

Model Reduced Variational Data Assimilation for Shallow Water Flow Models

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K. C. A. M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 31 januari 2011 om 10.00 uur

door

Muhammad Umer ALTAF,
Master of Science in Mathematics, Quaid-e-Azam University, Pakistan

geboren te Rawalpindi, Pakistan.

Dit proefschrift is goedgekeurd door de promotor:
Prof. dr. ir. A. W. Heemink

Copromotor:
Dr. ir. M. Verlaan

Samenstelling promotiecommissie:

Rector Magnificus	voorzitter
Prof. dr. ir. A. W. Heemink,	Technische Universiteit Delft, promotor
Dr. ir. M. Verlaan,	Technische Universiteit Delft, copromotor
Prof. dr. ir. A. Mynett,	UNESCO-IHE/Technische Universiteit Delft
Prof. dr. ir. C. Vuik,	Technische Universiteit Delft
Prof. dr. ir. E. Deleersnijder,	Universite Catholique de Louvain, Belgium
Prof. dr. ir. J.H. van Schuppen,	Technische Universiteit Delft
Dr. H. Madsen,	DHI Group, Denmark

This thesis has been completed in partial fulfillment of the requirements of Delft University of Technology (Delft, The Netherlands) for the award of the Ph.D. degree. The research described in this thesis was carried out in the Delft Institute of Applied Mathematics, Delft University of Technology.

Published and distributed by: Muhammad Umer ALTAF
E-mail: umaraltaf74@hotmail.com

ISBN # 978-90-8570-718-9

Copyright © 2010 by Muhammad Umer ALTAF

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

Printed in The Netherlands: by Wöhmann Print Service

Contents

1	Introduction	1
1.1	Storm surge forecasting	2
1.2	Data assimilation and model calibration	5
1.3	Model reduction	6
1.3.1	Proper Orthogonal Decomposition	7
1.4	Motivation and overview	8
2	The Dutch Continental Shelf Model	11
2.1	The model equations	11
2.2	Numerical approximation	14
2.2.1	State space model	17
2.3	Adjoint method	17
2.3.1	Procedural flow for the classical adjoint method	20
2.3.2	The LBFGS method	20
2.4	An overview on DCSM calibration	21
2.5	Operational forecasts using DCSM	22
3	POD and BPOD within variational data assimilation	25
3.1	Introduction	25
3.2	Model reduction	26
3.2.1	Proper Orthogonal Decomposition	27
3.2.2	Balanced Truncation	28
3.2.3	Balanced POD	29
3.3	Inverse modeling using reduced models	31
3.3.1	Linearization and reduced model formulation	32
3.3.2	Collection of the snapshots and the reduced basis	34
3.3.3	Approximate objective function and its adjoint	34
3.4	Numerical experiments	35
3.4.1	The Model	35
3.4.2	Experiment 1	36
3.4.3	Experiment 2	42

3.5	Conclusions	45
4	Inverse shallow water flow modeling using model reduction	47
4.1	Introduction	47
4.2	Inverse modeling	49
4.3	Reduced order modeling	50
4.3.1	Proper Orthogonal Decomposition	50
4.4	Inverse modeling using reduced models	51
4.4.1	Linearization and reduced Basis	51
4.4.2	Reduced model formulation	52
4.4.3	Approximate objective function and its adjoint	52
4.4.4	Procedural flow with reduced model	53
4.4.5	Computational cost	54
4.5	Application: The Dutch Continental Shelf Model	54
4.5.1	Estimation of depth	56
4.6	Conclusions	64
5	Case Study: Identification of uncertain parameters in a large scale tidal model	67
5.1	Introduction	67
5.2	The new Dutch continental shelf model	70
5.2.1	Model computational grid of the new model	70
5.2.2	Model bathymetry and bottom roughness	70
5.2.3	Model boundary conditions	71
5.2.4	Initial conditions and computational time	73
5.3	Inverse modeling using POD	73
5.3.1	Linearization and reduced model formulation	74
5.3.2	Collection of the snapshots and POD basis	75
5.3.3	Approximate objective function and its adjoint	75
5.3.4	Workflow with POD algorithm	76
5.3.5	Convergence criterion for inner and outer iterations	77
5.3.6	Computational efficiency of the algorithm	77
5.4	Parameter identification for the new DCSM	78
5.4.1	Measurement data	78
5.4.2	The calibration and validation datasets and time period	79
5.4.3	Time and frequency domain analysis	80
5.4.4	Ensemble generation	82
5.4.5	Scaling	82
5.4.6	Model calibration	82
5.4.7	Discussion on results	92
5.5	Conclusions	99
6	Parameter Estimation using Simultaneous Perturbation	101
6.1	Introduction	101
6.2	Parameter estimation using SPSA	103
6.2.1	Choice of a_l and c_l	104

6.2.2	Average stochastic gradient	104
6.3	Numerical experiments	105
6.3.1	Experiment 1	105
6.3.2	Experiment 2	110
6.4	Conclusions	114
7	Conclusions	117
Appendices		
A	The Reduced Adjoint State	121
	Bibliography	123
	Summary	131
	Samenvatting	133
	Acknowledgements	135
	Curriculum Vitae	137

List of Tables

2.1	Warning and alarm levels used by SVSD (level relative to normal Amsterdam level (NAP))	23
3.1	The results of the estimation of two parameters γ_1 and γ_2 . Here α and β are the number of inner and outer iterations respectively.	38
3.2	Results of the estimation of two parameters γ_1 and γ_2 with the reduced model that captured 99.99% of the relative energy.	39
3.3	Results of the estimation of four parameters $\gamma_1, \gamma_2, \gamma_3$ and γ_4 with the POD based estimation procedure.	41
3.4	RRMSE in the forward simulation with different model reduction methods	44
5.1	RMSE results for the minimization process with the 97% relative energy after 1 st and 2 nd outer iteration (Experiment 1)	85
5.2	RMSE results for the minimization process with the 90% relative energy after 1 st and 2 nd outer iteration (Experiment 1)	86
5.3	Results after 2 nd outer iteration obtained by generating new ensemble and using the same ensemble as in 1 st outer iteration (Experiment 1).	87
5.4	RMSE results for the minimization process with the 90% relative energy after 3 rd and 4 th outer iterations (Experiment 2)	89
5.5	RMSE results for the minimization process with the 97% relative energy after 5 th outer iteration (Experiment 3)	91
5.6	Computational costs of the calibration experiments after each outer iteration β of the minimization process	96
6.1	Comparison of estimated parameters to true parameters for the twin experiment	109
6.2	RMSE results for the minimization process after 5 th , 10 th , 15 th and 20 th iterations	109

List of Figures

1.1	Eastern Scheldt barrier at North Sea during storm	1
1.2	The Eastern Scheldt storm surge barrier, or Oosterschelde. (photo by Rob Broek/iStockPhoto.com)	3
1.3	Storm surge barrier at the New Waterwey. (photo taken from Microsoft Encarta Online Encyclopaedia 2008)	4
1.4	The inverse structure of the data assimilation problems	5
2.1	Area of the operational DCSM.	12
2.2	The computational grid	15
2.3	Block diagram of calibration process using adjoint method [2]	21
3.1	Flowcharts of the parameter estimation procedure with the classical adjoint method (left) and the reduced model approaches (middle (POD) and right (BPOD))	35
3.2	Velocity Field	36
3.3	The 2D pollution model domain; The locations of the selected data points (circles) and straight line divides the model domain in two regions one for each estimated parameter	37
3.4	The POD modes captured energy for an ensemble of 100 snapshot vectors	37
3.5	The value of the objective function J at successive outer iterations for reduced models with different relative energy attained	39
3.6	The locations of the selected data points (circles) and straight line divides the model domain in two regions one for each estimated parameter	40
3.7	The locations of the selected data points (circles) and the four subdomains used used in case 3 to estimate four parameters	40
3.8	The POD modes captured energy for an ensemble of 200 snapshots in the outer iterations a) β_1 b) β_2	41
3.9	The 2D pollution model domain; The locations of the selected data points (circles) and straight line divides the model domain in two regions one for each estimated parameter	42

