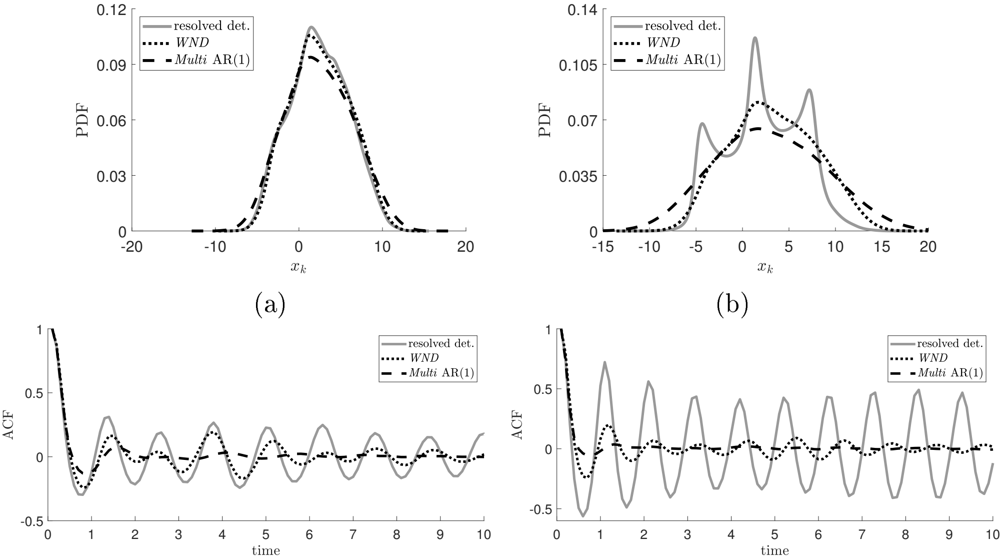
5.3 Double regressor parameterizations

5.3.1 Diagonal covariance

As motivated in Section 3.2.1, we suggest the (VARX(14) Σ_D) model here for parameterization in the case of the unimodal L96 configuration:

$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_{14}\widetilde{\boldsymbol{b}}^{n-14} + D\widetilde{\boldsymbol{x}}^n + \Sigma_D\boldsymbol{\xi}^n. \tag{VARX(14) Σ_D}$$

Figure 4.8 shows that the state-dependence and temporal correlation introduced by D and A_{14} in (VARX(14) Σ_D) result in near-perfect approximations of the reference statistics. Not only does the distribution of $\widetilde{x}_k$ match perfectly to the reference (Figure 4.8a), but also the wave criteria (Figures 4.8b and 4.8c) and correlations (Figures 4.8d and 4.8e) match almost exactly. We emphasize



(a) (b)

(c) (d)

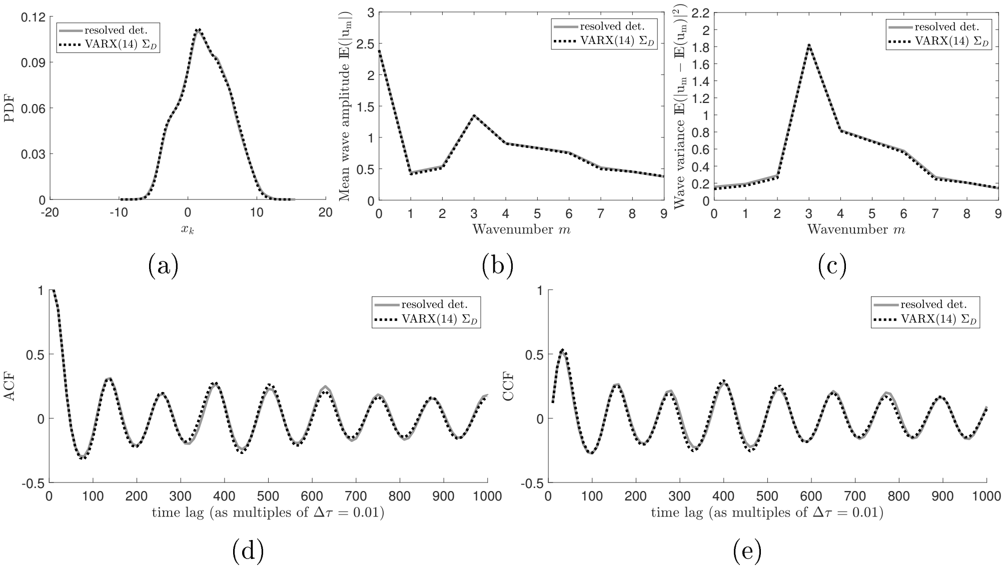
Figure 4.7: Comparison between **(a)**-**(b)** the PDFs and **(c)**-**(d)** the ACFs of interest for both the unimodal and trimodal L96 configurations (see Table 4.1), respectively, resulting from stochastic simulations with single regressor variables.

the accuracy of the reproduced long sinusoidal decorrelation structure visible in 4.8d and 4.8e, a particularly challenging feature of the reference L96 simulations.

However, this strong performance does not extend fully to the trimodal L96 configuration. For this configuration we suggested $p = 30$ in Section 3.2.1, i.e. the following (VARX(30) Σ_D) model:

$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_{30}\widetilde{\boldsymbol{b}}^{n-30} + D\widetilde{\boldsymbol{x}}^n + \Sigma_D\boldsymbol{\xi}^n. \qquad \text{(VARX(30) } \Sigma_D\text{)}$$

The results with this model for parameterization are shown in Figure 4.9. The PDF of x_k (Figure 4.9a), the wave mean (4.9b), the wave variance (4.9c) and ACF (4.9d) are qualitatively correct, but not fully accurate. For example, the wave variance (Figure 4.9c) has peaks at wavenumbers 6 and 12 that are too high. Also, the oscillation periods of the ACF and CCF are too long (by circa 10%) with the reduced model.



(a) (b) (c)

(d) (e)

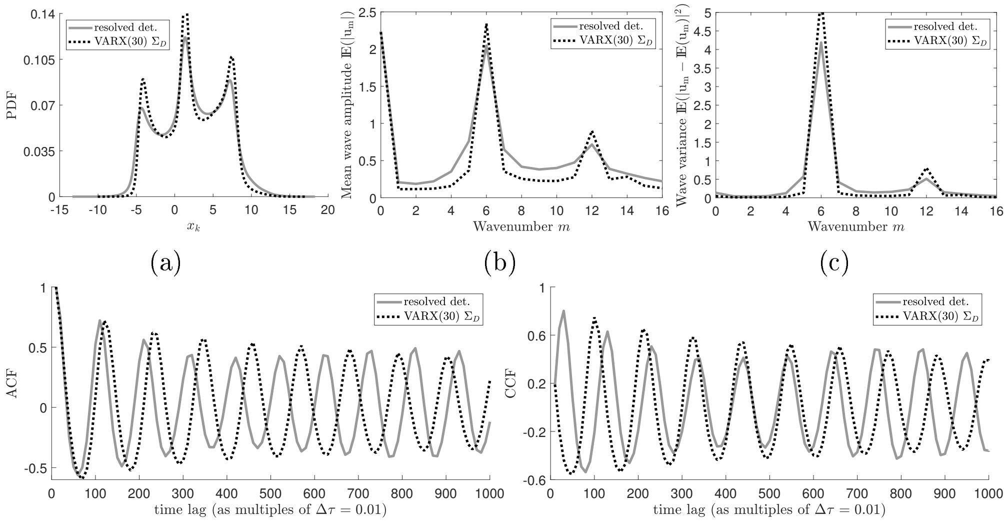
Figure 4.8: Comparison of **(a)** PDFs, **(b)** mean wave amplitude, **(c)** wave variance, **(d)** ACFs, and **(e)** CCFs of the reduced model using (VARX(14) Σ_D) and of the unimodal deterministic reference.

5.3.2 Fully dense covariance

The trimodal L96 configuration has strongly non-Gaussian features, making this a particularly challenging test case for our approach to use VARX (i.e. Gaussian) processes for parameterization. As displayed in Figures 4.5b and 4.7b, the trimodal nature of the PDF for x_k is not captured at all with the (*WN*), (*WND*) and (*Multi* AR(1)) parameterizations. The results with (VARX(30) Σ_D) in the previous section are a major improvement. In this section we aim to improve further by using a fully-dense covariance matrix $\Sigma_L \Sigma_L^T$ instead of a diagonal one, as described in Section 2.2:

$$\widetilde{\boldsymbol{b}}^n = \boldsymbol{a}_0 + A_{30}\widetilde{\boldsymbol{b}}^{n-30} + D\widetilde{\boldsymbol{x}}^n + \Sigma_L \boldsymbol{\xi}^n. \qquad \text{(VARX(30) } \Sigma_L\text{)}$$

Figure 4.10 shows the results using (VARX(30) Σ_L). The trimodal structure in the PDF of x_k is reproduced accurately, as shown in Figure 4.10a. The main deviation from the trimodal L96 reference is a slightly higher kurtosis in the PDF for $\widetilde{x}_k$. Furthermore, the oscillations in the ACF and CCF have somewhat shorter period compared to those resulting from (VARX(30) Σ_D), and align



(a) (b) (c)

(d) (e)

Figure 4.9: Comparison of **(a)** PDFs, **(b)** mean wave amplitude, **(c)** wave variance, **(d)** ACFs, and **(e)** CCFs of the reduced model using (VARX(30) Σ_D) and of the trimodal deterministic reference.

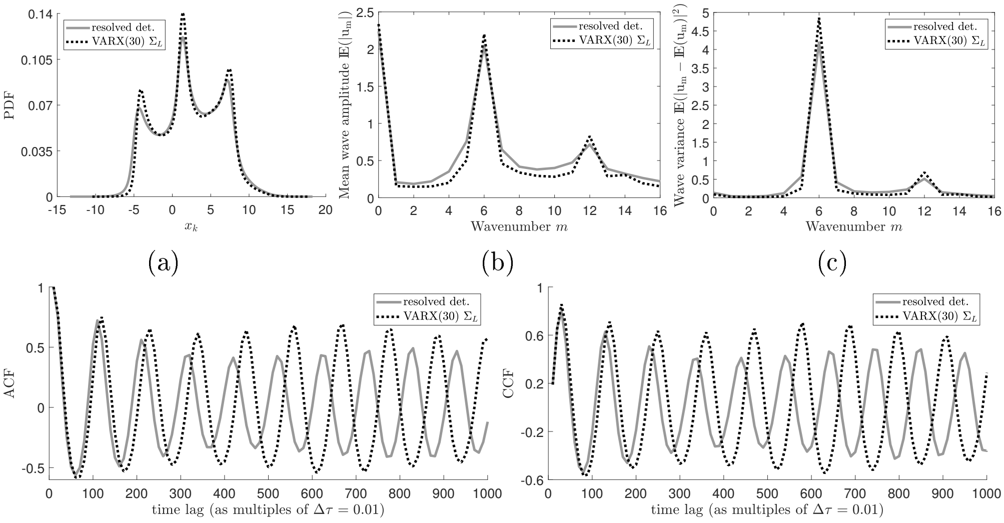
better (albeit not perfectly) with the reference trimodal L96 model, compare in particular Figures 4.9e and 4.10e to see an improved CCF reproduction. Finally, the mean amplitude and variance of most wave numbers differ only slightly from the reference values in Figures 4.10b and 4.10c. Altogether, the results, while not perfect, are very satisfactory for this highly challenging test case.

5.4 NARMAX parameterizations

As motivated in Section 2.4, we compare the VARX parameterizations from Sections 5.2 and 5.3 to the NARMAX parameterization proposed in Chorin and Lu [25]. Specifically, we compare to the performance of the two configurations of NARMAX used in Chorin and Lu [25] defined by the function Φ (for further details on the NARMAX description see Chorin and Lu [25]):

$$\Phi^n = \mu + a_1 z^{n-1} + b_{1,1} x^{n-1} + b_{2,1} x^{n-2} + d_1 \xi^{n-1}, \qquad (\mathrm{NARMAX}_{1,2,0,1})$$

$$\begin{aligned}\Phi^n = \mu &+ a_1 z^{n-1} + b_{1,1} x^{n-1} + b_{1,2}(x^{n-1})^2 + \\ &b_{1,3}(x^{n-1})^3 + c_{1,1}(R_\delta(x^{n-1})), \qquad (\mathrm{NARMAX}_{1,1,1,0})\end{aligned}$$



(a) (b) (c)

(d) (e)

Figure 4.10: Comparison of **(a)** PDFs, **(b)** mean wave amplitude, **(c)** wave variance, **(d)** ACFs, and **(e)** CCFs of the reduced model using (VARX(30) Σ_L) and of the trimodal deterministic reference.

where ξ^n are independent Gaussian random variables with zero mean and variance σ^2, $R_\delta(x)$ represents the resolved features of the L96 model that are only dependent on x, and $\mu, \sigma^2, a_i, b_i, c_i, d_i$ are the parameters to be estimated. The NARMAX parameterization is applied independently to each grid point k. Because the L96 model is spatially homogeneous, the estimated NARMAX parameters are equal for all grid points k.

Chorin and Lu [25] show that the NARMAX models above perform very well for the unimodal L96 configuration (see Table 4.1), using different sampling intervals. The (NARMAX$_{1,2,0,1}$) model gives good results with $\delta t = 10^{-2}$, whereas (NARMAX$_{1,1,1,0}$) performs well with $\delta t = 5 \cdot 10^{-2}$. It is not dicussed in Chorin and Lu [25] how these specific configurations of NARMAX were selected. The choice of configuration is important though: we applied (NARMAX$_{1,1,1,0}$) to the case with $\delta t = 10^{-2}$ (including re-estimation of parameters) and found it to be less accurate than (NARMAX$_{1,2,0,1}$) (results not shown).

Analogous to the tests in Section 5.3 we test the performance of the NARMAX models also with the trimodal L96 configuration (see Table 4.1). The estimated model parameters resulting from the maximum likelihood estimation (see Chorin and Lu [25]) are shown in Table 4.2.

Figure 4.11 shows that the NARMAX models have comparable performance for the trimodal L96 configuration. Neither of the NARMAX models reproduces the trimodal distribution of x_k accurately, as shown in Figure 4.11a. However, they do reproduce accurately the mean and variance of the distribution.

Table 4.2: Estimated parameters in the NARMAX models for $\delta t = 0.01$

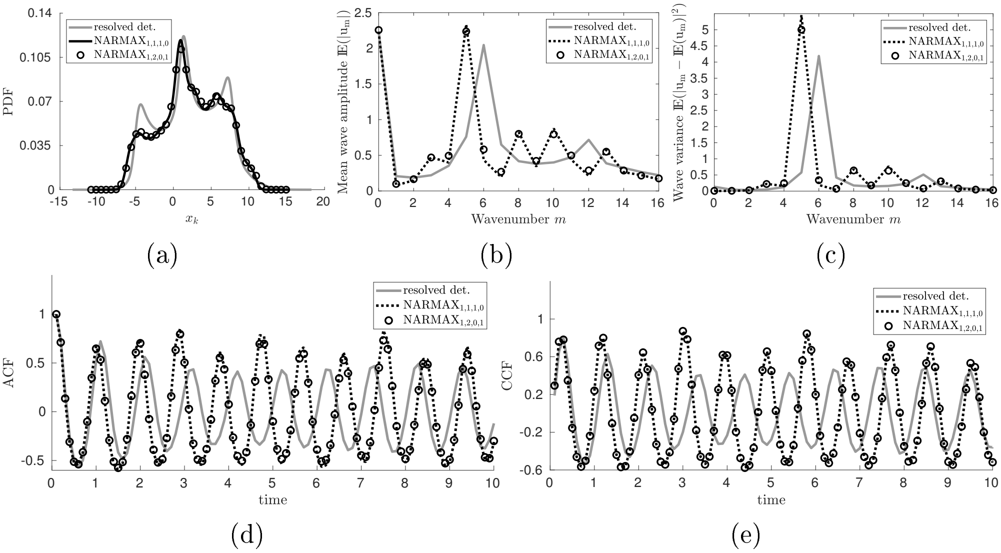
(NARMAX$_{1,2,0,1}$)	a_1	$b_{1,1}$	$b_{2,1}$	d_1		μ	σ^2
	0.9780	-0.1276	0.1134	0.9998	-	0.0096	0.0028
(NARMAX$_{1,1,1,0}$)	a_1	$b_{1,1}$	$b_{1,2}$	$b_{1,3}$	$c_{1,1}$	μ	σ^2
	0.9729	-0.0669	-0.0001	0.0001	-0.0028	0.0467	0.0106

Figures 4.11b and 4.11c show that the wave statistics are also not reproduced accurately. The most prominent peak at wavenumber 5 is shifted, and some of the higher wavenumbers have overestimated mean and variance. For the correlation functions (ACF and CCF), both (NARMAX$_{1,2,0,1}$) and (NARMAX$_{1,1,1,0}$) result in oscillations with periods that are somewhat too short (Figures 4.11d and 4.11e), whereas the VARX models in section 5.3 gave periods that are a bit too long in the trimodal case (e.g. Figure 4.10).

Overall, the VARX models (in particular (VARX(30) Σ_L)) show better performance on the trimodal test case than the NARMAX models, with more accurate reproduction of the PDF and wave statistics. It must be noted that although we estimated the parameters of the NARMAX models specifically for the trimodal test case (see Table 4.2), we did not alter their configurations (i.e., the parameters p, r, s, q that determine the structure of the NARMAX model). A different NARMAX configuration may be more optimal for the trimodal test case, however we have no guidance on how to select such a configuration.

6 Discussion

In this study we proposed a method for data-driven stochastic parameterization using vector autoregressive processes with exogenous variable (VARX). This method is used to parameterize the feedback from unresolved processes in reduced models of multiscale dynamical systems. The choice for VARX is aimed specifically at spatially extended dynamical systems, for which it is important to capture spatial correlations, while keeping the number of parameters that



(a) (b) (c)

(d) (e)

Figure 4.11: Comparison of **(a)** PDFs, **(b)** mean wave amplitude, **(c)** wave variance, **(d)** ACFs, and **(e)** CCFs of the NARMAX models proposed in Chorin and Lu [25] and of the trimodal deterministic reference.

must be estimated from data as low as possible.

We tested the proposed VARX parameterization method on the 2-layer L96 model (4.8) - (4.10), replacing the feedback vector $\boldsymbol{b}$ by a VARX $\widetilde{\boldsymbol{b}}$ so that the "small-scale" variables $y_{j,k}$ no longer had to be resolved. The process $\widetilde{\boldsymbol{b}}$ was trained to emulate the dynamical effects of $\boldsymbol{b}$. With a proper formulation of $\widetilde{\boldsymbol{b}}$ the simulations of the reduced model were able to reproduce the statistical criteria of the reference simulation accurately. We note that these criteria focus on long-term statistical properties rather than on the accuracy of short-term predictions.

The stochastic approach formulated in this study was developed with the aim to limit the amount of required computer memory and number of parameters, as these can become computational bottlenecks in large, spatially extended systems (see e.g. Verheul et al. [145]). To this end, we modeled the VARX models with diagonal coefficient matrices A_i and D. The covariance was estimated in a straightforward manner from the regression residuals. We considered both a diagonal and a fully dense covariance matrix. Our VARX model set-up is a particular case of a Gaussian process that uses generalized linear models (GLIMs) to describe its mean matrix, where the covariates of the GLIM represent spatio-temporal process variables. In this study we chose to formulate our approach in the more specific terms of VARX processes.