3.10	The successive outer iterations β of the objective function (J) for the POD and the BPOD based reduced order models of order \mathfrak{R}^{10+2} . . .	43
3.11	The successive outer iterations β of the objective function (J) for the POD and the outer projection based reduced order models of order \mathfrak{R}^{10+2}	44
4.1	DCSM area and assimilation stations: 1. N51, 2. Southend, 3. Innerdowsing, 4. Oostende, 5. H.v.Holland, 6. Den Helder, 7. N4	55
4.2	(a) The domain Ω (dashed rectangle) of DCSM; (b) The subdomains $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ and Ω_7	57
4.3	The energy captured by POD modes for an ensemble of 210 snapshots of the water level h , velocities u and v	58
4.4	Value of the objective function J versus the number of outer iterations β	59
4.5	Convergence of the depth parameters that coincide along the UK, Dutch and Belgium coasts	60
4.6	Water level timeseries for the period from 16 December 1997 00:00 - 18 December 1997 24:00 for simulated data (truth), deterministic model without assimilation (background), deterministic model with data assimilation after four outer iterations (reduced estimation) at a) Den Helder; b) Hoek van Holland	61
4.7	RMSE for water level observations at a) assimilation stations and b) validation stations	62
4.8	Value of the objective function J versus the number of outer iterations β for the calibration over 15 days	63
4.9	RMSE for water level observations at assimilation stations for calibration over 15 days	64
4.10	Water level timeseries at Hoek van Holland for the period from 24 December 1997 00:00 - 24 December 1997 24:00 for simulated data (truth), the initial setting of the calibration parameters (background) and deterministic model after calibration (reduced estimation)	65
5.1	Newly developed hydrodynamic DCSM area. The dashed line represents the area of the operational DCSM extent	71
5.2	DCSM model bathymetry in meters. The bathymetry greater than 2000m is shown as 2000m	72
5.3	DCSM area with stations included in the model calibration	79
5.4	DCSM area with stations included in the validation dataset	80
5.5	Stations along the Dutch coast included in the Dutch dataset for the validation period	81
5.6	The four subdomains Ω_k of the DCSM used in Experiments 1 and 3 .	83
5.7	The energy captured by POD modes for an ensemble of 132 snapshots of the water level h , velocities u and v (Experiment 1)	84
5.8	The reduction of the value of the objective function J at successive outer iterations β_1 and β_2 (Experiment 1)	85
5.9	The values of a) $\hat{J}_\alpha / \hat{J}_0$ and b) convergence criterion μ at successive inner iterations α for the outer iterations β_1 and β_2 (Experiment 1) . . .	86

5.10	The energy captured by POD modes for a new ensemble of 132 snapshots of the water level h , velocities u and v in the outer iteration β_2 (Experiment 1)	87
5.11	The 12 subdomains Ω_k of the DCSM used in Experiment 2	88
5.12	The energy captured by POD modes for an ensemble of 396 snapshots of the water level h , velocities u and v in the 3^{rd} outer iteration β_3 (Experiment 2)	89
5.13	The reduction of value of the objective function J at successive outer iterations β_3 and β_4 (Experiment 2)	90
5.14	The POD modes captured energy for an ensemble of 132 snapshots of the water level h , velocities u and v in the outer iteration β_5 (Experiment 3)	91
5.15	The values of a) $\hat{J}_\alpha / \hat{J}_0$ and b) convergence criterion μ at successive inner iterations α for the outer iteration β_5 (Experiment 3)	92
5.16	RMSE for the calibration and validation datasets after each outer iterations β of the minimization process	93
5.17	The mean value of the total depth adaptation after each outer iteration	94
5.18	Water level timeseries for the period from 15 January 2007 00:00 - 16 January 2007 24:00 obtained from measurement data (observations), forecast using deterministic model without data assimilation and forecast with data assimilation (calibrated) after β_4 respectively at the two tide gauge stations a) Delfzijl and b) Vlissingen	95
5.19	RMS of water level at the selected tide gauge stations along the Dutch coast with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1, β_3 and β_5 respectively	96
5.20	Model performance for a) Hoek van Holland; b) Dover; c) Wick expressed as vector differences of the 7 important constituents with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1, β_3 and β_5 respectively	97
5.21	Model performance for the datasets a) Calibration; b) Validation; c) Dutch expressed as RMS(VD) for the 7 important constituents with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1, β_3 and β_5 respectively	98
6.1	DCSM area with calibration stations: 1. N51, 2. Southend, 3. Innerdowsing, 4. Oostende, 5. H.v.Holland, 6. Den Helder, 7. N4	106
6.2	(a) Shows the domain Ω (dashed rectangle), of DCSM. (b) Shows the subdomains $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6$ and Ω_7	107
6.3	Successive iterations β of the minimization process	108
6.4	Water level timeseries for the period from 18 December 1997 00:00 - 18 December 1997 24:00 obtained from truth model, deterministic model with initial values of the estimated parameters and deterministic model after calibration respectively at the two tide gauge stations a) Den Helder and b) Southend	110
6.5	Newly developed hydrodynamic DCSM area. The dashed line represents the area of the operational DCSM extent	111

6.6 The 12 subdomains Ω_k of the DCSM used in Experiment 2 112

6.7 DCSM area with stations included in the model calibration 113

6.8 Successive iterations β of the minimization process 114

Chapter 1

Introduction

In the Netherlands, large areas of the land along the North Sea lie below mean Sea level. The North Sea is characterized by strong tides and storm surges which have strong impact on Dutch coast. Another important aspect is that the coast is open. It has many inlets since the country serves as last passage way of the three main river systems (Rhine, Meuse and Scheldt) before they enter into the North Sea.

Considering the geophysical conditions and the fact that large areas of the land lie below mean sea level, it is very important to protect the country against flooding. On the other hand, these waters are used for drinking water, fishing, transport and for similar kind of activities. In predicting the effects of the various activities of interests along the Dutch coast and to use the rivers, estuaries and sea in a safe and economical way, numerical flow modeling plays an important role. It is also an important part of successful water management.



Figure 1.1: Eastern Scheldt barrier at North Sea during storm

The numerical models are conceptual approximations that describe physical systems by means of mathematical equations. So the applicability of the numerical models is actually dependent on how closely these mathematical equations approximate the physical phenomenon. Although in the last few years the computational resources have grown immensely but the approximation scale is always related to the available computing power.

For the last few decades computers have been fast enough to model tidal flow using large numerical flow models. The predictions made by these numerical flow models always contain errors due to the numerical approximation and the fact that a part of the inputs for these models is not known accurately (e.g. true depth, bottom friction, open boundaries, etc). The model results can be improved by identifying these uncertain parameters using real observations. One method that assimilates observations data into a model simulation to estimate uncertain parameters is data assimilation. The main objective of the research presented in this thesis is to examine and develop efficient methods with low computational costs to identify the uncertain parameters in the numerical flow models.