In order to test the performance of the proposed stochastic parameterizations, we compared the reduced stochastic model simulations with two different configurations of the deterministic L96 reference model. First, the unimodal configuration, where "unimodal" refers to the overall shape of the probability distribution of $\boldsymbol{x}$, the variable of interest. This is a "standard" configuration of the L96 model that has also been used in previous studies. Second, to provide a very challenging test case and push our methodology to its limits, we also considered a *trimodal* configuration of the L96 model. This is a non-standard configuration for the L96 model that exhibits three clear distinct peaks in the distribution of $\boldsymbol{x}$. The trimodal configuration tests the robustness of the proposed VARX process. As mentioned, the performance was assessed using a number of statistical criteria of the resolved model variable $\boldsymbol{x}$: the probability density function (PDF), the autocorrelations (ACFs), cross-correlations (CCFs), and the mean and variance of the wavenumber vector of $\boldsymbol{x}$.

In our results we compared different stochastic parameterizations for the reduced model Equations (4.12)–(4.13). First, we tested both conditioning on the state vector $\boldsymbol{x}$ and self-conditioning on the stochastic process $\boldsymbol{b}$ in parameterizations (*WND*) and (*Multi* AR(1)), respectively. Here, self-conditioning refers to conditioning on the process itself at previous times. The results show that these regressors serve different roles in the conditioning. The state-dependent regressor $\boldsymbol{x}$ served effectively as predictor variable for the unresolved process, whereas the self-conditioning on $\boldsymbol{b}$ was instrumental in preserving temporal (de)correlations in the VARX. Each of these regressors by themselves was unsuccessful in giving satisfying results. However, combining the state-dependent and self-conditioning regressors proved very successful. The statistical criteria of the reference unimodal L96 model were reproduced very accurately using just a diagonal covariance matrix. For the trimodal test case, VARX with a diagonal covariance matrix gave qualitatively correct but not very accurate results. We showed quantitative improvement of results using a fully-dense covariance structure.

Finally, we also compared the performance of the VARX models to the NARMAX models proposed in Chorin and Lu [25]. As shown in Chorin and Lu

[25], the NARMAX models perform very accurately for the unimodal L96 test case. However, we showed that for the trimodal test case, the NARMAX models were not able to reproduce the trimodal distribution of the resolved variable x_k accurately, nor its wave statistics.

The NARMAX models provide a parameterization for a single grid point, so they are applied independently to all grid points. By contrast, the VARX model can give a parameterization for the entire grid at once (as the VARX process is vector-valued), making it easier to include spatial correlations and spatial inhomogeneity. These spatial characteristics can be important for applications such as ocean modeling.

In future work we plan to develop the VARX stochastic parameterization methodology further. An important issue to consider is how to compute efficiently with a covariance structure that allows for spatial correlations without having to construct a fully dense matrix. This should involve a number of parameters that is at most linear in the number of spatial degrees of freedom, e.g. grid points. We intend to apply these methods in tests with a complex ocean model.

Chapter 5

A comparison of methods for estimating sparse covariance and precision matrix roots

We study the problem of estimating the covariance and precision matrices, as well as their roots, from data of a high-dimensional spatially correlated process. To facilitate efficient sampling of a stochastic process that has approximately the same covariance as the data, it is highly beneficial if the root of either the covariance or the precision is a sparse matrix. Moreover, the dimension of the process can be so high that loading the fully dense sample covariance matrix in its entirety into memory is not practically feasible, not even as part of an algorithm to estimate a sparse approximation.

A variety of methods exists for estimating sparse covariance and precision matrices and their roots, many of which are variations of the graphical lasso. In this study we compare several state-of-the-art estimation methods, using data generated by a prototype SPDE model. We assess the errors and the sparsity of the estimates obtained by the different methods, and we compare their eigenspectrum against that of the (non-sparse) sample covariance computed directly from the data. We find that the Convex Sparse Cholesky Selection (CSCS) algorithm proposed by Khare et al. [88] gives the best performance. The CSCS method yields efficient and accurate solutions and gives the user control over the balance between accuracy and sparsity by means of its regularization parameter.

1 Introduction

In this study we consider the problem how to sample from a high-dimensional correlated Gaussian distribution without placing a fully dense covariance matrix (or its square root) in memory. It is motivated by our recent work on stochastic parameterization with a vector autoregressive model with exogenous variables (VARX process) in Verheul and Crommelin [144]. That work was aimed towards application in the domain of numerical ocean modeling where the system state can be of very high dimension. Therefore an important concern is how to handle cases where the dimension of the VARX process is high. As an example, the (idealized) ocean model used in Verheul et al. [145] was formulated on a horizontal numerical grid of size 401×401, so that if a VARX parameterization were used for that model over its entire spatial domain, the VARX process would have dimension 401^2.

To make simulations computationally efficient it is highly beneficial if the coefficient matrices that determine the VARX process are sparse. Let us write the VARX process of order p as $\boldsymbol{y}^n = \boldsymbol{a}_0 + \boldsymbol{A}_1\boldsymbol{y}^{n-1} + \cdots + \boldsymbol{A}_p\boldsymbol{y}^{n-p} + \boldsymbol{D}\boldsymbol{x}^n + \boldsymbol{S}\boldsymbol{\xi}^n$, where $\boldsymbol{\xi}^n$ is the noise vector and $\boldsymbol{x}^n$ the vector of exogenous variables. $\boldsymbol{A}_i$ $(i = 1, ..., p)$, $\boldsymbol{D}$ and $\boldsymbol{S}$ are coefficient matrices. If $\dim(\boldsymbol{y}^n) = K$, these matrices are of size $K \times K$ so it is clear that sparsity becomes important if e.g. $K = \mathcal{O}(10^5)$ as in the example mentioned earlier.

In Verheul and Crommelin [144] we focused on sparse matrices for the drift components of a VARX process, i.e. on the matrices $\boldsymbol{A}_i$ and $\boldsymbol{D}$. In the current study we turn to the noise component of the VARX process with its matrix $\boldsymbol{S}$ and the associated covariance matrix $\boldsymbol{S}\boldsymbol{S}^T$, and to the question how to estimate these matrices. We explore and review state-of-the-art methods for estimating a sparse covariance matrix, its inverse, and their matrix roots. To our knowledge, no previous studies exist that explore such methods to use them for stochastic parameterization in ocean modeling, the application of primary interest to us.

In the remainder of this introduction we discuss the following. In Section 1.1 we summarize several assumptions and limitations that arise from the ocean modeling application. These are important factors for the selection and assessment of the methods discussed in this study. We also list some main notation used throughout the paper. In Section 1.2 we briefly introduce some basic concepts and results from the field of sparse covariance and inverse covariance matrix estimation. In Section 1.3 we introduce the graphical lasso, the method that forms the basis for most of the methodologies considered in this paper. Section 1.4 presents an overview of the material discussed in the rest of the paper.

1.1 Assumptions, limitations, and notation

The study we present here is motivated by ocean modelling and stochastic parameterization. This intended application entails some assumptions and limitations that we summarize here as they are relevant to the selection and evaluation of the estimation methods in later sections. For more extensive discussions of the relevant background of ocean modeling and of stochastic parameterizations, see Verheul et al. [145], and Verheul and Crommelin [143] and Verheul and Crommelin [144], respectively.

1. **Stochastic modelling**: In ocean modeling, the so-called mesoscale turbulent eddies in the ocean need to be parameterized because it is computationally infeasible to resolve them explicitly. Deterministic parameterizations are insufficient because they can, at best, represent the ensemble-mean of the effect of the unresolved scales (denoted with y). Stochastic parameterizations are required to capture the variability around this mean and its impact on the resolved scales (denoted x).

2. **Unresolved scale distribution**: We assume that the dynamical effects of the unresolved scales on the resolved scales can be represented with a high-dimensional VARX process with correlated noise (corresponding to spatial correlations).

3. **Spatial decorrelation**: We assume there exists a finite spatial decorrelation scale $\rho > 0$ such that the model variables y_i and y_j are (approximately) pairwise *conditionally independent* if $d(i, j) > \rho$ (where d is a *distance function*). That is, the process y evaluated in spatial grid indices i and j for which $d(i, j) > \rho$ is pairwise conditionally independent, meaning that y_j is independent from y_i conditional on all other y-components: $y_j \perp y_i \,|\, y_{\setminus\{i,j\}}$. This is a reasonable assumption for the mesoscale eddy forcing in ocean modeling.

4. **Spatial inhomogeneity**: The correlation between model variables y_i and y_j is determined by both the distance between i and j and the absolute location of i and j on the spatial grid.

5. **Data availability**: We assume the availability of sample data of length N both for the resolved-scale ocean state (x) and for the effect of the unresolved scales (y), e.g. in the form of satellite measurements or a highly accurate simulation over short spatiotemporal intervals.

6. **High-dimensionality**: In ocean modelling, the number of spatial grid points K can be on the order of 10^5 or more. For the methodologies we investigate it is important that they can be scaled to such high dimensions. In particular, the problem dimension can be higher than the number of available data points, i.e. $K > N$.

7. **Memory limitations**: For high-dimensional problems it is unfeasible to load the covariance matrix (or its inverse) in computer memory if it is dense. This makes methodologies that need to construct either of the two matrices in full unsuitable.

8. **Gradual fraction of explained variance**: The mesoscale turbulent eddy forcing cannot be represented well with a small number of principal components [123]. Thus, if $\boldsymbol{\lambda}$ are the (ordered) eigenvalues of the covariance matrix $\boldsymbol{\Sigma}$ (i.e. $\lambda_k \boldsymbol{v}_k = \boldsymbol{\Sigma} \boldsymbol{v}_k$), the *fraction of explained variance*,

$$\sum_{k=1}^{i} \lambda_k \,/\, \sum_{k=1}^{K} \lambda_k, \tag{5.1}$$

increases only gradually with i. There is no small number i such that this fraction is close to 1.

In the above, we introduced the general grid index i. Of course, a single index i only coincides precisely with corresponding grid coordinates in the case of a one-dimensional grid. In case of a D-dimensional grid with multi-index $\boldsymbol{i} := (i_1, \ldots, i_D)$, we use i as a single-index notation (e.g. by defining $i = i_1 + K_1(i_2 - 1)$ if $D = 2$ and i_1 runs from 1 to K_1). The distance function $d(i, j)$ must be defined accordingly.

Finally, in Table 5.1 we list notation used throughout this study.

Table 5.1: Notation used throughout this study

Notation	Format	Explanation
$p \sim P$	Real-valued scalar	Random variable p is sampled from distribution P
x_i^n	Real-valued scalar	Deterministic model variable at time $n\Delta t$ at grid point i
$\widetilde{y}_i^n$	Real-valued scalar	Stochastic model variable at time $n\Delta t$ at grid point i
$\boldsymbol{y}_i$	Vector of length N	Model variables at grid point i, i.e. $\boldsymbol{y}_i = (y_i^1, \ldots, y_i^N)$
$\boldsymbol{y}^n$	Vector of length K	Model variables at time $n\Delta t$, i.e. $\boldsymbol{y}^n = (y_1^n, \ldots, y_K^n)$
$\boldsymbol{Y}$	Matrix of size (N, K)	Model variable y across all temporally and spatially sampled points
Σ_{ij}	Real-valued scalar	True covariance between stochastic variables y_i and y_j
$\boldsymbol{\Sigma}$	Matrix of size $K \times K$	True covariance matrix of stochastic field $\boldsymbol{y}$
$\overline{\boldsymbol{\Sigma}}$	Matrix of size $K \times K$	Sample covariance matrix based on sample data $\boldsymbol{Y}$
$\widehat{\boldsymbol{\Sigma}}$	Matrix of size $K \times K$	Approximated covariance matrix based on sample data $\boldsymbol{Y}$
$\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Vector of length K	Sample from K-variate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$

1.2 Sparse covariance and precision matrix estimation

In Verheul and Crommelin [144] we developed a stochastic parameterization method that fits *vectorized autoregressive processes with exogenous parameters* (VARXs) to given model data $(\boldsymbol{X}, \boldsymbol{Y})$ resulting from a deterministic reference model. A VARX of order p, denoted by VARX(p), with exogenous parameter $\widetilde{\boldsymbol{x}}$ is given by (see e.g, Lütkepohl [100]):

$$\widetilde{\boldsymbol{y}}^n = \boldsymbol{a}_0 + \boldsymbol{A}_1\widetilde{\boldsymbol{y}}^{n-1} + \cdots + \boldsymbol{A}_p\widetilde{\boldsymbol{y}}^{n-p} + \boldsymbol{D}\widetilde{\boldsymbol{x}}^n + \boldsymbol{S}\boldsymbol{\xi}^n. \tag{5.2}$$

In the VARX(p) definition above, $\boldsymbol{a}_0$ is the linear offset, $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_p$ are the *endogenous* drift matrices, $\boldsymbol{D}$ is the *exogenous* drift matrix, $\boldsymbol{\xi}^n \sim \mathcal{N}(0, I)$ is a vector of independent normally distributed random variables, and $\boldsymbol{S}$ is the root of the covariance matrix, i.e. $\boldsymbol{\Sigma} = \boldsymbol{S}\boldsymbol{S}^T$. Numerical implementation of such a process is straightforward (e.g. Pavliotis [114]). We can infer accurate sparse approximations of the VARX drift matrices and linear offsets from the second-order statistics estimated from the available sample data $(\boldsymbol{X}, \boldsymbol{Y})$ [144].

To obtain a sparse VARX representation of $\boldsymbol{y}$, we also need a sparse representation (or approximation) of the covariance root $\boldsymbol{S}$ in (5.2), within the limitations put forward in Section 1.1. The covariance matrix $\boldsymbol{\Sigma}$ is sparse if many of the random variables are uncorrelated. That is, let $\boldsymbol{z} \sim \mathcal{N}(\mu, \boldsymbol{\Sigma})$ be a vector of Gaussian random variables, then we have:

Property 1. $\boldsymbol{\Sigma}_{ij} = 0$ *if and only if* z_i *and* z_j *are mutually independent.*

In an ocean model, variables at different grid points are typically correlated, albeit only weakly over longer distances [123]. In such cases, $\boldsymbol{\Sigma}$ will not be sparse. As an alternative, we can use the precision matrix $\boldsymbol{\Theta}$ and its root $\boldsymbol{L}$. These are defined by $\boldsymbol{\Theta} := \boldsymbol{\Sigma}^{-1}$ and $\boldsymbol{L}^T\boldsymbol{L} = \boldsymbol{\Theta}$, so that we have $\boldsymbol{\Theta}^{-1} = \boldsymbol{L}^{-1}(\boldsymbol{L}^{-1})^T$. The distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Theta}^{-1}) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{L}^{-1}(\boldsymbol{L}^{-1})^T)$ is equivalent to $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. Therefore, the VARX in (5.2) is equivalent to:

$$\widetilde{\boldsymbol{y}}^n = \boldsymbol{a}_0 + \boldsymbol{A}_1\widetilde{\boldsymbol{y}}^{n-1} + \cdots + \boldsymbol{A}_p\widetilde{\boldsymbol{y}}^{n-p} + \boldsymbol{D}\boldsymbol{x}^n + \boldsymbol{L}^{-1}\boldsymbol{\xi}^n \tag{5.3}$$

where the product $\boldsymbol{L}^{-1}\,\boldsymbol{\xi}^n =: \boldsymbol{b}^n$ can be computed by solving the system $\boldsymbol{L}\,\boldsymbol{b}^n = \boldsymbol{\xi}^n$ with forward substitution.

It is convenient to work with the equivalent VARX formulation (5.3) because in many applications the precision matrix is sparse, e.g. speech recognition [124, 125], genetics [24, 75, 154], and macroeconomics [37, 93].

The assumption we make here is that one can more easily get a sparse yet accurate approximation of the precision matrix than of the covariance matrix. This is due to the inherent relationship between the precision matrix and *conditional dependence* in the random variable: each zero element in the precision matrix corresponds to conditional independence in the random variable. That is:

Property 2. $\boldsymbol{\Theta}_{ij} = 0$ *if and only if* z_i *and* z_j *are conditionally independent given all other elements in* $\boldsymbol{z}$.

For example, the Matérn covariance function is shown to have sparse precision matrix when $\nu + d/2$ is integer [96, 150, 151]. This is illustrated in Figure 5.1, where the Matérn covariance matrix for a one-dimensional grid with 100 grid points is fully dense, but the corresponding precision matrix is highly sparse.

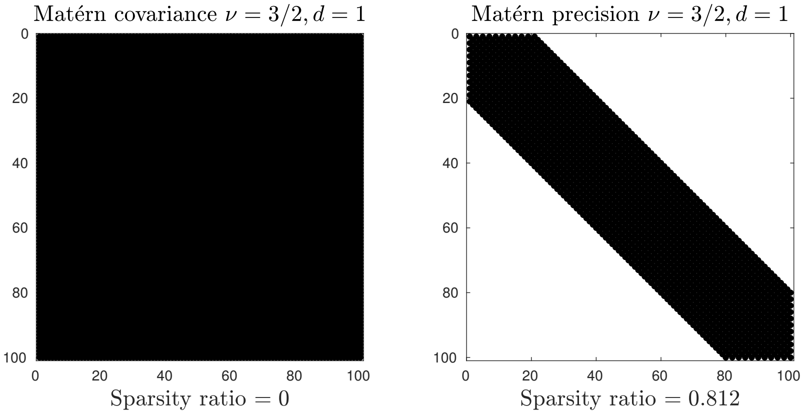


Figure 5.1: Sparsity patterns for the covariance (left) and precision (right) matrices generated by a Matérn covariance function.

Due to memory limitations (see Limitation 7), we want to avoid explicit calculation of the full sample covariance matrix. Consequently, calculating the matrix root or the inverse directly from the full sample covariance matrix is also unwanted. Therefore, an approximation method to find $\widehat{\boldsymbol{\Sigma}}$ or $\widehat{\boldsymbol{\Theta}}$, or directly $\widehat{\boldsymbol{S}}$ or $\widehat{\boldsymbol{L}}$, should not require to put either $\overline{\boldsymbol{\Sigma}}$ or $\overline{\boldsymbol{\Theta}}$ fully in memory. So here "the approximated covariance" is a different approximation than the sample covariance. Instead, it simultaneously attempts to introduce sparsity in its covariance approximation.

Thus, our aim is to arrive at a sparse yet accurate approximation of either the covariance root or the precision root matrix, estimated from data. We can distinguish three groups among the methods we consider to get these sparse approximations ($\widehat{\boldsymbol{S}}$ or $\widehat{\boldsymbol{L}}$). In the first group, a sparse approximation of the covariance is estimated (e.g. using a tapering method), followed by Cholesky decomposition to get the approximate covariance root $\widehat{\boldsymbol{S}}$. Thus, $\widehat{\boldsymbol{S}}$ is estimated indirectly. In the second group it is $\widehat{\boldsymbol{L}}$ that is estimated indirectly: first one estimates a sparse approximation of the precision matrix (e.g. using a variant of the graphical lasso, see Section 1.3), and then one computes $\widehat{\boldsymbol{L}}$ by Cholesky. We note that only under specific conditions does sparsity in a matrix lead to sparsity in its matrix root (a well-known case being a band-diagonal matrix: the Cholesky decomposition of a banded matrix with bandwidth b is itself a banded matrix with bandwidth $b/2$) [64]. Finally, in the third group, $\widehat{\boldsymbol{L}}$ is estimated directly: we consider a state-of-the-art method that uses many parallelized approximations of the columns of $\boldsymbol{L}$ based on likelihood penalization. More on this in Section 4.1.