This chapter serves as an introduction to the work presented in the thesis. Section 1.1 briefly explains the storm surge forecasting system in the Netherlands. Section 1.2 covers the basic concepts of data assimilation. Section 1.3 highlights the importance of using model reduction and more specifically Proper Orthogonal Decomposition. Section 1.4 gives a description of the motivation of the work presented as well as an overview of the thesis.

1.1 Storm surge forecasting

In the Netherlands there is a long tradition of building dikes and dams to protect the country against floods. When a storm develops and the wind stirs up the waves, the Dutch coastal area can become flooded. The last time this happened on a large scale was around fifty five years ago. On 1st February 1953, the Dutch coast was hit by a very strong storm surge that inundated large areas in the western part of the country. The damage was enormous and nearly two thousand people died. More than 150,000 hectares of land were flooded and tens of thousands of live stocks were killed. The damage to buildings, dikes, dams and other infrastructure was huge ([36]).

In order to prevent such a catastrophe in the future the Dutch Government introduced a plan that aimed at reinforcing the water defences and shortening the coastline by closing estuaries with large dams. This plan became known as the Delta Plan [28].

This huge project aimed at protecting the coast in a manner that reduced the risk of a disaster such as the 1953 flood to a probability of one in 10^4 years for the most densely populated areas. However, the Delta Plan was not just limited to the coastal areas but stretched along the rivers all the way up to the German border. The key recommendation of the Delta Plan was the full closure of Eastern Scheldt with a regular dam to create a closed coastline (www.eh-resources.org/floods.html).

The reinforcement of dikes in the river area of the Netherlands began in 1960's. Soon this kind of river dike reinforcement attracted heavy criticism from the local population. Concerns regarding economical and ecological issues were raised by dif-

ferent social groups. People were concerned that the unique salt water environment of the Eastern Scheldt would cease to exist. Specifically, not only the environment, but also the fishing industry would suffer from a dam.

In 1976, the Dutch government agreed to an alternative plan that the unique freshwater environment and the favourable fishery conditions would be maintained. Thus, instead of building a closed dam, a move able storm surge barrier was built. Sixty two openings, each forty metres wide, were installed to allow as much salt water through as possible. These openings are closed when the water level is higher than a safety threshold to protect the land from the water. The New Waterway movable storm surge



Figure 1.2: The Eastern Scheldt storm surge barrier, or Oosterschelde. (photo by Rob Broek/iStockPhoto.com)

barrier was the final element of the Delta project. This storm surge barrier was the largest hydraulic project constructed in the Netherlands. It is nearly as long as the Eiffel tower in Paris and weighs four times as much. This storm surge barrier together with dikes reinforcement protects around 1 million people in the province of South Holland from severe storm surges in the North Sea.

One of the major reasons for the construction of a movable barrier at New Waterway is the enormous economical activity at the Rotterdam port. This barrier connects the North Sea with this port. The port of Rotterdam is directly situated on the North Sea and is the biggest seaport in Europe. The port serves as the gateway to European market and has an annual throughput of more than 400 million tonnes of goods. The port and surrounding industrial area stretch over a length of 40 kilometres and cover around 10,000 hectares. Considering the economical activity at the port, the movable storm surge barrier should be closed only in extreme situation.

The above discussion demonstrates the importance of movable storm surge barriers to fulfil the safety, economical and ecological demands. In addition to better defence against flooding from the North Sea, the warning system was also improved. It was decided that in the case of severe storms the dikes should be staffed in time to prevent them from breaching. The Dutch storm surge warning service (SVSD) in

close cooperation with Royal Netherlands Meteorological Institute (KNMI) is responsible for the water level predictions. Accurate water level forecasts at least six hours ahead are required for timely closure of the movable storm surge barriers in Eastern Scheldt and New Waterway. Besides being required for the closure of these movable barriers, water level predictions are also required for raising alarms in extreme situations. These alarms are used to ensure that the people can have sufficient time to evacuate from the area in case of flood prediction.



Figure 1.3: Storm surge barrier at the New Waterway. (photo taken from Microsoft Encarta Online Encyclopaedia 2008)

Since mid 1980's these forecasts are based on numerical hydrodynamic model called the Dutch Continental Shelf Model (DCSM). The model describes how the water level and velocities are related and evolve in time as a response to the wind forcing exerted on the sea surface as well as to the tidal waves coming from the Atlantic Ocean. A large scale numerical model such as DCSM usually suffers from inaccurately known parameters or boundary conditions. Contrary to the numerical model, which covers the whole area and time interval of interest, measurements (observations) are usually sparse. Typically, there are only few points in the model area where the measurements are available. The measurements are often quite accurate. Moreover measurements are expensive and several quantities cannot be measured at all.

Numerical flow models and observations are usually applied separately to obtain information about the water levels and currents. However, it seems logical to tune the model by adapting some of the uncertain model parameters to obtain a better fit with available measurements. One method to assimilate measurements data into a model simulation to get a better estimate of the uncertain model parameters is data assimilation which is main topic of the following chapters of this thesis

1.2 Data assimilation and model calibration

Data assimilation (DA) is a method in which the observations of the state of a system are combined with the results from numerical model to produce accurate estimates of all the current (and future) state variables of the system. A data assimilation system consists of three components: a set of observations, a dynamical model, and a data assimilation scheme.

The central concept of the data assimilation is the concept of errors, error estimation and error modeling. The observations have errors arising from various sources: e.g. instrumental noise and the representativeness errors. All dynamical models are imperfect with errors arising from: the approximate physics (or biology or chemistry), which parametrizes the interaction of the state variables and the discretization of continuum dynamics into a numerical model. An aspect common for all the data assimilation schemes is that the quantitative basis of the assimilation is formed by the relative uncertainties of the dynamics and observations. Thus, the new estimate does not degrade the reliable information of the observational data but rather enhances that information content.

The most well-known application of DA is in weather forecasting problems in which it was applied in real life for the first time in 1950's and 1960's to improve the weather forecasts. A good description of the development of data assimilation in meteorology can be found in [27]. The method has already proven to be useful in other fields of application like tidal models [42], oceanography [30], nonlinear shallow-water storm surge models [96] and atmospheric chemistry and transport modeling (e.g. [29], [76]). Among all the data assimilation methods, four dimensional variational data assimilation (4DVAR) called as adjoint method is the one of the most effective and powerful approaches. The method has an advantage of directly assimilating all the available observations distributed in time and space into the numerical model while maintaining dynamical and physical consistency with the model [83].

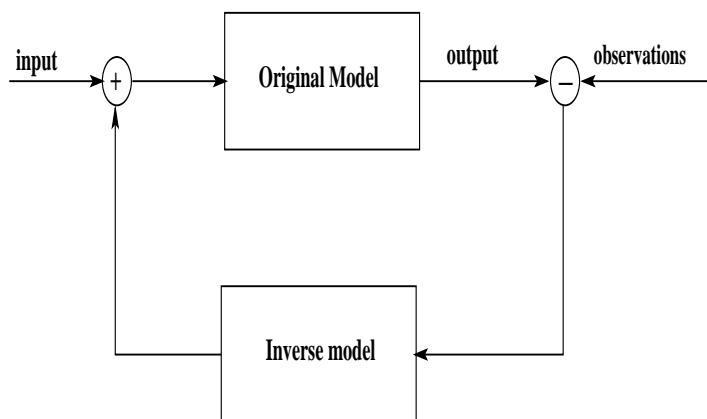


Figure 1.4: The inverse structure of the data assimilation problems

One possible application of data assimilation is model calibration, which is also called parameter estimation (inverse modeling). Physical models usually contain one or more empirical formulae which involve some constant values that must be tuned for a good performance of the model. Using the constant parameter values, the model can be used to compute simulated observations. If the parameter values in the model are incorrect, the simulated observations will not be equal to the real observations. This provides the means for the estimation of parameters. Originally it was done manually by comparing both simulated and true observations for various values of the parameters and thereafter adapting the parameters for each simulation. Many automated calibration algorithms also operate in a similar iterative manner. To speed up this iterative procedure, the derivatives with respect to the estimated parameters are used in several automated calibration methods.

One of the very efficient ways to compute these derivatives is by using the adjoint method. In the adjoint method the dynamical model plays a role of a strong constraint. The objective function is defined, which is usually the sum of penalty terms. These penalty terms are usually weighted sums over time of all data-model misfits at observation locations. Weights are determined by the observation error covariances. At each iteration of the adjoint method, the the gradient (derivative) of the objective function is computed using one forward simulation with the original nonlinear model and another simulation backwards in time with the adjoint model. Minimization along the gradient's direction at the end of each iteration leads to new parameter values. Another forward and backward iteration can then be started, and so on, until the convergence is achieved. A detailed description of the application of the adjoint method in atmosphere and ocean problems can be found in [64].

The estimation of uncertain parameters in shallow water flow models using the adjoint method has been studied e.g. [23], [24], [92], [43], [58]. [23] shows that bottom friction coefficient and depth corrections could be inferred from sea level observations at given tidal stations. [92] studied the spatial variability of the bottom friction coefficient in a tidal model of the lower Hudson estuary by using the adjoint method. [43] developed an inverse 3D shallow water flow model in which depth values, bottom friction coefficients and boundary conditions were estimated. The method has also been studied in groundwater flow models (e.g. [16], [84]).

These inverse (adjoint) models require adjoint codes to evaluate products of the transposed Jacobian of the underlying numerical model. One of the drawbacks of the adjoint method is the programming effort required for the implementation of this adjoint model code. Research has recently been carried out on automatic generation of computer codes for the adjoint, and adjoint compilers have now become available (see e.g. [38], [47]). Even with the use of these adjoint compilers, this is a significant programming effort that hampers new applications of the method.

1.3 Model reduction

Despite the continued rapid advances in computing speed and memory, the increase of the complexity of numerical models used by engineers persists in outpacing them. Nowadays, numerical modeling of physical processes used by engineers requires a lot

of computer resources. Even when there is an access to the latest hardware, simulations are often extremely computationally intensive and time-consuming when full-numerical models of dimensions $10^4 - 10^9$ are under consideration. The numerical simulations in such large-scale settings lead in turn to unmanageably large demands on computational resources, which is the main motivation for system approximation (model reduction).

In system and control theory we associate model reduction with model-order reduction (MOR). Model order reduction aims at reducing the complexity of large-scale dynamical systems, while preserving (to the possible extent) their input-output behaviour. The goal of the model reduction is to produce a low dimensional system that has the same response characteristics as the original system. Low dimensionality means far less storage and thus far less evaluation time. We achieve this by suppressing redundant data that exist within the multi-dimensional system of full order. So model reduction can be referred as data compression.

The desire to approximate a complicated function with simpler formulation goes back to 1807 when Fourier gave an idea to approximate a function with few trigonometric terms, although fundamental automated methods in the field of model order reduction were published in 1980's and 1990's. In 1981 Moore published the method of truncated balanced realization [62]. In 1984 Glover gave the Hankel-norm reduction method [39]. In 1987 snapshot proper orthogonal decomposition method was proposed by Sirovich [78]. Full description of these methods can be found in [11], [75].

1.3.1 Proper Orthogonal Decomposition

Computers have also increased our capacity to not only simulate complicated systems, but also to collect and analyze large amounts of data. One might then process hundreds of millions of data points to obtain a few quantities of final interest. For example, expensive machinery might be instrumented and monitored over days with the sole objective of efficiently scheduling maintenance.

Proper orthogonal decomposition (POD), which is also named as the principal component analysis (PCA) or the discrete Karhunen-Loeve transform (KLT), is a powerful and elegant method of data analysis aimed at obtaining low-dimensional approximate descriptions of high-dimensional processes. It involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible.

It is a data driven projection based method invented by Karl Pearson (see [68]). [48] and [56] had used it as a statistical tool to analyze random process data. [59] gave the name POD, when the method was used to study the turbulent flow. The POD method has application in many fields like image processing, signal processing, data compression, oceanography, chemical engineering and fluid mechanics (see [71] for details). The method involves the calculation of the eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix obtained from numerical simulations of high a dimensional model, which is expected to provide

information about the dynamical behaviour of the system.

Nowadays significant research efforts are made to reduce the computational burden of the data assimilation by reducing the dimension of the system. For oceanic models the use of POD modes to identify a low-rank control space has shown promising results in the studies of [72], [44] and [15]. The common feature of these studies is that the computation of the principal components relies on the model dynamics. Recently [22] used the information of data assimilation system into the reduced-order procedure by implementing a dual-weighted proper orthogonal decomposition (DW-POD). The advantage of using DWPOD basis is that it may include lower energy modes that are more significant to the representation of the 4DVAR objective function. For atmospheric models [90] has recently used POD technique in combination with Monte Carlo method. The method not only simplifies the data assimilation procedure but also maintains the main advantages of the traditional 4DVAR.

In recent years, POD has also been applied for parameter estimation problems. POD analysis has been applied to inverse problems in hydrology to determine the uncertainty by [26]. [102] has used POD basis that includes the sensitivity with respect to different set of the parameters. Recently [99] has proposed a method for the parameter estimation with computational efficiency comparable to the adjoint method, but that does not require the implementation of the adjoint model code for the original nonlinear model.

1.4 Motivation and overview

It has been shown that variational data assimilation based on adjoint approach is suitable for estimating uncertain parameters in the shallow water flow models. This requires the implementation of the adjoint code which is a huge programming effort. This limitation of the adjoint method for the large scale numerical models has led to the search of suboptimal algorithms, which still preserve the advantages of the adjoint method.

POD based calibration method has been developed which does not require the implementation of the adjoint method ([99]). The method has been used successfully for the 3D ground water flow modeling by [101]. This thesis extends the previous work by investigating the new application of POD based calibration method to the large scale tidal model of the entire continental shelf, the DCSM. The thesis describes the potential use and benefits of this approach for estimating uncertain parameters in both ideal and real settings.

Balanced proper orthogonal decomposition (BPOD) is a model reduction method which considers both inputs and outputs of the system while determining the reduce subspace [73, 46]. The current thesis also investigates the possible usage of BPOD model reduction method within the framework of variational data assimilation.

The next chapter of this thesis consists of general introduction to the research. This is followed by four chapters describing the content of the carried out study. Finally, in the last chapter the most important results of this work are summarized.

Chapter 2 introduces the concept of the Sea level predictions in the Netherlands. The chapter provides a brief introduction to the Dutch continental shelf model (DCSM),

the numerical model mostly used in this study. The chapter considers topics mainly related to model equations, numerical approximations, operational forecasting using DCSM and model calibration. The Chapter also describes adjoint method commonly used for inverse modeling. The difficulties in the practical implementation of the adjoint method are also mentioned and discussed.