In what follows, we make use of two different decompositions of the precision matrix: the Cholesky decomposition $\boldsymbol{\Theta} = \boldsymbol{L}^T\boldsymbol{L}$ and the modified Cholesky decomposition $\boldsymbol{\Theta} = \boldsymbol{M}^T\boldsymbol{D}^{-1}\boldsymbol{M}$. Here $\boldsymbol{L}$ is a lower triangular matrix, $\boldsymbol{M}$ is a lower triangular matrix with unit diagonal, and $\boldsymbol{D}$ is a diagonal matrix with positive diagonal. The values in $\boldsymbol{\Theta}$ and $\boldsymbol{M}$ have related interpretations: the entries in the ith row of $\boldsymbol{\Theta}$ can be interpreted as regression coefficients of the ith variable on *all other* variables, whereas the non-zero entries in $\boldsymbol{M}$ are the regression coefficients of its corresponding variable on the *preceding* variables. The entries in $\boldsymbol{D}$ are the residual variance of the regression fit. The three matrices $\boldsymbol{D}, \boldsymbol{L}, \boldsymbol{M}$ are connected by the straightforward relationship $L_{ij} = M_{ij}/\sqrt{D_{jj}}$ for every $i \leq j$. This interpretation assumes some ordering of the variables in $\boldsymbol{\Theta}$ because the Cholesky decomposition is not invariant to variable reordering.

One wonders how sparsity patterns in $\boldsymbol{\Theta}$ and $\boldsymbol{M}$ are related. While in general the sparsity patterns are different between the two matrices, there are two special cases. First, as mentioned above, if $\boldsymbol{\Theta}$ is a banded matrix with bandwidth b, then $\boldsymbol{M}$ is itself a banded matrix with bandwidth $b/2$. Second, we know that the sparsity pattern of $\boldsymbol{\Theta}$ and $\boldsymbol{M}$ are equal if and only if the corresponding graph of $\boldsymbol{M}$ is chordal and the vertices are ordered based on a perfect vertex elimination scheme [112]. However, neither of these cases are expected for ocean flows: finding a small bandwidth is unlikely for grids of dimension greater than 1, and imposing a chordal structure on $\boldsymbol{M}$ is overly restrictive.

1.3 Graphical lasso

Consider the problem of estimating the $K \times K$ precision matrix $\boldsymbol{\Theta}$ of a K-variate probability distribution from N independently and identically distributed samples $\boldsymbol{y}^1, \ldots, \boldsymbol{y}^N$. Particularly, we consider the estimation of a sparse precision matrix in a memory-efficient setup so that it is suitable for applications with large K, potentially much larger than the number of available samples, i.e. $K \gg N$. A natural way [46, 60] to estimate the precision matrix $\boldsymbol{\Theta}$ given sample data $\boldsymbol{Y}$ for a Gaussian model is by maximizing the log-likelihood:

$$\widehat{\boldsymbol{\Theta}}^{\mathrm{MLE}} = \underset{\widehat{\boldsymbol{\Theta}}}{\arg\max} \quad \log\det(\widehat{\boldsymbol{\Theta}}) - \operatorname{tr}(\widehat{\boldsymbol{\Theta}}\overline{\boldsymbol{\Sigma}}), \tag{5.4}$$

where $\overline{\boldsymbol{\Sigma}} = \boldsymbol{Y}^T\boldsymbol{Y}/N$ is the sample covariance matrix. $\widehat{\boldsymbol{\Theta}}^{\mathrm{MLE}}$ is the maximum likelihood estimator of $\boldsymbol{\Theta}$. However, matrix elements that are zero in $\boldsymbol{\Theta}$ will in general not be exactly zero in $\widehat{\boldsymbol{\Theta}}^{\mathrm{MLE}}$. Moreover, if $K > N$, then the sample covariance matrix may not be invertible [7, 33]. To ensure that the estimated covariance is invertible, a common approach (e.g. Banerjee et al. [7], Dempster [35], Friedman et al. [57], Rothman et al. [120], Yuan and Lin [159]) is to add ℓ_1-regularization that also explicitly imposes sparsity on $\widehat{\boldsymbol{\Theta}}$:

$$\widehat{\boldsymbol{\Theta}}^{\mathrm{GL}} = \underset{\widehat{\boldsymbol{\Theta}} \succ 0}{\arg\max} \quad \log\det(\widehat{\boldsymbol{\Theta}}) - \operatorname{tr}(\widehat{\boldsymbol{\Theta}}\overline{\boldsymbol{\Sigma}}) - \lambda\left\|\widehat{\boldsymbol{\Theta}}\right\|_1. \tag{5.5}$$

Here, the search for estimators considers only positive definite matrices $\widehat{\boldsymbol{\Theta}} \succ 0$. The first term of (5.5) works toward positive definiteness, because $\det \boldsymbol{\Theta} > 0$ is a necessary (but not sufficient) condition for $\boldsymbol{\Theta}$ to be positive definite. The solution $\widehat{\boldsymbol{\Theta}}^{\mathrm{GL}}$ is positive definite for all $\lambda > 0$, even if $\overline{\boldsymbol{\Sigma}}$ is singular. The second term penalizes $\widehat{\boldsymbol{\Theta}}$ moving away from $\overline{\boldsymbol{\Sigma}}^{-1}$. Finally, the ℓ_1-regularization term encourages sparsity in $\widehat{\boldsymbol{\Theta}}$. One of the main challenges of solving this optimization problem is finding the appropriate regularization strength λ. This parameter controls the trade-off between penalizing non-sparsity and minimizing the cost-function. But this term guarantees that if λ is sufficiently large, then the precision matrix estimator $\widehat{\boldsymbol{\Theta}}^{\mathrm{GL}}$ is sparse due to the lasso-type penalty [141].

This approach is more commonly known as the *graphical lasso* (GL) [7, 57, 159]. The graphical lasso (5.5) is a convex problem for which there are solution algorithms available that solve the problem in polynomial time with interior point methods [22] or even faster with coordinate descent algorithms [34, 57]. Moreover, this estimator is known to give certain statistical guarantees, such as consistency between Frobenius and spectral norms under the condition of bounded minimum and maximum eigenvalues. The most important beneficial statistical properties for the study here are listed below:

- The ℓ_1 based optimizer recovers the sparsity pattern of the precision matrix $\boldsymbol{\Theta}$ under loose restrictions on the number of edges in the adjacency matrix of $\boldsymbol{\Theta}$ and the number of observations N [120].
- The ℓ_1 based optimizer is consistent between the Frobenius and spectral norms [120].
- The ℓ_1 based optimizer is suitable for data $\boldsymbol{Y}$ sampled from multivariate distributions that can be either Gaussian or non-Gaussian. This is because the graphical lasso corresponds to maximization of an ℓ_1-penalized log-determinant Bregman divergence [118].

From the field of sparse inverse covariance estimation (or, equivalently, sparse precision matrix estimation) we know there are many algorithms that solve (5.5). Some well-known methods include the block-coordinate descent method COVSEL [8] and the greedy coordinate-descent method aimed at large scale parallellization SINCO [127]. State-of-the-art solvers for (5.5) include the graphical lasso (GLASSO [57]) and the quadratic approximate inverse covariance (QUIC [79]) method. Each of these methods solve (5.5) through (block)-coordinate descent. GLASSO has been widely adopted by statisticians through the `GLASSO` R-package. GLASSO, and other first-order methods like it [34, 126, 128], can be implemented straightforwardly and perform fast computations because they mainly use gradient information at every step. However, these first-order methods are limited to linear convergence at best [17]. To achieve superlinear convergence, recent studies have looked at second-order methods [94, 129] that use, at least in part, the Hessian of the cost function [78]. QUIC is a popular second-order method that solves the ℓ_1-objective directly. QUIC has been shown to achieve superlinear convergence [79].

However, the traditional graphical lasso is commonly considered to be intractable for large datasets because of memory limitations [105, 161] (fitting the $K \times K$ sample covariance matrix in memory can already be problematic) and difficulties with convergence of the established solvers [78]. In recent years several improvements have been proposed over both GLASSO (variants with better convergence properties (P-GLASSO and DP-GLASSO [106])) and QUIC (to support "big data" (BiqQUIC [80]) or to take advantage of sparsity in its components in a massively parallellizable way (SQUIC [19])).

One of the methods we investigate in the current study is the max-det matrix completion (MDMC) method as proposed by Zhang et al. [161]. This methodology is able to inform a "restricted graphical lasso" (RGL) formulation with prior information on the precision sparsity pattern.

1.4 Overview of work

The remainder of this study is structured as follows. In Section 2 we discuss the prototype stochastic partial differential equation (SPDE) model used to assess the various methodologies. We perform simulations with this model to generate data sets with varying statistical properties. These data sets are used for testing and comparing the different estimation methods. In Section 3 we present multiple state-of-the-art methodologies from the literature for estimating sparse covariance, precision, or precision root matrices. These methodologies are then tested for accuracy and sparsity in Section 4, using the data sets mentioned above. Finally, in Section 5 we present the final takeaways of this comparison study and what next steps could look like to bring the results into practice.

2 Prototype model to generate data

To evaluate the capability of the methodologies that will be presented in Section 3 we generate datasets with a simple prototype model. The model is based on a SPDE driven by spatially correlated white noise. We consider this SPDE both on a 1D and on a 2D spatial domain, with periodic boundary conditions. The 1D and 2D versions are

$$y_t(u,t) = b\,y_{uu}(u,t) - a\,y(u,t) + w'(u,t), \tag{5.6}$$

$$y_t(u,v,t) = b\,(y_{uu}(u,v,t) + y_{vv}(u,v,t)) - a\,y(u,v,t) + w'(u,v,t), \tag{5.7}$$

where $y_t := \partial t\, y$ is the time derivative of y, u and v are the spatial coordinates and y_{uu} and y_{vv} are the second spatial derivatives of y. The linear term ay represents damping. The force w' that drives the SPDE is Gaussian noise that is white in time and correlated in space.

The SPDEs (5.6) and (5.7) are discretized in space on a uniform grid with K gridpoints and spacing Δu, using central finite differences for the diffusion terms (y_{uu} and y_{vv}), and integrated in time using an Euler scheme. The resulting system can be written schematically as $y^{n+1} = M\,y^n + \boldsymbol{w}^n$, with the matrix M determined by the discretization and n the time index. We use two types of noise:

$$\boldsymbol{w}^n = \boldsymbol{I}\boldsymbol{\xi}^n, \tag{5.8}$$

$$\boldsymbol{w}^n = \boldsymbol{S}\boldsymbol{\xi}^n, \tag{5.9}$$

where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{\xi}^n \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$, and $\boldsymbol{S}$ is the matrix root of covariance matrix $\boldsymbol{\Sigma}$ defined through the distance function $d(i,j)$:

$$\boldsymbol{\Sigma}_{i,j} := \exp\left(-0.5\, d(i,j)^2\right). \tag{5.10}$$

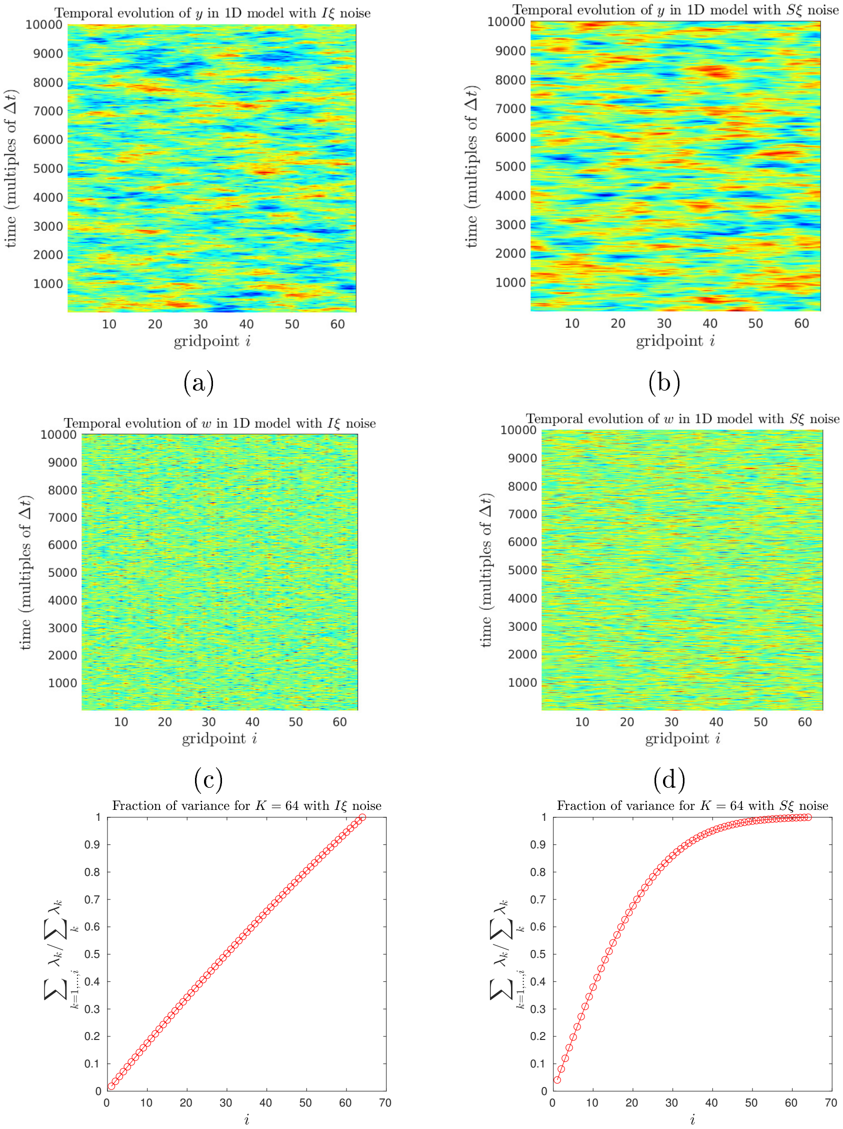
The main purpose of this SPDE-based model is that is enables us to easily generate datasets with different spatial correlation properties and different principal component eigenspectra. Throughout this study we consider four separate use cases of the prototype model: either the 1D (5.6) or 2D SPDE (5.7) with either of the noise realizations defined in (5.8) and (5.9). Thus, a particular use case is identified by concatenating these two choices, e.g. model (5.6) with noise (5.8) refers to the 1D SPDE model with spatially uncorrelated white noise. These models allow for nontrivial, but simple, ways to generate spatially correlated noise that distort the temporal evolution of y. Table 5.2 shows the parameter settings used throughout this study that are chosen to both resemble interesting dynamics and to satisfy the model constraints $a\,\Delta t < 1$ and $2\,b\,\Delta t \Delta u^{-2} < 1$.

Table 5.2: Model configurations for prototype model in Section 2.

Parameter	Value (5.6) $K = 16$	Value (5.6) $K = 64$	Value (5.7) $K = 256$
a	1	1	1
b	0.1	0.1	0.1
dim	1	1	2
$d(i,j)$	$\lvert i-j \rvert$	$\lvert i-j \rvert$	$\lVert i-j \rVert_2$
Δt	0.005	0.005	0.005
N	10^4	10^4	10^4
Δu	0.1	0.1	0.1
K	16	64	(16×16)

Figure 5.2 illustrates the dynamical behavior of the 1D SPDE model (5.6) for the model parameter settings $K = 64$ in Table 5.2. Particularly, one sees stronger spatial correlation in the model variable $\boldsymbol{y}$ when using the covariance root S instead of I in the noise (difference between Figures 5.2a and 5.2b). Additionally, Figures 5.2e and 5.2f show that, while more steep for the model with spatially correlated noise, both models have gradual increase in fraction of explained variance across the eigenspectrum, in line with Assumption 8. This enables a clean test bed for the methodologies presented in Section 3.

We use three distinct configurations related to dimensionality of the prototype model in Section 2: two configurations of the 1D model (5.6) with different spatial resolutions and one configuration of the 2D model (5.7). These model configurations can be found in Table 5.2.



(a) (b) (c) (d) (e) (f)

Figure 5.2: Examples of the dynamics of the 1D SPDE model (5.6) with parameter configurations from Table 5.2. Figures (a), (c), and (e) illustrate spatio-temporal dynamics of (5.6) with spatially uncorrelated noise (5.8). Analogously, Figures (b), (d), and (f) for spatially correlated noise (5.9). Figures (a)-(b) illustrate the spatio-temporal evolution of model variable $\boldsymbol{y}$, Figures (c)-(d) illustrate the spatio-temporal evolution of the white noise $\boldsymbol{w}$. Figures (e)-(f) show the variance fraction explained by the first i eigenmodes.

3 Sparse matrix estimation methodologies

3.1 Sparse covariance estimation

In this section we consider how to estimate a sparse covariance matrix from data. The covariance root can be obtained in a next step by Cholesky decomposition (as also summarized in section 1.2). Many approaches exist to estimate a sparse covariance matrix based on a sample covariance matrix $\overline{\boldsymbol{\Sigma}}$. The latter can be constructed directly from the data as $\overline{\boldsymbol{\Sigma}} = \boldsymbol{Y}^T\boldsymbol{Y}/N$. Here we present an overview of the thresholding, banding, and tapering methods that do not require to put $\overline{\boldsymbol{\Sigma}}$ fully in memory, in line with Limitation 7 in section 1.1.

3.1.1 Thresholding methods

First, the thresholding methods aim to find a covariance matrix estimate $\widehat{\boldsymbol{\Sigma}}_\omega$ that truncates small elements in the sample covariance to zero:

$$(\widehat{\boldsymbol{\Sigma}}_\omega)_{i,j} = \begin{cases} \overline{\boldsymbol{\Sigma}}_{\boldsymbol{i},\boldsymbol{j}}, & \text{if } \boldsymbol{i} = \boldsymbol{j} \\ \overline{\boldsymbol{\Sigma}}_{\boldsymbol{i},\boldsymbol{j}} \mathbf{1}\{|\overline{\boldsymbol{\Sigma}}_{\boldsymbol{i},\boldsymbol{j}}| \geq \omega\}, & \text{if } \boldsymbol{i} \neq \boldsymbol{j} \end{cases}, \tag{5.11}$$

where $\mathbf{1}$ denotes the indicator function and ω denotes some scalar thresholding value. Note that the thresholding covariance preserves symmetry and can be easily obtained without storing the sample covariance fully in memory. In fact, the thresholding operator can be calculated with a limited memory usage of $O(K)$ because $\widehat{\boldsymbol{\Sigma}}_\omega$ can be constructed element-wise and can fully parallelized.

However, the thresholding operator does not guarantee positive definiteness of $\widehat{\boldsymbol{\Sigma}}_\omega$. Positive definiteness is guaranteed if $\left\|\widehat{\boldsymbol{\Sigma}}_\omega - \overline{\boldsymbol{\Sigma}}\right\| \leq \epsilon$ and $\lambda_{\text{min}}(\overline{\boldsymbol{\Sigma}}) > \epsilon$, where λ_{min} indicates the smallest eigenvalue and $\epsilon > 0$ [18]. This condition restricts the degree to which $\overline{\boldsymbol{\Sigma}}$ can be truncated. Unfortunately, this condition is quite restrictive.

3.1.2 Banding methods

Second, banding methods truncate elements $\boldsymbol{\Sigma}_{i,j}$ to 0 based on the distance between grid points i and j:

$$(\widehat{\boldsymbol{\Sigma}}_k)_{i,j} := \overline{\boldsymbol{\Sigma}}_{i,j}\mathbf{1}(d(i,j) < k), \tag{5.12}$$

where $d(i, j)$ is the distance function. This approximation method is ideal when the indices of the spatial grid are ordered in such a way that $d(i, j) > k \Rightarrow \mathbf{\Sigma}_{i,j} \approx 0$. The banding operator preserves symmetry by definition of the distance function. The banded matrix $\widehat{\mathbf{\Sigma}}_k$ can also be obtained in a memory-efficient way because the sparsity pattern of $\widehat{\mathbf{\Sigma}}_k$ can be identified independently from the sample covariance $\overline{\mathbf{\Sigma}}$. However, the banding operator does not guarantee that $\widehat{\mathbf{\Sigma}}_k$ is positive definite.

3.1.3 Tapering methods

Third, tapering methods are an extension of banding methods that additionally preserves positive definiteness of the approximated covariance [58]. A tapering method 'regularizes' the covariance matrix by calculating the (element-wise) Hadamard product (denoted by $\odot$):

$$\widehat{\mathbf{\Sigma}}_{\boldsymbol{T}} = \overline{\mathbf{\Sigma}} \odot \boldsymbol{T}, \tag{5.13}$$

for tapering matrix $\boldsymbol{T}$ that is chosen to be positive definite and compactly supported:

$$\boldsymbol{T}_{i,j} := c(d(i, j)), \tag{5.14}$$

where $c : \mathbb{R}^+ \to \mathbb{R}^+$ is a compactly supported covariance function that decreases monotonically to 0 and has the property $c(0) = 1$, and $d(i, j)$ is the distance function.

The tapering operation (5.13) preserves positive definiteness because the Hadamard product between positive definite matrices is itself positive definite [76, 130]. Additionally, the support of covariance function c is typically chosen in such a way that $\boldsymbol{T}_{i,j} = 0$ (and thus $(\widehat{\mathbf{\Sigma}}_{\boldsymbol{T}})_{i,j} = 0$) when $d(i, j) \geq k$ for some range parameter k. Therefore, the tapering approach promotes sparsity in the covariance approximation.

While the tapering definition (5.13) might suggest that one needs to construct the sample covariance $\overline{\mathbf{\Sigma}}$ fully to calculate the tapered covariance, let us remark that the Hadamard product operates element-wise and that the choice (5.14) allows one to identify the sparsity pattern of $\boldsymbol{T}$ (and thus $\widehat{\mathbf{\Sigma}}_{\boldsymbol{T}}$) independently from that of $\overline{\mathbf{\Sigma}}$.

Besides introducing sparsity due to so-called localization, tapering has also been shown to regulate the variance of the sample covariances [73, 77]. It can also be shown that for tapering matrices with appropriate polynomial covariance function c the mean squared error (MSE) of the tapered covariance predictions converges asymptotically to the minimal error as available data approaches infinity [59].

Different tapering methods are identified by their choice of covariance function c. Here, we consider the following covariance functions (with support $u := \frac{d(i,j)}{d_{\max}} \leq 1$ with $d_{\max} > 0$) as examples, where $c(u) = 0$ outside the support:

$$\text{Epanechnikov covariance [45]:} \quad c(u) := 1 - u^2 \tag{5.15}$$

$$\text{Wendland covariances [149]:} \quad c(u) = \psi_{1,0} = 1 - u \tag{5.16}$$

$$c(u) = \psi_{2,1} = (1-u)^3(3u+1) \tag{5.17}$$

$$c(u) = \psi_{3,2} = (1-u)^5(8u^2+5u+1) \tag{5.18}$$

3.2 Sparse precision estimation

Here we present an overview of sparse precision matrix estimation methods we consider. Similar to the covariance estimation methods, these methods have to be followed by a Cholesky decomposition step to obtain the matrix root $\boldsymbol{L}$ that can be used during simulation of the VARX process, as discussed in section 1.2. They can be roughly subdivided in two categories: penalized likelihood methods (discussed in Section 3.2.1) and column-by-column methods (discussed in Section 3.2.2). The penalized likelihood methods provide various ways of solving:

$$\widehat{\boldsymbol{\Theta}} = \arg\max_{\boldsymbol{\Theta}} \quad \log\det(\boldsymbol{\Theta}) - \operatorname{tr}(\boldsymbol{\Theta}\overline{\boldsymbol{\Sigma}}) - \sum_{i \neq j} P(|\boldsymbol{\Theta}_{i,j}|), \tag{5.19}$$

where $P(|\boldsymbol{\Theta}_{i,j}|)$ is a penalty function applied over the off-diagonals of $\boldsymbol{\Theta}$ to promote sparsity. The various ways of defining $P(|\boldsymbol{\Theta}_{i,j}|)$ and the different routines to solving (5.19) define the different penalized likelihood methods. Note that the graphical lasso (5.5) is an example of penalized likelihood.

The column-by-column methods follow a different approach, combining the conditional distributions of multivariate Gaussians and linear regressions to approximate each column of $\boldsymbol{\Theta}$ separately. Different routines to solve such a cost function constitute the different column-by-column methods. All of the presented column-by-column methods use a specific variation of the lasso regression method to determine the columns of $\boldsymbol{\Theta}$.

3.2.1 Penalized likelihood methods

Most penalized likelihood methods are a variation of the generalized graphical lasso (GL) (5.5) to find sparse $\widehat{\boldsymbol{\Theta}}$. As discussed in Section 1.3, the GL estimator (5.5) is widely considered to be intractable for large datasets because of its memory requirements. Recent research has focused on the usage of heuristics, e.g. by using thresholding to simplify and parallelize matrix calculations. These heuristics often come at the cost of convergence problems [51]. Currently, however, relaxations (e.g. heuristics) are the only viable way to solve the graphical lasso once the number of variables becomes very large [34, 51, 80, 105].

Following the work in Fattahi and Sojoudi [49], Fattahi et al. [50], Mazumder and Hastie [105], Sojoudi [135] an extension of the graphical lasso was derived in Zhang et al. [161] that imposes an a priori sparsity pattern H and penalizes only the off-diagonals of $\widehat{\boldsymbol{\Theta}}$ with a weighted lasso penalty $\lambda_{i,j}$:

$$\begin{aligned} \widehat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta} \succ 0}{\arg\max} \quad & \log\det(\boldsymbol{\Theta}) - \operatorname{tr}(\boldsymbol{\Theta}\overline{\boldsymbol{\Sigma}}) - \sum_{i=1}^{n} \sum_{j=i+1}^{n} \lambda_{i,j} |\boldsymbol{\Theta}_{i,j}| \\ \text{subject to } & \boldsymbol{\Theta}_{i,j} = 0 \quad \forall (i,j) \notin H, \end{aligned} \tag{5.20}$$

This problem is called the restricted graphical lasso (RGL). Here, the sparsity pattern H represents a priori knowledge on the correlation structure. For example, in our target application each variable represents the ocean flow state at a particular grid point and we can assume that variables that are geographically far apart are (approximately) pairwise conditionally independent, see Assumption 3 in section 1.1.

The GL estimator $\widehat{\boldsymbol{\Theta}}$ can be recovered in a more computationally efficient way by solving a variation of (5.5) that imposes the sparsity pattern G on $\widehat{\boldsymbol{\Theta}}$ Fattahi and Sojoudi [49], Fattahi et al. [50], Zhang et al. [161]:

$$\begin{aligned} \widehat{\boldsymbol{\Theta}} = \underset{\boldsymbol{\Theta} \succ 0}{\arg\max} \quad & \log\det(\boldsymbol{\Theta}) - \operatorname{tr}(\boldsymbol{\Theta}\boldsymbol{\Sigma}_{\lambda}) \\ \text{subject to } & \boldsymbol{\Theta}_{i,j} = 0 \quad \forall (i,j) \notin G. \end{aligned} \tag{5.21}$$

where the soft-thresholded sample covariance matrix $\mathbf{\Sigma}_\lambda$ is given by:

$$(\mathbf{\Sigma}_\lambda)_{i,j} = \begin{cases} \overline{\mathbf{\Sigma}}_{i,j}, & \text{if } i = j, \\ \overline{\mathbf{\Sigma}}_{i,j} - \lambda, & \text{if } \overline{\mathbf{\Sigma}}_{i,j} > \lambda \text{ and } i \neq j, \\ 0, & \text{if } |\overline{\mathbf{\Sigma}}_{i,j}| \leq \lambda, \text{ and } i \neq j, \\ \overline{\mathbf{\Sigma}}_{i,j} + \lambda, & \text{if } \overline{\mathbf{\Sigma}}_{i,j} \leq -\lambda \text{ and } i \neq j. \end{cases} \tag{5.22}$$

The problem in (5.21) is named maximum determinant matrix completion (MDMC). The graph associated with G (that we also call G, unless the difference becomes important) is an undirected graph with K nodes and edges between vertices i and j if and only if $(i, j) \in G$.

The MDMC problem has a closed-form solution, that is extremely fast to compute if G is chordal [50]. Some state-of-the-art iterative solutions embed a nonchordal graph G in a chordal graph and solve the resulting problem as a semidefinite program with an interior-point method [1, 2, 32]. Recently, Zhang et al. [161] proposed an efficient Newton-CG (conjugate gradient) method also based on chordal embedding. They show that their method is highly efficient both from a computational and convergence perspective. They also demonstrate that the MDMC solution to the RGL problem supports high-dimensional problems. Specifically, they show that their algorithm converges to an ϵ-accurate solution in $O(K \log \epsilon^{-1} \log \log \epsilon^{-1}) \approx O(K)$ time and $O(K)$ memory. The only costly step in their suggested methodology is the soft-thresholding of the sample covariance matrix which is $O(K^2)$ in time and memory. We do not see this as a problem because this step is fully parallelizable.

The MDMC problem relies on the equivalence between the thresholding techniques and the GL. There are certain conditions on the GL estimator that guarantee this equivalency [135]. However, these conditions are impractical in use because they can not be verified without solving the GL. A follow-up study [49] found more practical restrictions that can be verified using only the sample covariance matrix. While these restrictions are easier to verify, doing so is still not trivial. Most practically, Fattahi and Sojoudi [49] additionally show that the thresholding techniques (5.21) and RGL (5.20) are equivalent when the ℓ_1-regularization parameter λ in (5.5) is "large". We consider this a very loose restriction because sparsity is precisely what we strive for.

3.2.2 Column-by-column methods

The column-by-column precision matrix estimation methods [46] use the connection between conditional multivariate Gaussian distributions and linear regression to separately recover sparse estimates for each column of $\boldsymbol{\Theta}$. This approach is restricted by the assumption that each column of $\boldsymbol{\Theta}$ is sparse. In exchange, column-by-column estimation methods have the benefits of computational simplicity and more evident avenues for theoretical analysis.

If we let $\boldsymbol{y}^n \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$, $\sigma_j^2 = \Sigma_{j,j} - \boldsymbol{\Sigma}_{j,\backslash j}(\boldsymbol{\Sigma}_{\backslash j,\backslash j})^{-1}\boldsymbol{\Sigma}_{\backslash j,j}$, and $\boldsymbol{\alpha}_j = \left(\boldsymbol{\Sigma}_{\backslash j,\backslash j}\right)^{-1}\boldsymbol{\Sigma}_{\backslash j,j}$ where $\boldsymbol{\Sigma}_{\backslash j,j}$ is the $(K-1)\times K$ matrix equal to $\boldsymbol{\Sigma}$ excluding the jth row, then after some algebra on conditional distributions one can show:

$$y_j^n \,|\, \boldsymbol{y}_{\backslash j}^n \;\sim\; \mathcal{N}(\boldsymbol{\alpha}_j^T \boldsymbol{y}_{\backslash j}^n, \sigma_j^2),$$

where $\boldsymbol{y}_{\backslash j}^n$ is the vector of length $K-1$ equal to $\boldsymbol{y}^n$ excluding y_j^n. Equivalently, one can regress y_j^n on $\boldsymbol{y}_{\backslash j}^n$ through the linear regression model

$$y_j^n = \boldsymbol{\alpha}_j^T \boldsymbol{y}_{\backslash j}^n + \epsilon_j,$$

where $\epsilon \sim \mathcal{N}(0, \sigma_j^2)$ is independent of $\boldsymbol{y}_{\backslash j}$. This results in a column-by-column approximation of $\boldsymbol{\Theta}$ through the block matrix inversion formula:

$$\boldsymbol{\Theta}_{j,j} = \sigma_j^{-2}, \qquad \boldsymbol{\Theta}_{\backslash j,j} = -\sigma_j^{-2}\boldsymbol{\alpha}_j, \tag{5.23}$$

Therefore, one can recover column j of $\boldsymbol{\Theta}$ by estimating regression coefficients $\boldsymbol{\alpha}_j$ and residual variance σ_j^2.

Two examples of column-by-column methods are the CLIME and Tiger methods, as discussed in Sections 3.2.2 and 3.2.2, respectively.

CLIME The Constrained ℓ_1-Minimization for Inverse Matrix Estimation (CLIME) [23] directly estimates the jth columns of $\boldsymbol{\Theta}$ by solving:

$$\begin{aligned} &\widehat{\boldsymbol{\Theta}}_{:,j} = \underset{\boldsymbol{\Theta}_{:,j}}{\arg\min} \|\boldsymbol{\Theta}_{:,j}\|_1 \\ &\text{subject to } \left\|\overline{\boldsymbol{\Sigma}}\boldsymbol{\Theta}_{:,j} - \boldsymbol{e}_j\right\|_\infty \leq \delta_j, \qquad \text{for } j = 1, \ldots, K, \end{aligned} \tag{5.24}$$

where $\boldsymbol{\Theta}_{:,j}$ is the jth column of $\boldsymbol{\Theta}$, $\boldsymbol{e}_j$ is the jth unit vector and δ_j is a tuning parameter. The CLIME estimator is symmetric by its definition [23]. It is also shown in Cai et al. [23] that (5.24) can be separated into K vector minimization problems. This allows for better parallelization in case of high-dimensional problems, but each of the K vector minimization problems still depends on the full sample covariance matrix $\overline{\boldsymbol{\Sigma}}$. This requires the sample covariance matrix $\overline{\boldsymbol{\Sigma}}$ to be in memory. However, the solver CLIME-ADMM [148] parallelizes the CLIME estimator in column-blocks, scaling well to high-dimensional problems. Cai et al. [23] also shows that the estimator $\widehat{\boldsymbol{\Theta}}$ (5.24) is positive definite with high probability.

One downside of the CLIME estimator (5.24) is the need to tune the regularization parameter δ_j. Typically, this is done with k-fold cross-validation tests.

TIGER The Tuning-Insensitive Graph Estimation and Regression (TIGER) method [98] provides a tuning-insensitive approach for estimating the precision matrix that achieves minimax optimal rates of convergence in precision matrix estimation.

Let us use the σ and $\boldsymbol{\alpha}$ variables (see Section 3.2.2) to introduce some useful notation: the $K \times K$ diagonal matrix $\widehat{\boldsymbol{\Gamma}} := \operatorname{diag}(\widehat{\boldsymbol{\Sigma}})$, the normalized data matrix $\boldsymbol{Z} = (\boldsymbol{z}_1, \dots, \boldsymbol{z}_K)^T := \boldsymbol{X}\widehat{\boldsymbol{\Gamma}}^{-1/2}$, $\boldsymbol{\beta}_j := \widehat{\Sigma}_{j,j}^{-1/2}\widehat{\boldsymbol{\Gamma}}^{1/2}_{\setminus j,\setminus j}\boldsymbol{\alpha}_j$, $\tau_j^2 = \widehat{\sigma}_j^2\widehat{\Sigma}_{j,j}^{-1}$, and the sample correlation matrix $\widehat{\boldsymbol{R}} := \widehat{\boldsymbol{\Gamma}}^{-1/2}\overline{\boldsymbol{\Sigma}}\widehat{\boldsymbol{\Gamma}}^{-1/2}$. Then, the TIGER method uses the following estimator for each precision matrix column:

$$\begin{aligned}
\widehat{\boldsymbol{\beta}}_j &:= \operatorname*{arg\,min}_{\boldsymbol{\beta}_j \in \mathbb{R}^{K-1}} \left[\frac{1}{\sqrt{N}} \left\| \boldsymbol{Z}_{:,j} - \boldsymbol{Z}_{:,\setminus j}\boldsymbol{\beta}_j \right\|_2 + \lambda \|\boldsymbol{\beta}_j\|_1 \right], \\
\widehat{\tau}_j &:= \frac{1}{\sqrt{N}} \left\| \boldsymbol{Z}_{:,j} - \boldsymbol{Z}_{:,\setminus j}\widehat{\boldsymbol{\beta}}_j \right\|_2, \\
\widehat{\boldsymbol{\Theta}}_{j,j} &= \widehat{\tau}_j^{-2}\widehat{\Sigma}_{j,j}^{-1} \quad \text{and} \quad \widehat{\boldsymbol{\Theta}}_{\setminus j,j} = -\widehat{\tau}_j^{-2}\widehat{\Sigma}_{j,j}^{-1/2}\widehat{\boldsymbol{\Gamma}}_{\setminus j,\setminus j}^{-1/2}\widehat{\boldsymbol{\beta}}_j,
\end{aligned} \tag{5.25}$$

where λ is a tuning parameter that can be chosen such that it gives optimal convergence, as shown in Liu et al. [98]. Specifically, Liu et al. [98] shows that the estimators achieves minimax optimal rates of convergence when the tuning parameter $\lambda = \pi\sqrt{\log(K)/2N}$ is chosen. For finite samples they recommend to select optimal $\lambda = \zeta\pi\sqrt{\log(K)/2N}$ where ζ sweeps the interval $[\sqrt{2}/\pi, 1]$. Because ζ is completely independent of any unknowns, one calls

TIGER a tuning-insensitive approach.

The only non-diagonal matrix involved in obtaining the TIGER estimator (5.25) is the $N \times K$ normalized data matrix $\boldsymbol{Z}$. Therefore, the TIGER estimator scales well to high-dimensional problems (see Limitation 6 in section 1.1).

3.3 Sparse precision root estimation

Instead of first estimating a sparse covariance or precision matrix and then applying Cholesky decomposition to get the root matrix, one can aim to estimate a sparse covariance or precision root directly. Various works have proposed methods to achieve this. These works can be split into two categories, those that impose a specific sparsity pattern [121, 156, 158] and those that promote general sparsity in the root [82, 88, 132, 134]. The difficulties associated with ordering two-dimensional grid variables to get specific sparsity patterns motivate a general sparsity approach.