Chapter 3 presents a parameter estimation algorithm based on POD model reduction which does not require the implementation of the adjoint method [99]. Results of several synthetic experiments to estimate diffusion coefficient in a 2D advection diffusion model are presented to show the effectiveness of the POD based estimation method. The method has been extended by including balanced (information from the observations) transformation in the model reduction procedure referred as balanced proper orthogonal decomposition (BPOD). Results obtained from several twin experiments for the estimation of diffusion coefficient for both POD and BPOD procedures are also presented and discussed [4].

Preliminary results in Chapter 3 shows the validity of the POD based model reduction methods for parameter estimation. The POD based calibration algorithm was then developed to estimate the depth values in a tidal model of the North Sea, DCSM [5]. In Chapter 4, results of several twin experiments are presented which were used to evaluate the performance of the POD based calibration approach for the model DCSM.

A large scale tidal model for storm surge forecasting in the Netherlands has recently been developed. This model has approximately 10^6 computational grid points. As a next step to real life application, the POD based calibration approach was then implemented for the estimation of the water depth and space varying bottom friction coefficient values in this large-scale DCSM model. This was the first attempt to implement and evaluate the method with a very large-scale model and real data [9]. In Chapter 5, the results of these experiments are presented and discussed. The computational costs of the POD based calibration method is dominated by generation of an ensemble of forward model simulations. It is also found in the present study that a new ensemble is not always required with the new set of estimated parameters. Instead of defining a new model subspace of the leading eigenvectors with updated parameters by generating a new ensemble of the forward model simulations, the reduced model is obtained by projecting original model with the updated parameters onto the same subspace.

In Chapter 6, simulation perturbation method is presented and applied for the estimation of depth values in the model DCSM. Both twin and real experiments are discussed and the efficiency of the method is compared with the steepest descent and the POD based calibration methods [10].

Finally, in Chapter 7 the most important results of this research work are summarized and the conclusions of this thesis are formulated.

The Dutch Continental Shelf Model

Accurate water level forecasts at least six hours ahead are required for timely closure of the movable storm surge barriers in Eastern Scheldt and New Waterway. Besides being required for the closure of these movable barriers, water level predictions are also needed for raising alarms in extreme situations. Since mid 1980's these forecasts are based on numerical hydrodynamic model called the Dutch continental shelf model (DCSM). The model describes how the water level and velocities are related and evolve in time as a response to the wind forcing exerted on the Sea surface as well as to the tidal waves coming from the Atlantic Ocean.

This chapter is intended to provide a brief introduction about the storm surge prediction model DCSM the numerical model used in this study. The DCSM is based on WAQUA software package which is used for 2D modeling of water systems. The use of a 2D model implies that flow related quantities are computed in depth-averaged form, which is a commonly used approximation for tidal modeling. The details regarding the basic developments and WAQUA software are taken from technical documentation of WAQUA [1] and [98](www.waqua.nl).

Section 2.1 gives details of the basic equations of the model followed by their finite difference discretizations in Section 2.2. Section 2.3 elaborates the commonly used adjoint method for model calibration. Section 2.4 briefly overviews the historical efforts made in calibrating the DCSM model. Finally, Section 2.5 discusses issues of operational forecasting situation of the storm surges and explains how forecasts are made in operational setting.

2.1 The model equations

The governing equations used in DCSM are the non-linear 2D shallow water equations. The shallow water equations describe large scale water motions and depth-integrated horizontal flow. These equations are well suited for the numerical tidal flow

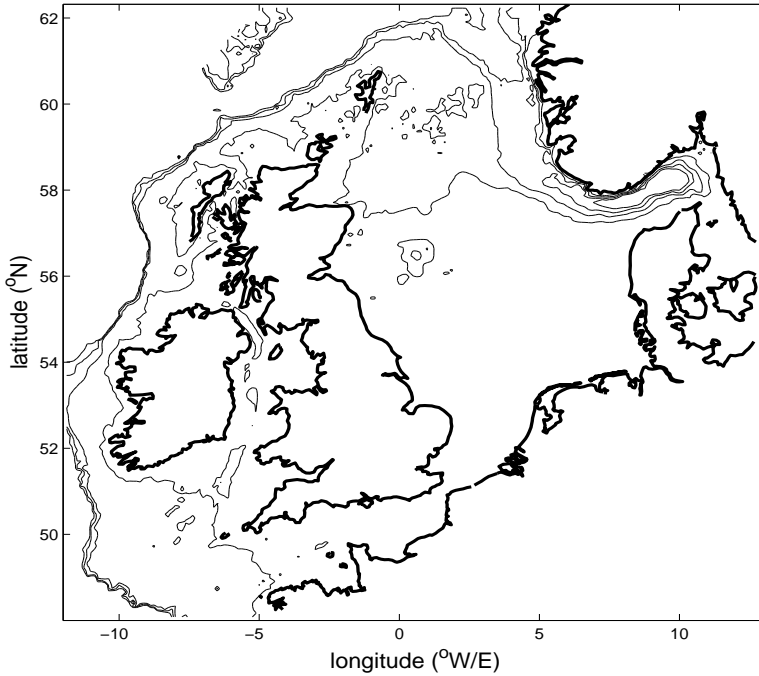


Figure 2.1: Area of the operational DCSM.

modeling. The equations of the model state, the conservation of mass and momentum read in a Cartesian coordinates system as

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial h}{\partial x} - fv + \frac{gu \sqrt{u^2 + v^2}}{HC_{2D}^2} = \frac{1}{\rho_w} \frac{\tau_x}{H} - \frac{1}{\rho_w} \frac{\partial p_a}{\partial x}, \quad (2.1)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial h}{\partial y} + fu + \frac{gv \sqrt{u^2 + v^2}}{HC_{2D}^2} = \frac{1}{\rho_w} \frac{\tau_y}{H} - \frac{1}{\rho_w} \frac{\partial p_a}{\partial y}, \quad (2.2)$$

$$\frac{\partial h}{\partial t} + \frac{\partial Hu}{\partial x} + \frac{\partial Hv}{\partial y} = 0, \quad (2.3)$$

where

- x, y = Cartesian coordinates in horizontal plane
- t = time coordinate
- u, v = depth-averaged current in x and y direction respectively
- h = water level above reference plane
- D = water depth below the reference plane
- H = total water depth ($D+h$)

f	=	coefficient for the Coriolis force
C_{2D}	=	2D Chezy coefficient
τ_x, τ_y	=	wind stress in x and y direction respectively
ρ_w	=	density of sea water
p_a	=	atmospheric pressure
g	=	acceleration of gravity

The Chezy coefficient C_{2D} used to account for bottom friction is a largely empirical parameter and assumed to be a function of depth. It is computed both in u and v points according to the one of the following two formulations

$$C_{2D} = \begin{cases} 65, & \text{if } D \leq 40m, \\ 65 + (D - 40), & \text{if } 40m < D \leq 65m, \\ 90, & \text{if } D \geq 65m, \end{cases} \quad (2.4)$$

or

$$C_{2D} = \frac{\{H\}^{\frac{1}{6}}}{c_m}, \quad (2.5)$$

where c_m is the manning coefficient. Without wind the stress at the free surface is zero. With wind the wind stress is prescribed using the wind velocity at the water surface as follows

$$\tau_x = C_d \rho_a u_w \sqrt{u_w^2 + v_w^2}, \quad (2.6)$$

$$\tau_y = C_d \rho_a v_w \sqrt{u_w^2 + v_w^2}, \quad (2.7)$$

where:

ρ_a	=	density of the air
C_d	=	wind stress coefficient
u_w	=	wind speed 10 m above the water surface in x direction
v_w	=	wind speed 10 m above the water surface in y direction

In order to obtain a unique solution to the above shallow water equations, a set of boundary conditions is prescribed at closed and open boundaries. At closed boundaries, such as coastlines and dams, the velocity normal to the boundary is zero:

$$v_{\perp} = 0. \quad (2.8)$$

So no inflow and outflow can occur through these boundaries. At the open boundaries no physical boundaries exist and thus artificial ones will have to be specified. Two boundary conditions are specified at open sea boundaries. The first is specified as

$$v_{\parallel} = 0. \quad (2.9)$$

For the second open boundary condition the water level h is specified in terms of the amplitude and phase of the harmonic components as follows

$$h(t) = h_0 + \sum_j A_j \cos(\omega_j t - \theta_j), \quad (2.10)$$

where

- h_0 = mean water level
 A_j = amplitude of tidal constituent j
 ω_j = angular velocity of j
 θ_j = phase of j

The open boundaries of the model are located in deep water in order to model explicitly the non-linearities of the surge-tide interaction. For simulations which include meteorological effects, the effects of atmospheric pressure are parametrized by a correction term at the boundaries. This correction is a function of the deviation from the average pressure p_{avg} :

$$h_0 = \frac{p - p_{avg}}{\rho a g}, \quad (2.11)$$

where p is the actual pressure and $p_{avg} = 1012hPa$ [43].

2.2 Numerical approximation

In this section the discretization method used to solve equations 4.26-4.28 is discussed. The dimension of the state vector for tidal models is usually huge and implies that the numerical codes should be optimized with respect to required memory access and computation time. Considering this optimization, the shallow water equations are solved on a standard staggered C-grid using an alternative direction implicit (ADI) finite difference method. Since the model is used in a wide variation of streaming conditions, unconditional stability and at least second order accuracy are basic requirements of the discretization method used. The ADI method is based on the work of Leendertse [55]. Stelling [82] stabilized the method by a special higher-order dissipative approximation of the cross advection term.

To improve the computational efficiency the ADI method splits one time step into two stages. Each stage consists of half a timestep. In both stages all the terms of the model equations are solved in a consistent way with at least second order accuracy in space. For each term the time levels are alternating. If in one stage a term is discretized implicitly in time, this term will be discretized explicitly in time in the other stage. As a result each term of the equations is solved second-order accurate in time. The splitting in an implicit and explicit part is arranged in such a way that implicit coupling only occurs along grid lines without coupling in the other directions.

The finite difference discretization scheme for the model equations (4.26-4.28) is illustrated here. We denote the index of a grid cell in the (x, y) coordinate by (m, n) and the time index by k , as shown in Figure 2.2. In the first part of the computation the velocity $v_{m,n}^{k+\frac{1}{2}}$ is determined explicitly based on $u_{m,n}^k, v_{m,n}^k$. The remaining variables $u_{m,n}^{k+\frac{1}{2}}$ and $h_{m,n}^{k+\frac{1}{2}}$ are found as the solution of implicit equations

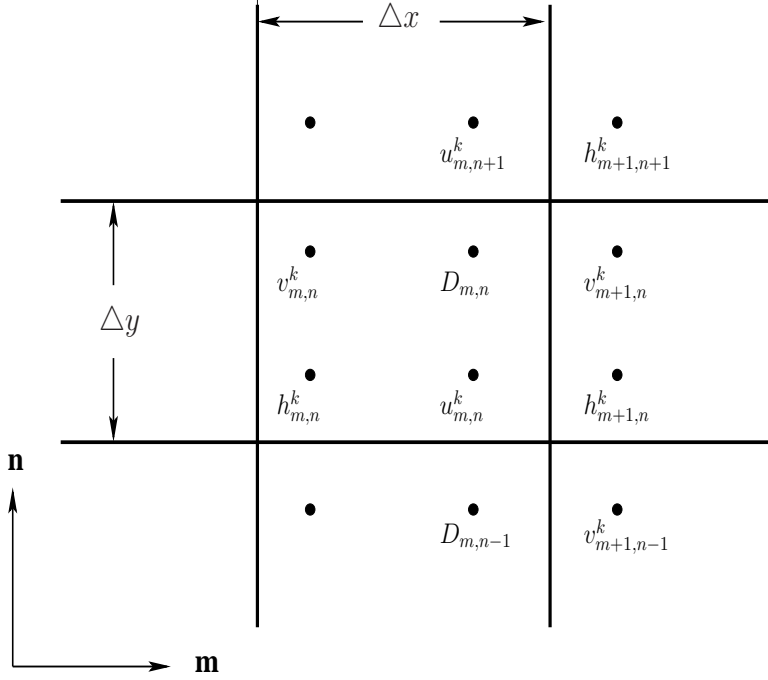


Figure 2.2: The computational grid

Step 1: time k to $k + \frac{1}{2}$

$$\frac{2}{\Delta t} \{v_{m,n}^{k+\frac{1}{2}} - v_{m,n}^k\} + u_{m,n}^k S_{+x}[v_{m,n}^{k+\frac{1}{2}}] + \frac{v_{m,n}^k}{2\Delta y} \{v_{m,n+1}^{k+\frac{1}{2}} - v_{m,n-1}^{k+\frac{1}{2}}\} + \frac{g}{\Delta y} \{h_{m,n+1}^k - h_{m,n}^k\} + f u_{m,n}^k + \frac{g v_{m,n}^{k+\frac{1}{2}} \sqrt{(u_{m,n}^k)^2 + (v_{m,n}^k)^2}}{(H^v)_{m,n}^k C_{2D}^2} = \frac{\tau_y}{\rho_w (H^v)_{m,n}^k} - \frac{1}{\rho_w \Delta y} \{p_{m,n+1}^k - p_{m,n}^k\}, \quad (2.12)$$

$$\frac{2}{\Delta t} \{u_{m,n}^{k+\frac{1}{2}} - u_{m,n}^k\} + \frac{u_{m,n}^{k+\frac{1}{2}}}{2\Delta x} \{u_{m+1,n}^k - u_{m-1,n}^k\} + v_{m,n}^{k+\frac{1}{2}} S_{oy}[u_{m,n}^k] + \frac{g}{\Delta x} \{h_{m+1,n}^{k+\frac{1}{2}} - h_{m,n}^{k+\frac{1}{2}}\} - f v_{m,n}^{k+\frac{1}{2}} + \frac{g u_{m,n}^{k+\frac{1}{2}} \sqrt{(u_{m,n}^k)^2 + (v_{m,n}^k)^2}}{(H^u)_{m,n}^k C_{2D}^2} = \frac{\tau_x}{\rho_w (H^u)_{m,n}^k} - \frac{1}{\rho_w \Delta x} \{p_{m,n+1}^k - p_{m,n}^k\}, \quad (2.13)$$

$$\frac{2}{\Delta t} \{h_{m,n}^{k+\frac{1}{2}} - h_{m,n}^k\} + \frac{(H^u u)_{m,n}^{k+\frac{1}{2}} - (H^u u)_{m-1,n}^{k+\frac{1}{2}}}{\Delta x} + \frac{(H^v v)_{m,n}^k - (H^v v)_{m,n-1}^k}{\Delta y} = 0, \quad (2.14)$$