Specifically, we present in this section the methodology proposed by Khare et al. [88] because, unlike the other approaches mentioned, it supports high-dimensional problems. Additionally, the objective function considered in Khare et al. [88] is jointly convex in the nonredundant entries of $\boldsymbol{L}$ and the objective function is bounded away from $-\infty$ even when $N < K$. Lack of a lower bound is an issue with the estimator proposed in Shojaie and Michailidis [132], wherein the global minimum of the objective function is at $-\infty$ when $N < K$.

3.3.1 CSCS

Khare et al. [88] consider the following Convex Sparse Cholesky Selection (CSCS) objective function:

$$Q_{CSCS}(\boldsymbol{L}) = 2\log|\boldsymbol{L}| - tr(\boldsymbol{L}^T\boldsymbol{L}\overline{\boldsymbol{\Sigma}}) - \lambda \sum_{1\leq j<i\leq K} |L_{ij}|, \tag{5.26}$$

where the first two terms correspond to the Gaussian log-likelihood terms and the third term induces sparsity in the off-diagonals of $\boldsymbol{L}$. Note that we added a minus sign to the objective to be consistent with other GL based objectives described in previous chapters. To demonstrate attractive theoretical and computational properties Khare et al. [88] shows that optimizing (5.26) is equivalent to separately optimizing for every $1 \leq i \leq K$:

$$Q_{CSCS,i}(\boldsymbol{\eta}^i) = 2\log \eta_i^i - (\boldsymbol{\eta}^i)^T(\overline{\boldsymbol{\Sigma}}^i)\boldsymbol{\eta}^i - \lambda \sum_{j=1}^{i-1} |\eta_j^i|, \tag{5.27}$$

where, for each $1 \leq i \leq K$, $\boldsymbol{\eta}_i := L_{i,1:i}$ denotes the vector of lower triangular and diagonal entries in the ith row of $\boldsymbol{L}$ and $\overline{\boldsymbol{\Sigma}}^i := \overline{\boldsymbol{\Sigma}}_{1:i,1:i}$ is the $i \times i$ submatrix of the first i rows and columns of $\overline{\boldsymbol{\Sigma}}$. This separated form is very suitable for parallelization.

Finally, Khare et al. [88] propose a cyclic coordinatewise maximization algorithm that maximizes (5.27) for each $1 \leq i \leq K$ by circulating a single varying element of $\boldsymbol{\eta}^i$ in order to iteratively move to a maximum for (5.27). Specifically, they show that the supremum for a single varying element η_a^i of $\boldsymbol{\eta}^i$ can be expressed in closed form as M_a for $1 \leq a \leq i$:

$$M_a(\boldsymbol{\eta}^i) = -\frac{f_\lambda(-2\sum_{b\neq a} \overline{\Sigma}_{ba}^i \eta_b^i)}{2\overline{\Sigma}_{aa}^i}, \tag{5.28}$$

for $1 \leq a \leq i-1$, and

$$M_i(\boldsymbol{\eta}^i) = \frac{\sum_{b\neq i} \overline{\Sigma}_{bi}^i \eta_b^i - \sqrt{(\sum_{b\neq i} \overline{\Sigma}_{bi}^i \eta_b^i)^2 + 4\overline{\Sigma}_{ii}^i}}{2\overline{\Sigma}_{ii}^i}, \tag{5.29}$$

where $f_\lambda(x) = \text{sign}(x)(|x| - \lambda)_+$ is a soft-thresholding operator. While the cyclic coordinatewise maximization algorithm solves a convex objective function that is not strictly convex, it does converge to a global maximum [88]. Moreover, this maximum leads to a positive definite estimate of the covariance matrix. Under standard regularity assumptions, Khare et al. [88] establish high-dimensional asymptotic consistency as N approaches infinity (both in terms of model selection and estimation).

Because each of the $\boldsymbol{\eta}^i$ need to be maximized (for $1 \leq i \leq K$) and each maximization problem is separate, this step can be completely parallelized. While not explicitly discussed or noted in Khare et al. [88], an additional benefit of this optimization method is that memory usage scales well to high-dimensional application because each evaluation of M_a for $1 \leq a \leq i$ Equations (5.28)–(5.29) needs only a vector of length i in memory, i.e. the first i entries of the ath row of $\overline{\boldsymbol{\Sigma}}$. This means that the full sample covariance matrix does not need to be fully in memory at any point.

Computational complexity of the CSCS algorithm, as presented in this section, is dependent on whether the K separate maximizations in (5.27) are parallelized [88]. When fully parallelized the computational complexity is $\min(O(KN), O(K^2))$. When fully sequential the computational complexity is $\min(O(NK^2), O(K^3))$.

4 Numerical results

Here we compare the performance of each of the methodologies presented in Section 3. We generate time series data $\boldsymbol{Y}$ with the prototype models described in section 2 and apply the different estimation methods to these data. We choose grid sizes for the prototype models that are not very large so that we can explicitly calculate the sample covariance matrix $\overline{\boldsymbol{\Sigma}}$ and invert it to get the precision matrix. It enables us to compare the results of the estimation methods with the sample covariance. This comparison can be seen as a so-called *oracle test*, with the sample covariance matrix acting as *oracle* reference or benchmark (in practical situations with high-dimensional data, such a reference is not available because it is too large to calculate due to memory constraints).

Besides comparing to the sample covariance, we also assess the sparsity of the matrices resulting from the estimation methods. Below we list the criteria by which we compare the performance of the different methods:

$$\text{Sparsity ratio: } r(\widehat{\cdot}) = \frac{1}{K^2} \sum_{i,j\in\{1,\ldots,K\}} \mathbf{1}(\widehat{\cdot}_{i,j} = 0). \text{ Applied either to } \widehat{\boldsymbol{L}} \text{ or to } \widehat{\boldsymbol{S}}, \text{ depending on the methodology.} \tag{5.30}$$

$$\text{Eigenvalue spectrum: } (\lambda_i)_{i\in\{1,\ldots,K\}} \text{ compared to } (\widehat{\lambda}_i)_{i\in\{1,\ldots,K\}}, \text{ where } \overline{\boldsymbol{\Sigma}}\boldsymbol{v} = \lambda\boldsymbol{v} \text{ and } \widehat{\boldsymbol{\Sigma}}\widehat{\boldsymbol{v}} = \widehat{\lambda}\widehat{\boldsymbol{v}}, \text{ respectively.} \tag{5.31}$$

$$\text{Normalized absolute error: } \epsilon(\widehat{\boldsymbol{\Sigma}}) := \left\| \widehat{\boldsymbol{\Sigma}} - \overline{\boldsymbol{\Sigma}} \right\|_F / \left\| \overline{\boldsymbol{\Sigma}} \right\|_F, \tag{5.32}$$

where the sparsity ratio $0 \leq r(A) \leq 1$, such that a fully dense matrix A and a zero matrix A correspond to $r(A) = 0$ and $r(A) = 1$, respectively. Criteria (5.31)-(5.32) are concerned with the accuracy of the estimates.

4.1 Practical matters

Consistency in comparing different methods The different methodologies discussed in Section 3 have different output matrices. Therefore, we post-process the results to a common output appropriate to criteria in Equations (5.30)–(5.32). For example, the CSCS method in Section 3.3.1 estimates a sparse root precision matrix approximation $\widehat{\boldsymbol{L}}$. In order to compare the normalized absolute error (5.32), we compute the matrix square $\widehat{\boldsymbol{\Theta}} = \widehat{\boldsymbol{L}}^T\widehat{\boldsymbol{L}}$ and invert it explicitly to find the associated covariance approximation $\widehat{\boldsymbol{\Sigma}}$.

Each of the estimation methods has a parameter that influences the sparsity of the resulting matrix. We apply the methods over a range of parameter values, to get insight on the balance between sparsity (5.30) and accuracy (as assessed by criteria (5.31)-(5.32)). The thresholding methods (see (5.11)) are sampled along the ω-dimension to vary sparsity of $\widehat{\boldsymbol{\Sigma}}_\omega$. The banding methods (see 5.12) and tapering methods (see (5.13)) are varied similarly by varying k and d_{max} between 1 and K or $K\sqrt{2}$, for (5.6) and (5.7), respectively. The MDMC method (5.21) is sampled by varying λ and equating the sparsity pattern G to the sparsity pattern of soft-thresholded matrix $\boldsymbol{\Sigma}_\lambda$ (5.22). The CLIME estimator (5.24) is tested by varying the $\delta_j := \delta$ (the prototype model is spatially homogeneous). Similarly, for the TIGER estimator (5.25) we vary λ. The CSCS method (5.27) is tested by varying the λ parameter as well.

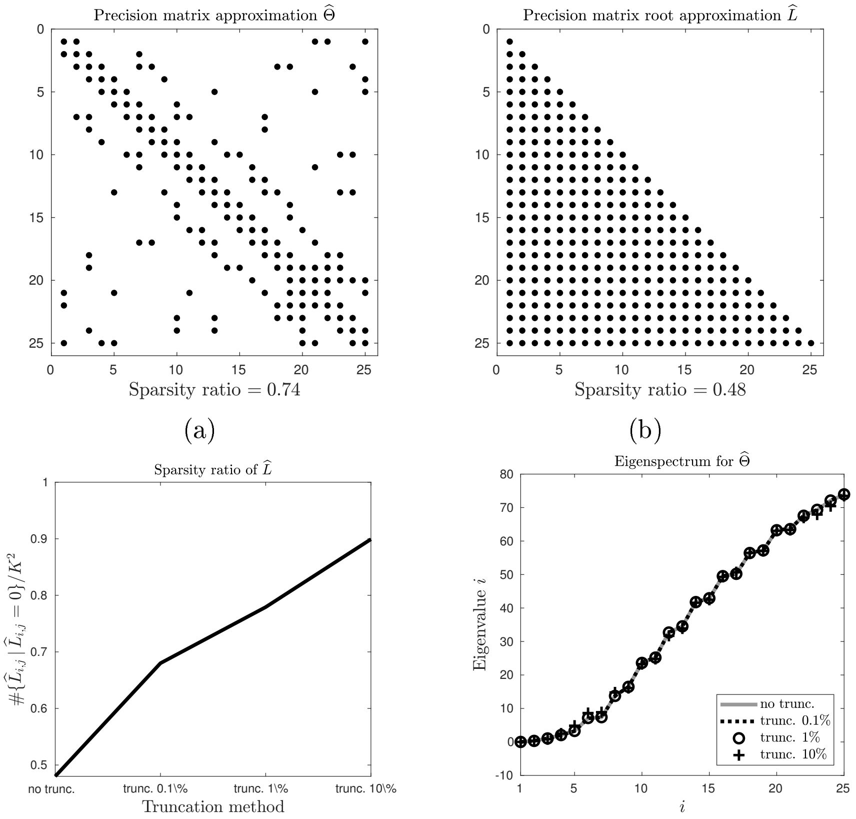
Assessing methodologies that do not guarantee positive definiteness In quite a few of the methodologies, a sparse $\widehat{\boldsymbol{\Sigma}}$ (or $\widehat{\boldsymbol{\Theta}}$) is estimated that then is used to explicitly calculate matrix root $\widehat{\boldsymbol{S}}$ (or $\widehat{\boldsymbol{L}}$) by Cholesky decomposition, such that $\widehat{\boldsymbol{\Sigma}} = \widehat{\boldsymbol{S}}\widehat{\boldsymbol{S}}^T$ (or $\widehat{\boldsymbol{\Theta}} = \widehat{\boldsymbol{L}}^T\widehat{\boldsymbol{L}}$). However, two problems can arise in practice.

First, not all of the methodologies guarantee positive definiteness in the estimated covariance or precision matrices. This is a serious shortcoming because a covariance matrix should be positive definite by definition. Additionally, it makes it impossible to do Cholesky decomposition. While some of these methods do not guarantee positive definiteness, they do strive for it (via a penalty term in the objective function). In practical tests, this frequently results in the situation that one or a few eigenvalues of the resulting estimators happen to be negative and very small in magnitude. In that case, we truncate such small negative eigenvalues u_i to zero to make the estimator non-negative definite. This allows one to get an impression of the accuracy and sparsity of the matrix root approximation. Then, using this truncated matrix we construct $\boldsymbol{R}^T = \boldsymbol{U}^{1/2}\boldsymbol{V}^T$, where $\boldsymbol{U}$ is the diagonal matrix of truncated eigenvalues and $\boldsymbol{V}$ is the eigenvector matrix. The resulting $\boldsymbol{R}^T$ matrix approximates the lower triangular matrix root of the covariance or precision matrix. One can consider other ways to resolve this issue but we do not explore them here.

Second, if the matrix root can be calculated, e.g. with the Cholesky decomposition or a QR decomposition, this matrix root does not necessarily inherit the sparsity of the covariance or precision matrix (as also discussed in section 1.2). The CSCS method as presented in Section 3.3.1 explicitly encourages sparsity in the matrix root, however the other methods do not.

Estimator values close to zero A secondary issue that arises from sequentially combining several numerical approximation methods (e.g. a sparse estimation of the precision matrix, followed by either numerical inversion or Cholesky decomposition) is the introduction of very small nonzero elements. Figure 5.3 illustrates that, while this approximated precision matrix is relatively sparse (Figure 5.3a), the associated matrix root is a fully dense lower triangular matrix (Figure 5.3b). However, many of the elements in this associated matrix root are very small nonzero elements. Their impact is relatively negligible as illustrated by Figures 5.3c and 5.3d, wherein a variety of options are shown that truncate $\widehat{\boldsymbol{L}}$ elements at values relative to the mean absolute diagonal. Figure 5.3c shows that even slight truncation $\widehat{\boldsymbol{L}}$ elements can result in significantly increased sparsity. Furthermore, Figure 5.3d shows that these different options have negligible effects on the precision matrix eigenspectra (that result from squaring the truncated matrix roots).

In this overview of results we truncate matrix root estimator values at 1% of its mean absolute diagonal, if and only if this does not significantly impact the eigenspectrum of the estimated matrix.



(a) (b)

(c) (d)

Figure 5.3: Example of handling very small nonzero values in estimators. Figure (a) shows the sparsity plot (26%) for an approximated precision matrix $\widehat{\boldsymbol{\Theta}}$ that is not positive definite, (b) shows the fully dense sparsity plot (52%) for the lower triangular matrix root $\widehat{\boldsymbol{L}}$ following from a QR decomposition, (c) shows the sparsity of several truncated versions of $\widehat{\boldsymbol{L}}$ at cutoffs proportional to its mean absolute diagonal, (d) shows the eigenspectra associated with the different truncated versions of $\widehat{\boldsymbol{L}}$.

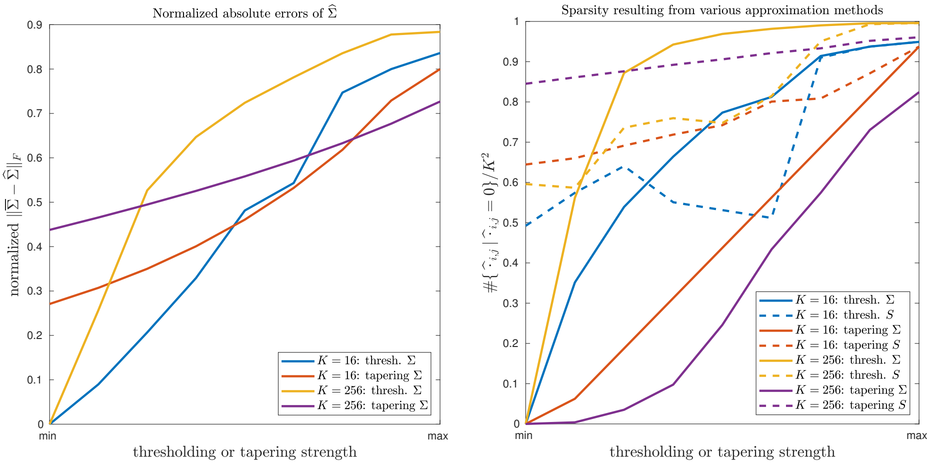
4.2 Sparse covariance results

In this section we compare the methods presented in Section 3.1. To observe the performance of each of these methodologies, we "sweep" their parameter space as specified in Section 4.1 in the $K = 16$ and $K = 256$ model configurations (see Table 5.2). To prevent redundancy in the presentation we focus exclusively on the thresholding and the Wendland tapering methods, at the end of this section we briefly summarize the omitted redundancies with regard to other tapering methods.

Figure 5.4 shows that neither the thresholding nor the Wendland tapering achieves simultaneously good sparsity and accuracy. By definition, minimal thresholding results in zero errors as well as zero sparsity. Also, by definition, maximal thresholding results in diagonal estimators with very high errors. It is evident from Figures 5.4a and 5.4b that the thresholding is more successful than the Wendland tapering in achieving high sparsity when comparing at similar error levels in $\widehat{\boldsymbol{\Sigma}}$ for both $K = 16$ and $K = 256$. On top of that, the tapering estimators have substantial errors even as sparsity reaches 0. Clearly, this is highly undesirable. The thresholding method is successful in introducing sparsity indirectly in $\widehat{\boldsymbol{\Sigma}}$, however the sparsity in $\widehat{\boldsymbol{S}}$ increases non-monotonically across a large range of thresholding strength ω. Additionally, high sparsity in $\widehat{\boldsymbol{S}}$ as introduced by the thresholding consistently comes at the cost of large errors.

The banding methods (5.12) perform roughly equivalently to the thresholding in the case of the $K = 16$ model. However, the banding estimators struggle more to introduce sparsity in the covariance root for the $K = 256$ model. This is because enforcing sparsity on the covariance by the banding method does not generate a matrix with small bandwidth in case of a 2D grid. The Cholesky decomposition can therefore give significant fill-in, reducing the sparsity. As for the tapering methods, each of the Wendland covariances performs similarly as the $\psi_{3,2}$ covariance function illustrated in Figure 5.4. On the other hand, the Epanechnikov covariance (5.15) gives reduced errors in $\widehat{\boldsymbol{\Sigma}}$ significantly, but fails to introduce sparsity in the covariance root.

Because none of the tested sparse covariance matrix estimation methods achieve both high accuracy and sparsity in the associated covariance root, we consider these methods to be unsuitable.