In the second part, the roles of the m and n directions are changed. The velocity

component $u_{m,n}^{k+1}$ is based on $u_{m,n}^{k+\frac{1}{2}}$, $v_{m,n}^{k+\frac{1}{2}}$ and $h_{m,n}^{k+\frac{1}{2}}$, while $v_{m,n}^{k+1}$ and $h_{m,n}^{k+1}$ are again the solutions of implicit equations:

Step 2: time $k + \frac{1}{2}$ to $k + 1$

$$\frac{2}{\Delta t} \{u_{m,n}^{k+1} - u_{m,n}^{k+\frac{1}{2}}\} + \frac{u_{m,n}^{k+\frac{1}{2}}}{2\Delta x} \{u_{m+1,n}^{k+1} - u_{m-1,n}^{k+1}\} + v_{m,n}^{k+\frac{1}{2}} S_{+y}[u_{m,n}^{k+1}] + \frac{g}{\Delta x} \{h_{m+1,n}^{k+\frac{1}{2}} - h_{m,n}^{k+\frac{1}{2}}\} - f v_{m,n}^{k+\frac{1}{2}} + \frac{g u_{m,n}^{k+1} \sqrt{(u_{m,n}^{k+\frac{1}{2}})^2 + (v_{m,n}^{k+\frac{1}{2}})^2}}{(H^u)_{m,n}^{k+\frac{1}{2}} C_{2D}^2} = \frac{\tau_x}{\rho_w (H^u)_{m,n}^{k+\frac{1}{2}}} - \frac{1}{\rho_w \Delta x} \{p_{m,n+1}^k - p_{m,n}^k\}, \quad (2.15)$$

$$\frac{2}{\Delta t} \{v_{m,n}^{k+1} - v_{m,n}^{k+\frac{1}{2}}\} + u_{m,n}^{k+1} S_{ox}[v_{m,n}^{k+\frac{1}{2}}] + \frac{v_{m,n}^{k+1}}{2\Delta y} \{v_{m,n+1}^{k+\frac{1}{2}} - v_{m,n-1}^{k+\frac{1}{2}}\} + \frac{g}{\Delta y} \{h_{m,n+1}^{k+1} - h_{m,n}^{k+1}\} + f u_{m,n}^{k+1} + \frac{g v_{m,n}^{k+1} \sqrt{(u_{m,n}^{k+1})^2 + (v_{m,n}^{k+\frac{1}{2}})^2}}{(H^v)_{m,n}^{k+\frac{1}{2}} C_{2D}^2} = \frac{\tau_y}{\rho_w (H^v)_{m,n}^{k+\frac{1}{2}}} - \frac{1}{\rho_w \Delta y} \{p_{m,n+1}^k - p_{m,n}^k\}, \quad (2.16)$$

$$\frac{2}{\Delta t} \{h_{m,n}^{k+1} - h_{m,n}^{k+\frac{1}{2}}\} + \frac{(H^u u)_{m,n}^{k+\frac{1}{2}} - (H^u u)_{m-1,n}^{k+\frac{1}{2}}}{\Delta x} + \frac{(H^v v)_{m,n}^{k+1} - (H^v v)_{m,n-1}^{k+1}}{\Delta y} = 0, \quad (2.17)$$

where the numerical operators $S_{+y}[u_{m,n}]$, $S_{oy}[u_{m,n}]$, $S_{+x}[v_{m,n}]$ and $S_{ox}[v_{m,n}]$ are defined as

$$S_{+y}[u_{m,n}] = \begin{cases} \frac{3u_{m,n} - 4u_{m,n-1} + u_{m,n-2}}{2\Delta y}, & \text{if } v_{m,n} \geq 0, \\ \frac{-3u_{m,n} + 4u_{m,n+1} - u_{m,n+2}}{2\Delta y}, & \text{if } v_{m,n} < 0, \end{cases} \quad (2.18)$$

$$S_{oy}[u_{m,n}] = \frac{u_{m,n+2} + 4u_{m,n+1} - 4u_{m,n-1} - u_{m,n-2}}{12\Delta y}, \quad (2.19)$$

$$S_{+x}[v_{m,n}] = \begin{cases} \frac{3v_{m,n} - 4v_{m-1,n} + v_{m-2,n}}{2\Delta x}, & \text{if } v_{m,n} \geq 0, \\ \frac{-3v_{m,n} + 4v_{m+1,n} - v_{m+2,n}}{2\Delta x}, & \text{if } u_{m,n} < 0, \end{cases} \quad (2.20)$$

$$S_{ox}[v_{m,n}] = \frac{v_{m+2,n} + 4v_{m+1,n} - 4v_{m-1,n} - v_{m-2,n}}{12\Delta x}, \quad (2.21)$$

To ensure the well-posedness of the numerical tidal model, Dirichlet conditions are imposed on boundaries. The conditions are derived in a similar way to the finite difference equations just described.

2.2.1 State space model

The main aim of the present work is to develop an efficient calibration method for general tidal models. The state space representation of the finite difference equations (2.12) and (2.17) for the deterministic model DCSM is given by

$$X(t_{i+1}) = M_i[X(t_i), \gamma], \quad (2.22)$$

where the nonlinear and deterministic operator M_i consists of the coefficients representing the ADI scheme and the forcings and $X(t_{i+1}) \in \mathfrak{X}^n$ is a state vector containing water levels h and velocities u and v at time $(i + 1)$ for all the grid points in the DCSM area. γ is a vector of uncertain parameters which need to be determined. The state space representation is used to provide a convenient and compact way to analyze the model DCSM with multiple inputs and outputs.

2.3 Adjoint method

In recent years there have been a large developments of various data assimilation applications based on variational methods. The approach is also referred to as four dimensional variational data assimilation (4DVAR). It computes a particular solution of the numerical model which matches in the best way the available data during a certain time interval. The approach is conceptually equivalent to performing a least square fit of a set of model parameters to the available data. The parameters to be estimated may be the unknown coefficients in the model equations, boundary conditions or forcing terms.

As explained earlier, data assimilation is a method of integrating observations with physically based mathematical models. The observations taken from the actual system are never perfect. Suppose now that we have imperfect observations $Y(t_i) \in \mathfrak{X}^q$ that are related to the model state at time t_i through

$$Y(t_i) = HX(t_i) + \eta(t_i), \quad (2.23)$$

where $H : \mathfrak{X}^n \rightarrow \mathfrak{X}^q$ is a linear observation operator that maps the model fields on observation space, $\eta(t_i)$ is a white Gaussian observation noise process with zero mean and covariance matrix R_i introduced to model the uncertainties associated with observation process.

The idea of parameter estimation is to estimate the uncertain model parameters by minimizing the measure of residuals. We first define an objective function $J(\gamma)$ as a measure for distance between the observations and model results:

$$J(\gamma) = \sum_i [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] \quad (2.24)$$

Here, we have chosen a most commonly used measure to define J , the generalized least square criterion or l_2 norm. The optimal parameter is obtained by minimizing this objective function J . If the information regarding observations is limited, a prior information about the parameters can also be included in the objective function

$$J(\gamma) = \sum_i [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] + (\gamma - \gamma^b)^T P_\gamma^b (\gamma - \gamma^b), \quad (2.25)$$

where the prior information term is referred to as background term and prevents unrealistic parameter estimates. Thus the estimated parameters will remain close to the prior parameter values.