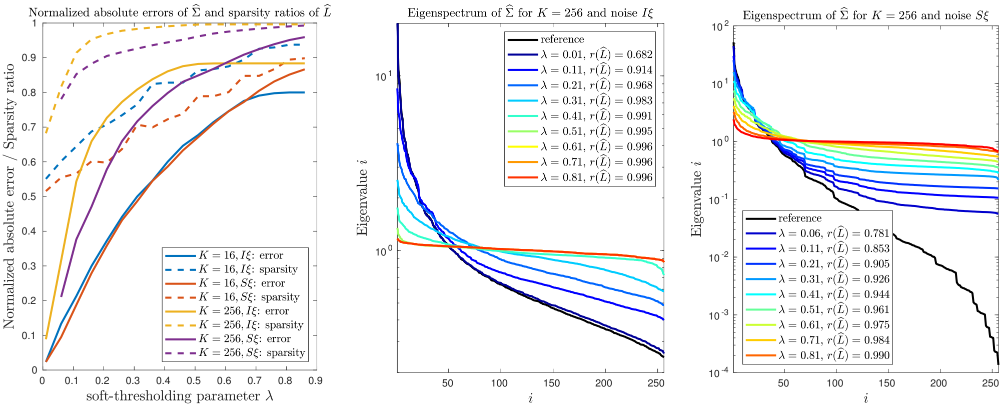
(a) (b)

Figure 5.4: Highlighted results for thresholding (5.11) and Wendland ($\psi_{3,2}$, 5.18) covariances for the $K = 16$ (5.6) and $K = 256$ (5.7) prototype models as specified in Table 5.2. Figure (a) shows the normalized absolute error (5.32) of the approximated covariance $\widehat{\boldsymbol{\Sigma}}$. Figure (b) shows the sparsity ratio (5.30) of both the approximated covariance $\widehat{\boldsymbol{\Sigma}}$ and the approximated covariance root $\widehat{S}$. The evaluation of the different configurations in Figures (a) and (b) is matched to the minimum and maximum sparsity ratios obtained with each method.

4.3 Sparse precision results

Here we compare the methods presented in Section 3.2 for sparse precision matrix estimation. As in the previous section, for the comparison we "sweep" their parameter space as specified in Section 4.1 for the $K = 16$ and $K = 256$ model configurations (see Table 5.2). Let us start with the MDMC method as presented in Section 3.2.1. Figure 5.5a illustrates that the MDMC is unable to introduce sparsity while achieving low errors for the $K = 16$ use cases. However, for the $K = 256$ the MDMC estimators achieve both high sparsity and low errors for values $\lambda \approx 0.06$. This is presumably because the 2D grid contains a higher proportion of conditionally independent grid points. Despite this mix of positive and negative results, the MDMC solver shows desirable behavior in that the error term converges mostly linearly to 0 as λ approaches 0. This desirable behavior is exhibited for both the $K = 16$ and $K = 256$ models, but more effectively for the $K = 256$ use case.

In contrast to the desirable accuracy convergence, Figure 5.5c shows that only the leading part of the reference covariance eigenspectrum (approximately 33% of all eigenvalues) is adequately reproduced by the $K = 256$ model (5.7) with $\boldsymbol{S}\xi$ noise term (5.9) for $\lambda = 0.06$, i.e. the best performing regularization setting. While the eigenspectrum of the $K = 256$ model (5.7) with $\boldsymbol{I}\xi$ noise term (5.8) is well reproduced by the MDMC estimator with $\lambda = 0.01$, the eigenspectra of the $K = 256, \boldsymbol{S}\xi$ case are unsatisfactory.

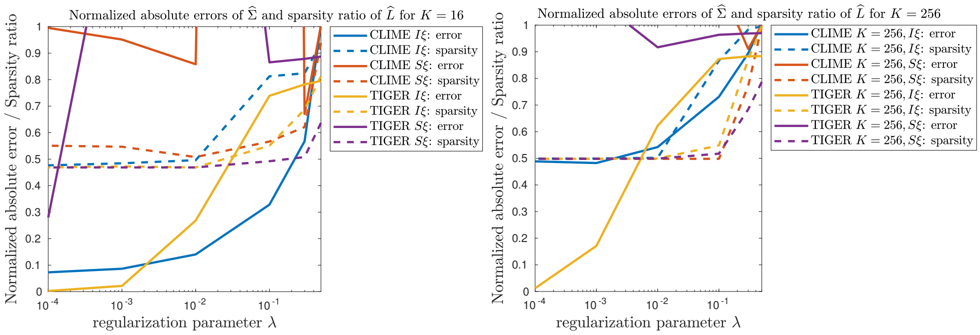


(a) (b) (c)

Figure 5.5: Results for MDMC (5.21) precision matrices for the $K = 16$ (5.6) and $K = 256$ (5.7) prototype models as specified in Table 5.2. Figure (a) shows the normalized absolute errors and sparsity ratios of the approximated precision matrices $\widehat{\boldsymbol{\Theta}}$. Figures (b) and (c) show the eigenspectra of the corresponding covariance matrices $\widehat{\boldsymbol{\Sigma}}$ for a representative range of λ values for both the $K = 256$ case with an $I\xi$ (b) and $S\xi$ (c) noise terms.

Note that the MDMC algorithm can not be properly initialized because it fails to find a chordal embedding for $\lambda < 0.06$ for the $K = 256$ model with $\boldsymbol{S}\xi$ noise term (5.9). This is a practical limitation of the code used in Khare et al. [88]. Because of this limitation and the unsatisfactory results discussed above we reject the MDMC approach for our purposes.

The results for the column-by-column estimation methodologies (see Section 3.2.2) prove to be generally unsatisfactory for achieving simultaneous sparsity and accuracy. Figure 5.6 illustrates the normalized absolute errors and sparsity ratios for the $K = 16$ and $K = 256$ (see Table 5.2). Despite the sparsity of the approximated $\widehat{\boldsymbol{L}}$ varying between fully dense lower triangular and diagonal over the sampled range of λ, the errors of the estimators only approach 0 in a few cases: the TIGER estimators at $10^{-4} \leq \lambda \leq 10^{-3}$ for both $K = 16$ and $K = 256$ with $\boldsymbol{I}\xi$ noise, but the CLIME estimators at $10^{-2} \leq \lambda \leq 10^{-4}$ only for the $K = 16$ model with $\boldsymbol{I}\xi$ noise. None of these configurations, however, manage to introduce sparsity in $\widehat{\boldsymbol{L}}$. Both Figures 5.6a and 5.6b clearly show that significant errors are introduced when λ is increased in order to increase sparsity. Worst of these use cases are all CLIME and TIGER estimators for the $\boldsymbol{S}\xi$ noise terms. Most of these estimators show $\mathcal{O}(1)$ errors over the sampled range of λ. Moreover, these high error levels are also accompanied by very low sparsity ratios. Both CLIME and TIGER estimators for the $\boldsymbol{S}\xi$ noise model correspond to almost fully dense lower triangular sparsity pattern for all $\lambda \leq 10^{-1}$.



(a) (b)

Figure 5.6: Accuracy and sparsity results for the CLIME (5.24) and TIGER (5.25) estimators. Figures (a) and (b) show the normalized absolute errors and sparsity ratios of the covariance matrices $\widehat{\boldsymbol{\Sigma}}$ resulting from the estimators for a range of regularization parameters for both the $K = 16$ (a) and $K = 256$ (b) models.

4.4 CSCS results

4.4.1 $AR(1)$ test

Before we discuss the “oracle tests” like in previous sections, let us start with a test case where the sparse precision matrix is known analytically. We do this to assess unambiguously whether the CSCS method is able to recover the precision root with correct properties in a case where it is known that the true precision matrix root is sparse. This can be considered a sanity test of the methodology. To this end let us consider a scalar zero-mean autoregressive process of order p

$(AR(p))$:

$$y^n = a_1 y^{n-1} + \ldots + a_p y^{n-p} + \xi^n, \quad \text{where} \;\; \xi^n \sim \mathcal{N}(0, \sigma^2), \;\; i.i.d. \tag{5.33}$$

Here we consider a zero-mean process for simplicity. The extension to a non-zero mean process is straightforward and can be found by adding a constant offset to (5.33). Let us begin with some analytical observations to connect the precision matrix to several properties of an $AR(p)$ process.

In general, if the covariance matrix $\mathbf{\Sigma}$ is positive definite and, therefore, invertible, then the partial autocorrelation between pairs x and z can be defined through the precision matrix $\mathbf{\Theta}$ ([26]). For the purposes of analysis of the scalar $AR(p)$ we consider here the covariance (and thus precision) matrix to be defined temporally: $\mathbf{\Sigma}_{mn} = \mathrm{Cov}(y^m, y^n)$. Then the partial autocorrelation function between y^m and y^n, i.e. $\mathrm{PACF}(y^m, y^n)$ is given by:

$$\mathrm{PACF}(y^m, y^n) = \mathbf{\Theta}_{mn} / \sqrt{\mathbf{\Theta}_{mm} \mathbf{\Theta}_{nn}}, \tag{5.34}$$

Simultaneously, the partial correlations $\mathrm{PACF}(y^n, y^{n-i})$ of an $AR(p)$ process are given by the AR coefficient a_i ([31]). Therefore, combining the above two observations, we know that $\mathbf{\Theta}_{ij} = 0$ if $|i - j| > p$ for an $AR(p)$ process. Thus, the precision matrix of an $AR(p)$ process is a banded matrix with bandwidth p. This banded precision pattern translates directly to sparse precision matrix roots (as discussed in Section 1.2).

In order for an $AR(p)$ process to be stationary, the model coefficients in (5.33) need to satisfy the $AR(p)$ stability constraint [100]:

$$\forall w : \quad w^p - a_1 w^{p-1} - \ldots - a_p = 0, \quad \Rightarrow \quad |w| < 1, \tag{5.35}$$

where the notation w^p means the root w to the power p.

Additionally, in order for an $AR(p)$ *time series* to be stationary, the marginal distributions of the p initial states must equal the asymptotic distribution of the process.

To find an analytic formula for the precision matrix of an $AR(p)$ process, let us define the joint distribution $P(\boldsymbol{y})$ in two equivalent ways. First, using Bayes theorem, and secondly, using the definition of a multivariate Gaussian:

$$\begin{aligned} P(y^1,\ldots,y^N) &= P(y^1)\ P(y^2 \mid y^1)\ \cdots\ P(y^N \mid y^{N-1},\ldots,y^1) \\ &\propto \exp\left(-\frac{1}{2}\boldsymbol{y}^T\boldsymbol{\Theta}\boldsymbol{y}\right) \end{aligned} \tag{5.36}$$

Concluding, one can show that an $AR(1)$ with the following properties results in a stationary time series:

- $|a_1| < 1$,
- $y^1 \sim \mathcal{N}\left(0, \frac{\sigma^2}{1-a_1^2}\right)$,

and one can show that the covariance matrix $\boldsymbol{\Sigma}$ for an $AR(1)$ is given by [97]:

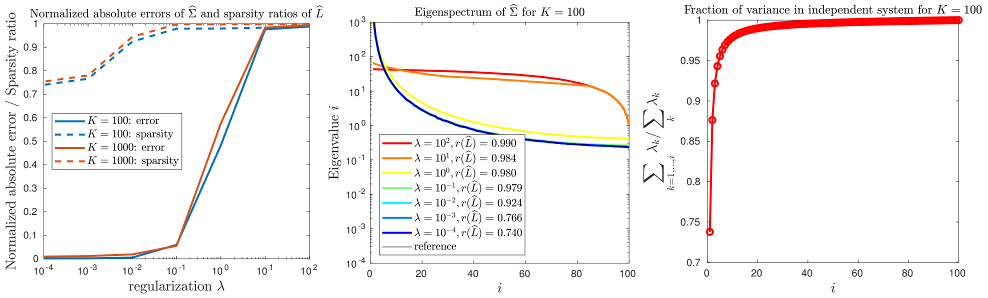
$$\boldsymbol{\Sigma} = \frac{\sigma^2}{1-a_1^2}\begin{bmatrix} 1 & a_1 & \ldots & a_1^{N-2} & a_1^{N-1} \\ a_1 & 1 & \ldots & a_1^{N-3} & a_1^{N-2} \\ \vdots & \vdots & & \ddots & \vdots & \vdots \\ a_1^{N-2} & a_1^{N-3} & \ldots & 1 & a_1 \\ a_1^{N-1} & a_1^{N-2} & \ldots & a_1 & 1 \end{bmatrix}. \tag{5.37}$$

Whereas the covariance matrix is fully dense, one can easily verify that the precision matrix $\boldsymbol{\Theta}$ for an $AR(1)$ process is very sparse (tridiagonal). It is given by:

$$\boldsymbol{\Theta} = \frac{1}{\sigma^2}\begin{bmatrix} 1 & -a_1 & 0 & \ldots & 0 & 0 \\ -a_1 & 1+a_1^2 & -a_1 & \ldots & 0 & 0 \\ 0 & -a_1 & 1+a_1^2 & \ldots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \ldots & 1+a_1^2 & -a_1 \\ 0 & 0 & 0 & \ldots & -a_1 & 1 \end{bmatrix} \tag{5.38}$$

The experimental test setup is now to use the Cholesky decomposition $\boldsymbol{S}\boldsymbol{S}^T = \boldsymbol{\Sigma}$ to iterate 10.000 independent $AR(1)$ processes with $N = 100$ and $N = 1000$ iteration steps. We then then use the CSCS methodology to directly calculate the sparse precision matrix root $\widehat{\boldsymbol{L}}$, and thereby to find indirectly the approximation $\widehat{\boldsymbol{\Sigma}}$.

Figure 5.7 shows a curated overview of the results obtained with the $AR(1)$ test case as described above. Figure 5.7a shows that the CSCS solver has an optimum around $\lambda = 10^{-1}$ where the normalized absolute errors are around 0.06 and the sparsity is 98% or 99.8%, for $K = 100$ and $K = 1000$, respectively. This corresponds to lower triangular $\widehat{L}$ with bandwidth 1, which is exactly the sparsity ratio of the Cholesky decomposition of (5.38) [122]. Thus, we see precisely the sparsity pattern we expect from theory for $\lambda = 10^{-1}$. Generally, all results with $10^{-2} \leq \lambda \leq 10^{-1}$ are very good in this test, with errors below 10% and sparsity ratio above 90%. Figure 5.7b shows that also the entire eigenspectrum is very closely reconstructed for $\lambda \leq 10^{-1}$, further emphasizing the strong performance of the CSCS method in this test.



(a) (b) (c)

Figure 5.7: Results for CSCS (5.27) precision matrix root estimators for the $K = 100$ $AR(1)$ process (5.33) test model for a variety of regularization parameters λ. Figure (a) shows the accuracy (5.32) and sparsity ratio (5.30) of the precision root estimator. Figure (b) shows the eigenspectra of the covariance matrices resulting from the estimators. Figure (c) shows the fraction of the variance in the independent coordinate system explained by the eigenvalues.

While these results are encouraging, we note that the $AR(1)$ test case shows a steep gradient in fraction of explained variance, see Figure 5.7c. It makes the tested $AR(p)$ process less suitable as a prototype model, because it is not consistent with the assumption that the fraction of explained variance increases gradually (Assumption 8 in section 1.1). Models with such steep increase in eigenvalues motivate a lower-dimensional representation of the covariance, like in Berloff et al. [12]. By contrast, models that show gradual increases in explained fraction of variance are much less amenable to such methods due to the large number of modes required to achieve low truncation errors, see for example the SPDE models (5.6)-(5.7) in Section 2 as illustrated in Figures 5.2e and 5.2f. Therefore, let us return to the oracle tests for the SPDE models to assess the

CSCS method in the next section.

4.4.2 Oracle test

As before, we assess the CSCS methodology with the SPDE prototype model tests by sweeping its parameter space for the $K = 16$, $K = 64$, and $K = 256$ model configurations (see Table 5.2) for both the $\boldsymbol{I}\xi$ (5.8) and $\boldsymbol{S}\xi$ (5.9) noise terms. As this section will show, the CSCS estimators are a significant improvement compared to the previously discussed methodologies. Therefore, let us discuss each of the performance criteria listed in Section 4 as illustrated in Figure 5.8.

Figures 5.8a and 5.8d show that the normalized absolute errors and sparsity ratios of $\widehat{\boldsymbol{\Sigma}}$ and $\widehat{\boldsymbol{L}}$ converge monotonically to their extremes for each model. This is desirable behavior because this makes it straightforward to find an optimal balance between accuracy and sparsity, with e.g, golden section search [90], the optimal [6] minimization method for monotonic functions. How to weigh accuracy versus sparsity is user-dependent; notwithstanding, the optimal balance is likely to lie in the range $10^{-2} \leq \lambda \leq 10^{-1}$, e.g. for the $K = 64$ model with $\boldsymbol{S}\xi$ noise (in Figure 5.8d) one exchanges sparsity ratios between 70% and 90% for normalized absolute errors between 3% and 20% in this range. Figures 5.8a and 5.8d show that at $\lambda \approx 6 \cdot 10^{-2}$ the normalized absolute errors of the $K = 16$ estimators are relatively low (about 10%) and the sparsity ratio is reasonable, around 75% (lower triangular matrix with bandwidth of approximately 4). This trade-off is less favorable for $K = 64$ and $K = 256$ for $\boldsymbol{I}\xi$ noise (Figure 5.8a), where 10% error is achieved when the sparsity is around 72% (at $\lambda \approx 2 \cdot 10^{-2}$) and 68% (at $\lambda \approx 10^{-2}$), respectively. Results are stronger for the $K = 64$ and $K = 256$ models with $\boldsymbol{S}\xi$ noise (Figure 5.8d), where sparsity ratios are reached of 76% at $\lambda \approx 2 \cdot 10^{-2}$ and 78% at $\lambda \approx 10^{-2}$, respectively, while errors are at 10%.

Unlike previous methodologies, Figures 5.8b, 5.8e, 5.8c, and 5.8f show that the CSCS estimators converge steadily to the reference eigenspectrum as the regularization λ decreases for all tested use cases. This is highly favorable as it allows the user to decide the balance between sparsity and accuracy as needed. Each of the eigenspectra shows that around $\lambda \in [10^{-2}, 10^{-1}]$ the approximated eigenspectra are already very close to the reference. For example, for the $K =$

256 with $\boldsymbol{I}\xi$ noise (in Figure 5.8c) there is a large difference in sparsity ratios for the $\lambda = 10^{-2}$ and $\lambda = 10^{-1}$ estimators, i.e. 61% and 94%. Clearly, this also means that the $\lambda = 10^{-2}$ estimator is closer to the reference eigenspectrum. How to weigh the importance of a higher sparsity ratio against being closer to the reference spectrum can be left to the user.

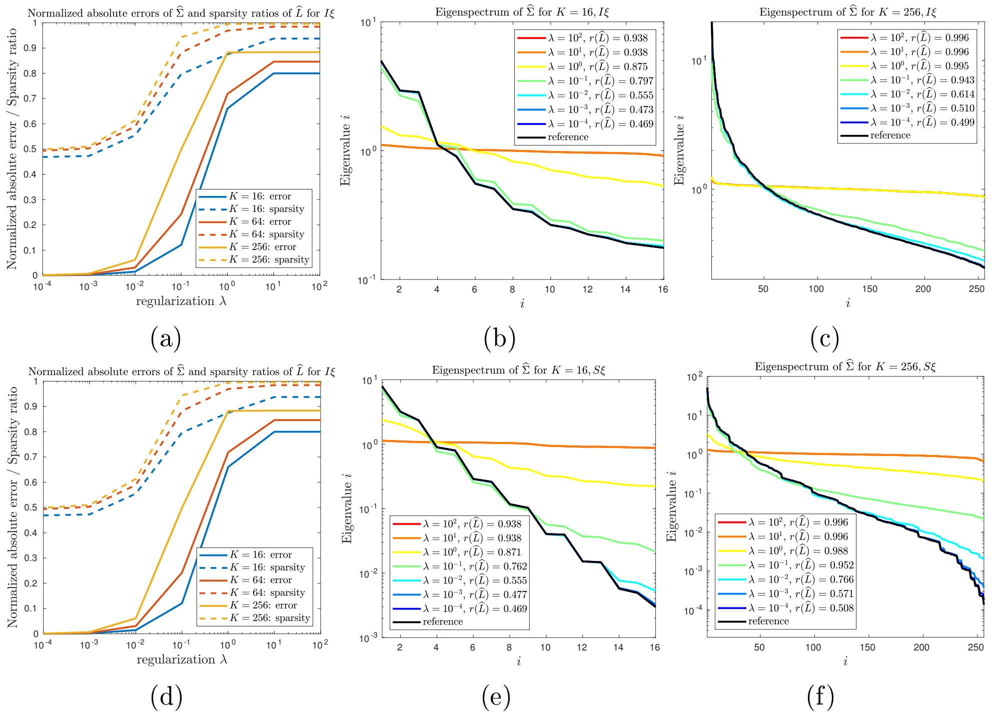


Figure 5.8: Results for CSCS (5.27) precision matrix root estimators for the $K = 16$ (5.6), $K = 64$ (5.6), and $K = 256$ (5.7) prototype models with $\boldsymbol{I}\xi$ (5.8) and $\boldsymbol{S}\xi$ (5.9) noise terms. Figure (a) shows the normalized absolute errors (5.32) and sparsity ratios (5.30) of the SPDE model with $\boldsymbol{I}\xi$ noise. Figures (b) and (c) show the eigenspectra of the covariance matrices resulting from the $\widehat{\boldsymbol{L}}$ estimators for the $\boldsymbol{I}\xi$ noise for the $K = 16$ and $K = 256$ models, respectively. Figures (d)-(f) are analog to Figures (a)-(c) for the $\boldsymbol{S}\xi$ noise term.

The results for the CSCS algorithm are both accurate and consistent both on the analytically known reference $AR(1)$ model (5.33) and on the SPDE models (5.6)-(5.7). Additionally, the results show desirable convergence behavior that enables straightforward optimization of the regularization parameter. This allows the user transparent choices in terms of deciding on the trade-off between accuracy and sparsity as the use case demands. This is an important benefit of the CSCS method over the previously discussed methodologies.

One practical drawback is that, while the code used in Khare et al. [88] allowed for the execution of these tests and some basic benchmarking, the code is, at this point, unsuitable for testing of large models ($K > 1000$). This is because the current implementation is fully sequential. The algorithm is fully parallellizable (see Section 3.3.1), but this work is yet to be done. Therefore, we leave the performance analysis of the CSCS algorithm on large use cases for future study.

5 Discussion

In this study we considered several methods to estimate sparse covariance and precision matrix roots from data. It was motivated by our aim to be able to sample from a high-dimensional correlated Gaussian distribution without placing a fully dense covariance matrix (root) in memory. The goal of this sparse representation was to conclude our previous work on stochastic parameterizations with a vector autoregressive model with exogenous variables (VARX) using only sparse coefficient matrices, see [144]. The target application for this study, numerical ocean modelling, gives rise to some constraints that were important for selecting appropriate estimation methodologies. We considered several state-of-the-art methodologies for sparse covariance and precision (i.e., inverse covariance) matrix estimation compatible with these constraints.

We tested the performance of each of the presented methodologies with so-called “oracle tests”. In these tests we used data generated by prototype stochastic partial differential equation (SPDE) models. The dimensions of the data were such that we were still able to compute the full sample covariance matrix. Having the full sample covariance matrix available can be unrealistic for practical applications, but this allowed us to evaluate and compare the performance of the different methodologies. We chose six different SPDE models varied along three axes: we varied the size of the model discretization grid, the space dimensionality, and the noise process. By varying the grid size we varied the different correlation scales between model terms. By varying the space dimensionality of the SPDE we tested the dependency between grid indexing and sparsity patterns. Finally, by varying the noise terms, one with i.i.d.

samples from a standard normal distribution and the other with i.i.d. samples from a spatially correlated Gaussian distribution with exponentially decaying covariance, we tested robustness to different covariances.

We considered two equivalent, but subtly different, methods to simulate a VARX model: one expressed in terms of the covariance root $\boldsymbol{S}$, and another expressed in terms of the precision matrix root $\boldsymbol{L}$. We considered the alternative to the typical covariance root formulation because of different interpretations of sparsity in each. In particular, an element i, j of the covariance matrix of model variable y is zero if and only if y_i and y_j are mutually independent. On the other hand, an element i, j of the precision matrix of model variable y is zero if and only if y_i and y_j are conditionally independent given all other elements in y. Hence, depending on the properties of the underlying distributions generating y, the covariance matrix might be fully dense, but the precision matrix might still be sparse, see the example in section 1.2.

For those methodologies that obtain a sparse approximation of a covariance or a precision matrix, there is an extra matrix decomposition step involved in obtaining the matrix root to evaluate the VARX model. We considered a total of three categories of methodologies to obtain a sparse matrix root: sparse covariance estimation (followed by matrix decomposition to obtain a covariance root), sparse precision matrix estimation (followed by matrix decomposition to obtain a precision matrix root), and sparse precision matrix root estimation. The latter is the only method that directly promotes sparsity in the required matrix root.

First, from the field of sparse covariance estimation we compared thresholding methods (truncating the covariance based on absolute values of its entries), banding methods (truncating the covariance based on distance between grid points), and tapering methods (truncating the covariance based on covariance functions that operate like kernel functions with finite support, determined by distance between grid points). Second, from the field of sparse precision matrix estimation we compared several variations of the graphical lasso (GL), i.e. a standard formulation that adds an ℓ_1-penalty term to the maximum likelihood equation of a Gaussian model to promote sparsity in the precision matrix. The maximum determinant matrix completion (MDMC) estimator promotes sparsity on the off-diagonal of the precision matrix and optionally imposes an a priori chosen sparsity pattern. The constrained ℓ_1-minimization for inverse matrix estimation (CLIME) and tuning-insensitive graph estimation and regression (TIGER) estimators estimate the columns of the sparse precision matrix separately. Third, from the field of sparse precision matrix root estimation we considered the convex sparse Cholesky selection (CSCS) estimator. The CSCS method decomposes an ℓ_1-penalized cost function into K separate cost functions that directly impose sparsity in the precision matrix root estimate.

We compared the above methodologies by inspecting (i) normalized absolute errors of the approximated covariance with respect to the reference covariance, (ii) sparsity ratio of the approximated matrix root, and (iii) errors in the eigenspectra of the approximated covariance compared to the reference covariance.

The thresholding and banding methods proved to be most successful in introducing sparsity in the approximated covariance, but not successful in introducing sparsity in the covariance root. While the tapering methods were more successful in introducing sparsity in the covariance root, they still showed substantial errors even with small tapering strength. None of the tested sparse covariance matrix estimation methods achieved both accuracy and sparsity in their estimators.

The MDMC estimators all showed proper convergence toward the sample precision matrix in terms of accuracy. Also, the MDMC estimators showed monotonic behavior in terms of sparsity. Moreover, the eigenspectra of the estimators converged smoothly to the reference eigenspectrum for the models with i.i.d. noise terms. However, the eigenspectra in case of the exponentially decaying noise covariance failed to converge to their reference. Additionally, the range of sampled regularization values could not be extended due to practical limitations of the original MDMC code implementation. The CLIME and TIGER estimators are generally unsatisfactory on the tested use cases. Both CLIME and TIGER methods fail on the use case with exponentially decaying noise covariance, showing errors larger than 100% for most of the tested values of their tuning parameters. Both methods are also unable to introduce significant sparsity in the precision matrix root for most tuning parameter values.

The CSCS estimators proved to be most successful across all tested criteria. The CSCS estimators showed monotonic convergence in both accuracy and eigenvalues for all use cases, allowing the user to choose the balance between accuracy and sparsity of the resulting estimators. Furthermore, the optimal tuning parameter value (depending on user preference) can be efficiently located due to the monotonic convergence. In all use cases parameter values were found that gave satisfactory results in terms of both sparsity ratio and errors. Furthermore, additional tests on data from a scalar $AR(1)$ process, for which the sparse precision matrix is known analytically, showed that the CSCS solutions converge to the reference, both in terms of accuracy and sparsity.

The current version of the CSCS code is fully sequential in its implementation. This precludes testing and evaluation the performance of the CSCS method on truly high-dimensional cases. Therefore, an obvious step for a follow-up study would be to extend the CSCS implementation to take advantage of its suitability for efficient parallelization. Another natural step is to combine the VARX model as proposed by Verheul and Crommelin [144] together with the CSCS method. This will enable us to construct a data-driven stochastic parameterization using a VARX model with sparse coefficient matrices. Such a VARX model can used to parameterize the mesoscale eddy forcing (or baroclinic feedback) in a dynamic ocean model, extending the line of our previous work in [143, 145].

Acknowledgements. We thank Professors Richard Y. Zhang, Salar Fattahi, and Somayeh Sojoudi for providing their code implementation of the maximum determinant matrix completion algorithm. We thank Professors Kshitij Khare, Sang-Yun Oh, Syed Rahman, and Bala Rajaratnam for providing their code implementation of the Convex Sparse Cholesky Selection.

Bibliography

[1] Andersen, M. S., Dahl, J., and Vandenberghe, L. (2010). Implementation of nonsymmetric interior-point methods for linear optimization over sparse matrix cones. *Mathematical Programming Computation*, 2(3-4):167–201.

[2] Andersen, M. S., Dahl, J., and Vandenberghe, L. (2013). Logarithmic barriers for sparse matrix cones. *Optimization Methods and Software*, 28(3):396–423.

[3] Arakawa, A. (1966). Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. Part I. *Journal of Computational Physics*, 1(1):119–143.

[4] Arnold, H., Moroz, I., and Palmer, T. (2013). Stochastic parametrizations and model uncertainty in the Lorenz'96 system. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1991):20110479.

[5] Asselin, R. (1972). Frequency filter for time integrations. *Monthly Weather Review*, 100(6):487–490.

[6] Avriel, M. and Wilde, D. J. (1966). Optimally proof for the symmetric fibonacci search technique. *Fibonacci Quarterly Journal*, pages 265–269.

[7] Banerjee, O., El Ghaoui, L., and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *The Journal of Machine Learning Research*, 9:485–516.

[8] Banerjee, O., Ghaoui, L. E., d'Aspremont, A., and Natsoulis, G. (2006). Convex optimization techniques for fitting sparse Gaussian graphical models. In *Proceedings of the 23rd international conference on Machine learning*, pages 89–96. ACM.

[9] Bengtsson, L., Steinheimer, M., Bechtold, P., and Geleyn, J.-F. (2013). A stochastic parametrization for deep convection using cellular automata. *Quarterly Journal of the Royal Meteorological Society*, 139(675):1533–1543.

[10] Berloff, P. (2015). Dynamically consistent parameterization of mesoscale eddies. Part I: Simple model. *Ocean Modelling*, 87:1–19.

[11] Berloff, P., Dewar, W., Kravtsov, S., and McWilliams, J. (2007a). Ocean eddy dynamics in a coupled ocean–atmosphere model. *Journal of Physical Oceanography*, 37(5):1103–1121.

[12] Berloff, P., Hogg, A. M. C., and Dewar, W. (2007b). The turbulent oscillator: A mechanism of low-frequency variability of the wind-driven ocean gyres. *Journal of Physical Oceanography*, 37(9):2363–2386.

[13] Berloff, P. S. (2005a). On dynamically consistent eddy fluxes. *Dynamics of Atmospheres and Oceans*, 38(3):123–146.

[14] Berloff, P. S. (2005b). Random-forcing model of the mesoscale oceanic eddies. *Journal of Fluid Mechanics*, 529:71–95.

[15] Berloff, P. S., McWilliams, J. C., and Bracco, A. (2002). Material transport in oceanic gyres. Part I: Phenomenology. *Journal of Physical Oceanography*, 32(3):764–796.

[16] Berner, J., Achatz, U., Batte, L., Bengtsson, L., Cámara, A. d. l., Christensen, H. M., Colangeli, M., Coleman, D. R., Crommelin, D., Dolaptchiev,

S. I., et al. (2017). Stochastic parameterization: Toward a new view of weather and climate models. *Bulletin of the American Meteorological Society*, 98(3):565–588.

[17] Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334.

[18] Bickel, P. J. and Levina, E. (2008). Covariance regularization by thresholding. *The Annals of Statistics*, 36(6):2577–2604.

[19] Bollhofer, M., Eftekhari, A., Scheidegger, S., and Schenk, O. (2019). Large-scale sparse inverse covariance matrix estimation. *SIAM Journal on Scientific Computing*, 41(1):A380–A401.

[20] Bolton, T. and Zanna, L. (2019). Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, 11(1):376–399.

[21] Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time Series Analysis: Forecasting and Control.* John Wiley & Sons.

[22] Boyd, S., Boyd, S. P., and Vandenberghe, L. (2004). *Convex Optimization.* Cambridge university press.

[23] Cai, T., Liu, W., and Luo, X. (2011). A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607.

[24] Castelo, R. and Roverato, A. (2006). A robust procedure for gaussian graphical model search from microarray data with p larger than n. *Journal of Machine Learning Research*, 7(Dec):2621–2650.

[25] Chorin, A. J. and Lu, F. (2015). Discrete approach to stochastic parametrization and dimension reduction in nonlinear dynamics. *Proceedings of the National Academy of Sciences USA*, 112(32):9804–9809.

[26] Christensen, R. (2002). *Plane Answers to Complex Questions*, volume 35. Springer.

[27] Cooper, F. C. and Zanna, L. (2015). Optimisation of an idealised ocean model, stochastic parameterisation of sub-grid eddies. *Ocean Modelling*, 88:38–53.

[28] Cox, M. (1987). Isopycnal diffusion in a z-coordinate ocean model. *Ocean modelling*, 74:1–5.

[29] Crommelin, D. (2018). Cellular automata for clouds and convection. In *Probabilistic Cellular Automata*, pages 327–339. Springer.

[30] Crommelin, D. and Vanden-Eijnden, E. (2008). Subgrid-scale parameterization with conditional Markov chains. *Journal of the Atmospheric Sciences*, 65(8):2661–2675.

[31] Cryer, J. D. and Chan, K.-S. (2008). *Time series analysis: with applications in R*, volume 2. Springer.

[32] Dahl, J., Vandenberghe, L., and Roychowdhury, V. (2008). Covariance selection for nonchordal graphs via chordal embedding. *Optimization Methods & Software*, 23(4):501–520.

[33] Danaher, P., Wang, P., and Witten, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(2):373–397.

[34] d'Aspremont, A., Banerjee, O., and El Ghaoui, L. (2008). First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and Applications*, 30(1):56–66.

[35] Dempster, A. P. (1972). Covariance selection. *Biometrics*, pages 157–175.

[36] Dijkstra, H. A. (2005). Nonlinear physical oceanography: a dynamical systems approach to the large scale ocean circulation and El Nino. *Institute for Marine and Atmospheric Research Utrecht*, 1:2.

[37] Dobra, A., Eicher, T. S., and Lenkoski, A. (2010). Modeling uncertainty in macroeconomic growth determinants using Gaussian graphical models. *Statistical Methodology*, 7(3):292–306.

[38] Dorrestijn, J., Crommelin, D., Biello, J., and Böing, S. (2013a). A data-driven multi-cloud model for stochastic parametrization of deep convection. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1991):20120374.

[39] Dorrestijn, J., Crommelin, D. T., Siebesma, A. P., and Jonker, H. J. (2013b). Stochastic parameterization of shallow cumulus convection estimated from high-resolution model data. *Theoretical and Computational Fluid Dynamics*, 27(1-2):133–148.

[40] Dorrestijn, J., Crommelin, D. T., Siebesma, A. P., Jonker, H. J., and Jakob, C. (2015). Stochastic parameterization of convective area fractions with a multicloud model inferred from observational data. *Journal of the Atmospheric Sciences*, 72(2):854–869.

[41] Dorrestijn, J., Crommelin, D. T., Siebesma, A. P., Jonker, H. J., and Selten, F. (2016). Stochastic convection parameterization with Markov chains in an intermediate-complexity GCM. *Journal of the Atmospheric Sciences*, 73(3):1367–1382.

[42] Durran, D. R. (1999). *Numerical methods for wave equations in geophysical fluid dynamics.*

[43] Eden, C., Jochum, M., and Danabasoglu, G. (2009). Effects of different closures for thickness diffusivity. *Ocean Modelling*, 26(1):47–59.

[44] Engquist, B., Li, X., Ren, W., Vanden-Eijnden, E., et al. (2007). Heterogeneous multiscale methods: a review. *Communications in Computational Physics*, 2(3):367–450.

[45] Epanechnikov, V. A. (1969). Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications*, 14(1):153–158.

[46] Fan, J., Liao, Y., and Liu, H. (2016). An overview of the estimation of large covariance and precision matrices. *The Econometrics Journal*, 19(1):C1–C32.

[47] Farneti, R. and Gent, P. R. (2011). The effects of the eddy-induced advection coefficient in a coarse-resolution coupled climate model. *Ocean Modelling*, 39(1):135–145.

[48] Fatkullin, I. and Vanden-Eijnden, E. (2004). A computational strategy for multiscale systems with applications to Lorenz 96 model. *Journal of Computational Physics*, 200(2):605–638.

[49] Fattahi, S. and Sojoudi, S. (2019). Graphical lasso and thresholding: Equivalence and closed-form solutions. *The Journal of Machine Learning Research*, 20(1):364–407.

[50] Fattahi, S., Zhang, R. Y., and Sojoudi, S. (2018). Sparse inverse covariance estimation for chordal structures. In *2018 European Control Conference (ECC)*, pages 837–844. IEEE.

[51] Fattahi, S., Zhang, R. Y., and Sojoudi, S. (2019). Linear-time algorithm for learning large-scale sparse graphical models. *IEEE access*, 7:12658–12672.

[52] Flierl, G. R. (1978). Models of vertical structure and the calibration of two-layer models. *Dynamics of Atmospheres and Oceans*, 2(4):341–381.

[53] Ford, G. and Kac, M. (1987). On the quantum Langevin equation. *Journal of Statistical Physics*, 46(5-6):803–810.

[54] Franzke, C. L., O'Kane, T. J., Berner, J., Williams, P. D., and Lucarini, V. (2015). Stochastic climate theory and modeling. *Wiley Interdiscip. Rev. Clim. Change*, 6(1):63–78.

[55] Frederiksen, J. S. and Davies, A. G. (1997). Eddy viscosity and stochastic backscatter parameterizations on the sphere for atmospheric circulation models. *Journal of the Atmospheric Sciences*, 54(20):2475–2492.

[56] Frederiksen, J. S. and Kepert, S. M. (2006). Dynamical subgrid-scale parameterizations from direct numerical simulations. *Journal of the Atmospheric Sciences*, 63(11):3006–3019.

[57] Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

[58] Furrer, R. and Bengtsson, T. (2007). Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *Journal of Multivariate Analysis*, 98(2):227–255.

[59] Furrer, R., Genton, M. G., and Nychka, D. (2006). Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3):502–523.

[60] Gao, X., Pu, D. Q., Wu, Y., and Xu, H. (2012). Tuning parameter selection for penalized likelihood estimation of gaussian graphical model. *Statistica Sinica*, pages 1123–1146.

[61] Gent, P. R. (2011). The Gent–McWilliams parameterization: 20/20 hindsight. *Ocean Modelling*, 39(1-2):2–9.

[62] Gent, P. R. and McWilliams, J. C. (1990). Isopycnal mixing in ocean circulation models. *Journal of Physical Oceanography*, 20(1):150–155.

[63] Ghil, M., Feliks, Y., and Sushama, L. (2002). Baroclinic and barotropic aspects of the wind-driven ocean circulation. *Physica D*, 167(1):1–35.

[64] Golub, G. H. and Van Loan, C. F. (2013). *Matrix Computations*. JHU press.

[65] Gottwald, G. A., Peters, K., and Davies, L. (2016). A data-driven method for the stochastic parametrisation of subgrid-scale tropical convective area fraction. *Quarterly Journal of the Royal Meteorological Society*, 142(694):349–359.

[66] Grabowski, W. W. (2001). Coupling cloud processes with the large-scale dynamics using the cloud-resolving convection parameterization (CRCP). *Journal of the Atmospheric Sciences*, 58(9):978–997.

[67] Grabowski, W. W. and Smolarkiewicz, P. K. (1999). CRCP: A cloud resolving convection parameterization for modeling the tropical convecting atmosphere. *Physica D*, 133(1):171–178.

[68] Griffies, S. M., Gnanadesikan, A., Pacanowski, R. C., Larichev, V. D., Dukowicz, J. K., and Smith, R. D. (1998). Isoneutral diffusion in a z-coordinate ocean model. *Journal of Physical Oceanography*, 28(5):805–830.

[69] Grooms, I. and Majda, A. J. (2013). Efficient stochastic superparameterization for geophysical turbulence. *Proceedings of the National Academy of Sciences USA*, 110(12):4464–4469.

[70] Grooms, I. and Majda, A. J. (2014). Stochastic superparameterization in a one-dimensional model for wave turbulence. *Communications in Mathematical Sciences*, 12(3):509–525.

[71] Hallberg, R. and Gnanadesikan, A. (2006). The role of eddies in determining the structure and response of the wind-driven Southern Hemisphere overturning: Results from the modeling eddies in the Southern Ocean (MESO) project. *Journal of Physical Oceanography*, 36(12):2232–2252.

[72] Haltiner, G. J. and Williams, R. T. (1980). *Numerical Prediction and Dynamic Meteorology*. Wiley, 2nd edition.

[73] Hamill, T. M., Whitaker, J. S., and Snyder, C. (2001). Distance-dependent filtering of background error covariance estimates in an ensemble kalman filter. *Monthly Weather Review*, 129(11):2776–2790.

[74] Hewitt, H. T., Roberts, M., Mathiot, P., Biastoch, A., Blockley, E., Chassignet, E. P., Fox-Kemper, B., Hyder, P., Marshall, D. P., Popova, E., et al. (2020). Resolving and parameterising the ocean mesoscale in earth system models. *Current Climate Change Reports*, 6(4):137–152.

[75] Honorio, J. and Jaakkola, T. S. (2013). Inverse covariance estimation for high-dimensional data in linear time and space: Spectral methods for riccati and sparse models. *arXiv preprint arXiv:1309.6838*.

[76] Horn, R. A. and Johnson, C. R. (2012). *Matrix Analysis*. Cambridge university press.

[77] Houtekamer, P. L. and Mitchell, H. L. (2001). A sequential ensemble kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 129(1):123–137.

[78] Hsieh, C.-J., Dhillon, I. S., Ravikumar, P. K., and Sustik, M. A. (2011). Sparse inverse covariance matrix estimation using quadratic approximation. In *Advances in neural information processing systems*, pages 2330–2338.

[79] Hsieh, C.-J., Sustik, M. A., Dhillon, I. S., and Ravikumar, P. (2014). Quic: quadratic approximation for sparse inverse covariance estimation. *The Journal of Machine Learning Research*, 15(1):2911–2947.

[80] Hsieh, C.-J., Sustik, M. A., Dhillon, I. S., Ravikumar, P. K., and Poldrack, R. (2013). BIG & QUIC: Sparse inverse covariance estimation for a million variables. In *Advances in neural information processing systems*, pages 3165–3173.

[81] Hua, B. and Haidvogel, D. (1986). Numerical simulations of the vertical structure of quasi-geostrophic turbulence. *Journal of the Atmospheric Sciences*, 43(23):2923–2936.

[82] Huang, J. Z., Liu, N., Pourahmadi, M., and Liu, L. (2006). Covariance matrix selection and estimation via penalised normal likelihood. *Biometrika*, 93(1):85–98.

[83] Hyndman, R. J. and Athanasopoulos, G. (2014). *Forecasting: principles and practice.* OTexts.

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

[85] Kalnay, E. (2003). *Atmospheric modeling, data assimilation and predictability.* Cambridge university press.

[86] Kevrekidis, I. G. and Samaey, G. (2009). Equation-free multiscale computation: Algorithms and applications. *Annual Review of Physical Chemistry*, 60:321–344.

[87] Khairoutdinov, M., Randall, D., and DeMott, C. (2005). Simulations of the atmospheric general circulation using a cloud-resolving model as a superparameterization of physical processes. *Journal of the Atmospheric Sciences*, 62(7):2136–2154.

[88] Khare, K., Oh, S.-Y., Rahman, S., and Rajaratnam, B. (2019). A scalable sparse cholesky based approach for learning high-dimensional covariance matrices in ordered data. *Machine Learning*, 108(12):2061–2086.

[89] Khouider, B., Biello, J., Majda, A. J., et al. (2010). A stochastic multicloud model for tropical convection. *Communications in Mathematical Sciences*, 8(1):187–216.

[90] Kiefer, J. (1953). Sequential minimax search for a maximum. *Proceedings of the American Mathematical Society*, 4(3):502–506.

[91] Kitsios, V., Frederiksen, J., and Zidikheri, M. (2013). Scaling laws for parameterisations of subgrid eddy–eddy interactions in simulations of oceanic circulations. *Ocean Modelling*, 68:88–105.

[92] Kondrashov, D. and Berloff, P. (2015). Stochastic modeling of decadal variability in ocean gyres. *Geophysical Research Letters*, 42(5):1543–1553.

[93] Ley, E. and Steel, M. F. (2006). *Jointness in Bayesian variable selection with applications to growth regression.* The World Bank.

[94] Li, L. and Toh, K.-C. (2010). An inexact interior point method for l_1-regularized sparse covariance selection. *Mathematical Programming Computation*, 2(3-4):291–315.

[95] Lilly, D. K. (1965). On the computational stability of numerical solutions of time-dependent non-linear geophysical fluid dynamics problems. *Monthly Weather Review*, 93(1):11–25.

[96] Lindgren, F., Rue, H., and Lindström, J. (2011). An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498.

[97] Lindsey, J. K. (2004). *Statistical analysis of stochastic processes in time*, volume 14. Cambridge University Press.

[98] Liu, H., Wang, L., et al. (2017). Tiger: A tuning-insensitive approach for optimally estimating gaussian graphical models. *Electronic Journal of Statistics*, 11(1):241–294.

[99] Lorenz, E. N. (1996). Predictability: A problem partly solved. In *Proc. Seminar on predictability*, volume 1.

[100] Lütkepohl, H. (2005). *New introduction to multiple time series analysis.* Springer Science & Business Media.

[101] Lütkepohl, H. (2006). Forecasting with VARMA models. *Handbook of economic forecasting*, 1:287–325.

[102] Majda, A. J. and Grote, M. J. (2009). Mathematical test models for superparametrization in anisotropic turbulence. *Proceedings of the National Academy of Sciences*, 106(14):5470–5474.

[103] Majda, A. J. and Khouider, B. (2002). Stochastic and mesoscopic models for tropical convection. *Proceedings of the National Academy of Sciences*, 99(3):1123–1128.

[104] Majda, A. J., Timofeyev, I., and Vanden-Eijnden, E. (2003). Systematic strategies for stochastic mode reduction in climate. *Journal of the Atmospheric Sciences*, 60(14):1705–1722.

[105] Mazumder, R. and Hastie, T. (2012a). Exact covariance thresholding into connected components for large-scale graphical lasso. *The Journal of Machine Learning Research*, 13(1):781–794.

[106] Mazumder, R. and Hastie, T. (2012b). The graphical lasso: New insights and alternatives. *Electronic journal of statistics*, 6:2125.

[107] McCullagh, P. and Nelder, J. A. (2019). *Generalized linear models*. Routledge.

[108] McWilliams, J. C. (2008). The nature and consequences of oceanic eddies. *Ocean Modeling in an Eddying Regime*, pages 5–15.

[109] Nimsaila, K. and Timofeyev, I. (2010). Markov chain stochastic parametrizations of essential variables. *Multiscale Modeling & Simulation*, 8(5):2079–2096.

[110] Olbers, D., Willebrand, J., and Eden, C. (2012). *Ocean dynamics*. Springer Science & Business Media.

[111] Palmer, T. N. (2001). A nonlinear dynamical perspective on model error: A proposal for non-local stochastic-dynamic parametrization in weather and climate prediction models. *Quarterly Journal of the Royal Meteorological Society*, 127(572):279–304.

[112] Paulsen, V. I., Power, S. C., and Smith, R. R. (1989). Schur products and matrix completions. *Journal of Functional Analysis*, 85(1):151–178.

[113] Pavliotis, G. and Stuart, A. (2007). Parameter estimation for multiscale diffusions. *Journal of Statistical Physics*, 127(4):741–781.

[114] Pavliotis, G. A. (2016). *Stochastic processes and applications.* Springer.

[115] Pelupessy, I., Van Werkhoven, B., Van Elteren, A., Viebahn, J., Candy, A., Zwart, S. P., and Dijkstra, H. (2017). The oceanographic multipurpose software environment (OMUSE v1. 0). *Geoscientific Model Development*, 10(8):3167.

[116] Plant, R. and Craig, G. C. (2008). A stochastic parameterization for deep convection based on equilibrium statistics. *Journal of the Atmospheric Sciences*, 65(1):87–105.

[117] Porta Mana, P. and Zanna, L. (2014). Toward a stochastic parameterization of ocean mesoscale eddies. *Ocean Modelling*, 79:1–20.

[118] Ravikumar, P., Wainwright, M. J., Raskutti, G., Yu, B., et al. (2011). High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980.

[119] Robert, A. J. (1966). The integration of a low order spectral form of the primitive meteorological equations. *Journal of the Meteorological Society of Japan. Ser. II*, 44:237–245.

[120] Rothman, A. J., Bickel, P. J., Levina, E., Zhu, J., et al. (2008). Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, 2:494–515.

[121] Rothman, A. J., Levina, E., and Zhu, J. (2010). A new approach to cholesky-based covariance regularization in high dimensions. *Biometrika*, 97(3):539–550.

[122] Rue, H. and Held, L. (2005). *Gaussian Markov random fields: theory and applications.* CRC press.

[123] Ryzhov, E., Kondrashov, D., Agarwal, N., McWilliams, J., and Berloff, P. (2020). On data-driven induction of the low-frequency variability in a coarse-resolution ocean model. *Ocean Modelling*, 153:101664.

[124] Sainath, T. N., Ramabhadran, B., Picheny, M., Nahamoo, D., and Kanevsky, D. (2011). Exemplar-based sparse representation features: From TIMIT to LVCSR. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(8):2598–2613.

[125] Saon, G. and Chien, J.-T. (2011). Bayesian sensing hidden Markov models for speech recognition. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5056–5059. IEEE.

[126] Scheinberg, K., Ma, S., and Goldfarb, D. (2010). Sparse inverse covariance selection via alternating linearization methods. In *Advances in neural information processing systems*, pages 2101–2109.

[127] Scheinberg, K. and Rish, I. (2009). SINCO-a greedy coordinate ascent method for sparse inverse covariance selection problem. *preprint*.

[128] Scheinberg, K. and Rish, I. (2010). Learning sparse Gaussian Markov networks using a greedy coordinate ascent approach. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 196–212. Springer.

[129] Schmidt, M., Berg, E., Friedlander, M., and Murphy, K. (2009). Optimizing costly functions with simple constraints: A limited-memory projected quasi-newton algorithm. In *Artificial Intelligence and Statistics*, pages 456–463.

[130] Schur, J. (1911). *Bemerkungen zur Theorie der beschränkten Bilinearformen mit unendlich vielen Veränderlichen.* Walter de Gruyter, Berlin/New York Berlin, New York.

[131] Shevchenko, I. and Berloff, P. (2017). On the roles of baroclinic modes in eddy-resolving midlatitude ocean dynamics. *Ocean Modelling*, 111:55–65.

[132] Shojaie, A. and Michailidis, G. (2010). Penalized likelihood methods for estimation of sparse high-dimensional directed acyclic graphs. *Biometrika*, 97(3):519–538.

[133] Shutts, G. (2005). A kinetic energy backscatter algorithm for use in ensemble prediction systems. *Quarterly Journal of the Royal Meteorological Society*, 131(612):3079–3102.

[134] Smith, M. and Kohn, R. (2002). Parsimonious covariance matrix estimation for longitudinal data. *Journal of the American Statistical Association*, 97(460):1141–1153.

[135] Sojoudi, S. (2016). Equivalence of graphical lasso and thresholding for sparse graphs. *The Journal of Machine Learning Research*, 17(1):3943–3963.

[136] Storch, J.-S. v., Eden, C., Fast, I., Haak, H., Hernández-Deckers, D., Maier-Reimer, E., Marotzke, J., and Stammer, D. (2012). An estimate of the Lorenz energy cycle for the world ocean based on the STORM/NCEP simulation. *Journal of Physical Oceanography*, 42(12):2185–2205.

[137] Strutz, T. (2010). *Data fitting and uncertainty: A practical introduction to weighted least squares and beyond.* Vieweg and Teubner.

[138] Stuart, A. and Warren, J. (1999). Analysis and experiments for a computational model of a heat bath. *Journal of Statistical Physics*, 97(3-4):687–723.

[139] Tang, C. Y. and Chen, S. X. (2009). Parameter estimation and bias correction for diffusion processes. *Journal of Econometrics*, 149(1):65–81.

[140] Thomas, J. W. (2013). *Numerical partial differential equations: finite difference methods*, volume 22. Springer Science & Business Media.

[141] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

[142] Vallis, G. K. (2006). *Atmospheric and oceanic fluid dynamics: fundamentals and large-scale circulation.* Cambridge University Press.

[143] Verheul, N. and Crommelin, D. (2016). Data-driven stochastic representations of unresolved features in multiscale models. *Communications in Mathematical Sciences*, 14(5):1213–1236.

[144] Verheul, N. and Crommelin, D. (2021). Stochastic parametrization with VARX processes. *Communications in Applied Mathematics and Computational Science*, 16(1):33–57.

[145] Verheul, N., Viebahn, J., and Crommelin, D. (2017). Covariate-based stochastic parameterization of baroclinic ocean eddies. *Mathematics of Climate and Weather Forecasting*, 3(1):90–117.

[146] Viebahn, J., Crommelin, D., and Dijkstra, H. (2019). Toward a turbulence closure based on energy modes. *Journal of Physical Oceanography*, 49(4):1075–1097.

[147] Viebahn, J. and Eden, C. (2010). Towards the impact of eddies on the response of the Southern Ocean to climate change. *Ocean Modelling*, 34(3):150–165.

[148] Wang, H., Banerjee, A., Hsieh, C.-J., Ravikumar, P., and Dhillon, I. S. (2013). Large scale distributed sparse precision estimation. In *NIPS*, volume 13, pages 584–592.

[149] Wendland, H. (1995). Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics*, 4(1):389–396.

[150] Whittle, P. (1954). On stationary processes in the plane. *Biometrika*, pages 434–449.

[151] Whittle, P. (1963). Stochastic-processes in several dimensions. *Bulletin of the International Statistical Institute*, 40(2):974–994.

[152] Wilks, D. (2012). 'Superparameterization' and statistical emulation in the Lorenz '96 system. *Quarterly Journal of the Royal Meteorological Society*, 138(666):1379–1387.

[153] Wilks, D. S. (2005). Effects of stochastic parametrizations in the Lorenz'96 system. *Quarterly Journal of the Royal Meteorological Society*, 131(606):389–407.

[154] Wille, A. and Bühlmann, P. (2006). Low-order conditional independence graphs for inferring genetic networks. *Statistical Applications in Genetics and Molecular Biology*, 5(1).

[155] Williams, P. D. (2009). A proposed modification to the Robert-Asselin time filter. *Monthly Weather Review*, 137(8):2538–2546.

[156] Wu, W. B. and Pourahmadi, M. (2003). Nonparametric estimation of large covariance matrices of longitudinal data. *Biometrika*, 90(4):831–844.

[157] Young, J. A. (1968). Comparative properties of some time differencing schemes for linear and nonlinear oscillations. *Monthly Weather Review*, 96(6):357–364.

[158] Yu, G. and Bien, J. (2017). Learning local dependence in ordered data. *The Journal of Machine Learning Research*, 18(1):1354–1413.

[159] Yuan, M. and Lin, Y. (2007). Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35.

[160] Zanna, L., Mana, P. P., Anstey, J., David, T., and Bolton, T. (2017). Scale-aware deterministic and stochastic parametrizations of eddy-mean flow interaction. *Ocean Modelling*, 111:66–80.

[161] Zhang, R., Fattahi, S., and Sojoudi, S. (2018). Large-scale sparse inverse covariance estimation via thresholding and max-det matrix completion. In *International Conference on Machine Learning*, pages 5766–5775. PMLR.

[162] Zwanzig, R. (1973). Nonlinear generalized Langevin equations. *Journal of Statistical Physics*, 9(3):215–220.

List of Publications

- **Verheul, N.**, and Crommelin, D. (2016). *Data-driven stochastic representations of unresolved features in multiscale models.* Communications in Mathematical Sciences, 14(5), 1213-1236. DOI: `https://doi.org/10.4310/CMS.2016.v14.n5.a2`

  Chapter 2 is based on this research article. The research work, writing of the article, and numerical simulations were carried out by N. Verheul under the supervision of D. Crommelin.

- **Verheul, N.**, Viebahn, J., and Crommelin, D. (2017). *Covariate-based stochastic parameterization of baroclinic ocean eddies.* Mathematics of Climate and Weather Forecasting, 3(1), 90-117. DOI: `https://doi.org/10.1515/mcwf-2017-0005`

  Chapter 3 is based on this research article. The research work, writing of the article, and numerical simulations were carried out by N. Verheul in collaboration with J. Viebahn under the supervision of D. Crommelin.

- **Verheul, N.**, and Crommelin, D. (2021). *Stochastic parametrization with VARX processes.* Communications in Applied Mathematics and Computational Science, 16(1), 33-57. DOI: `https://doi.org/10.2140/camcos.2021.16.33`

  Chapter 4 is based on this research article. The research work, writing of the article, and numerical simulations were carried out by N. Verheul under the supervision of D. Crommelin.

- **Verheul, N.**, and Crommelin, D. (2022). *A comparison of methods for estimating sparse covariance and precision matrix roots.* To be submitted.

Chapter 5 is based on this research article. The research work, writing of the article, and numerical simulations were carried out by N. Verheul under the supervision of D. Crommelin.