The efficient minimization of the objective function J is often based on quasi-Newton methods. In the case of the quadratic objective function the quasi-Newton routine iterates approximately $(n^p + 1)$ times [33], where n^p is the number of parameters to be estimated. These methods require the computation of the gradient of the objective function J . The gradient gives information about the direction (positive or negative) and the size of adjustments for each individual parameter. The gradient of the J is usually obtained either by finite difference method or adjoint methods.

In most situations it is not possible to establish explicit analytical expressions for the gradient. It is however possible to numerically and approximately determine the gradient through finite-difference method (perturbation method). To obtain the gradient of the objective function J with respect to the components γ_k we can use the one sided perturbation

$$\nabla J_k \approx \frac{J(\gamma_k + \Delta\gamma_k) - J(\gamma_k)}{\Delta\gamma_k}, k = \{1, \dots, n^p\}, \quad (2.26)$$

where $\Delta\gamma_k$ is the perturbation size. Since only one component $\Delta\gamma_k$ is perturbed at a time and each perturbation requires one simulation run of the original nonlinear model to evaluate $J(\gamma_k + \Delta\gamma_k)$, the method is not applicable when there is a large number of uncertain parameters to estimate.

Spall [80] proposed simultaneous perturbation stochastic approximation (SPSA) algorithm in which all the estimated parameters were perturbed at one time stochastically. Since the perturbation are stochastic, the calculated gradient is also stochastic. However its expectation is the true gradient [80], [34]. The method has recently been used successfully by [7] for shallow water flow models. The method has also been applied in petroleum engineering by [34] for history matching problems and by [103] to production optimization, but the results demonstrated that the computational cost of the method is comparable to finite difference method.

In most data assimilation problems the number of uncertain parameters is usually large. In such a case, the variational or adjoint approach is the most suitable approach to efficiently compute the gradient of objective function J . The adjoint method computes the exact gradient efficiently. The principle of the adjoint method is based on the systematic use of the chain rule for differentiation.

Since for every parameter γ the system state $X(t_{i+1})$ has to satisfy the model constraint (2.22), the objective function J in equation (2.24) can be re-written as

$$J(\gamma) = \sum_{i=1}^m [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] + \sum_{i=1}^m \nu(t_{i+1})^T [X(t_{i+1}) - M_i[X(t_i), \gamma]], \quad (2.27)$$

where $\nu(t_i)$ is the vector of Lagrange multipliers or adjoint state variables. Now the incremental changes in J , $X(t_i)$ and ν due to incremental change in one of the compo-

nents of γ gives

$$\begin{aligned} \Delta J(\gamma) = & \sum_{i=0}^{m-1} v(t_{i+1})^T \{ \Delta X(t_{i+1}) - \frac{\partial M_i[X(t_i), \gamma]}{\partial X(t_i)} \Delta X(t_i) \} + \\ & \sum_{i=0}^{m-1} \Delta v(t_{i+1})^T \{ X(t_{i+1}) - M_i[X(t_i), \gamma] \} - \\ & \sum_{i=0}^{m-1} v(t_{i+1})^T \frac{\partial M_i[X(t_i), \gamma]}{\partial \gamma} \Delta \gamma - \\ & \sum_{i=1}^m 2H^T R_i^{-1} [Y(t_i) - H(X(t_i))]. \end{aligned} \quad (2.28)$$

The above expression after rearrangement yields

$$\begin{aligned} \Delta J(\gamma) = & \sum_{i=1}^{m-1} \Delta X(t_i)^T \{ v(t_i) - [\frac{\partial M_i[X(t_i), \gamma]}{\partial X(t_i)}]^T v(t_{i+1}) \} - \\ & 2H^T R_i^{-1} [Y(t_i) - H(X(t_i))] + v(t_m) \Delta X(t_m)^T - \\ & \sum_{i=0}^{m-1} v(t_{i+1})^T \frac{\partial M_i[X(t_i), \gamma]}{\partial \gamma} \Delta \gamma. \end{aligned} \quad (2.29)$$

The adjoint states $v(t_i)$ are still free variables. An expression for the adjoint model $v(t_{i+1})$, $i \in \{m-1, \dots, 1\}$, solved backward in time follows from

$$v(t_i) = [\frac{\partial M_i[X(t_i), \gamma]}{\partial X(t_i)}]^T v(t_{i+1}) + 2H^T R_i^{-1} [Y(t_i) - H(X(t_i))] \quad (2.30)$$

with $v(t_m)$ equal to zero. Once the adjoint states $v(t_i)$ are known, the gradient $\frac{\partial J}{\partial \gamma}$ of the objective function J can be computed using the following expression

$$\nabla J_k = \frac{\partial J}{\partial \gamma} = - \sum_{i=0}^{m-1} v(t_{i+1})^T \frac{\partial M_i[X(t_i), \gamma]}{\partial \gamma}. \quad (2.31)$$

Regardless of the number of parameters, the time required to compute the gradient using adjoint technique is more or less identical and is comparable to the computational time needed for a single simulation run of the nonlinear model (2.22). It requires one forward simulation with the original nonlinear model (2.22) and a second additional simulation backward in time with the adjoint model.

The main hurdle in the use of adjoint method is its implementation. The adjoint model requires adjoint code to determine the tangent linear model of the original nonlinear model. This implies that tremendous programming effort is required to obtain the adjoint model which is approximately equivalent to the programming effort required to build an original model. Moreover, if the original model needs to be modified, as in the case of tidal models studied in this thesis, the adjoint model should also be modified and thus the maintenance costs of the software increases. In recent years research has been carried out on automatic generation of computer codes for the

adjoint, and adjoint compilers have now become available (see e.g. [38], [47]). Even with the use of these adjoint compilers, this is a significant programming effort that hampers new applications of the method. Moreover since the adjoint equations need to be integrated backward in time, the determined states of the original problem must be stored for all the time steps. The memory access will therefore be huge for large scale problems.

2.3.1 Procedural flow for the classical adjoint method

Variational schemes are used to minimize the objective function J iteratively. The following steps are performed for the estimation of uncertain parameter with the classical adjoint method.

1. Choose an initial value γ^b for the parameter to be determined.
2. Run the original nonlinear model to obtain the objective function J .
3. Run the adjoint model backward in time to compute the gradient of the objective function J .
4. Minimize the objective function J based on the quasi-Newton method to get the updated parameters.
5. Repeat from step 2 if the parameter value improves significantly or the predefined convergence criterion is achieved. The final values obtained in this way will give the optimal parameters.

2.3.2 The LBFGS method

For the problem of minimizing a multivariable function quasi-Newton methods [41] are widely employed. These methods involve the approximation of the Hessian matrix (or its inverse) of the objective function J . The LBFGS (Limited memory-Broyden-Fletcher-Goldfarb-Shanno) [65] method is basically a method to approximate the Hessian matrix in the quasi-Newton method of optimization. It is a variation of the standard BFGS method.

$$\gamma_{l+1} = \gamma_l - a_l \hat{H}_l \hat{g}_l(\gamma_l), l = 0, 1, \dots \quad (2.32)$$

where a_l is a step length, \hat{g}_l is the local gradient of the objective function, and \hat{H}_l is the approximate inverse Hessian matrix which is updated at every iteration by means of the formula

$$\hat{H}_{l+1} = V_l^T \hat{H}_l V_l + \varrho_l d_l d_l^T \quad (2.33)$$

where

$$\varrho_l = \frac{1}{y_l^T d_l} \quad (2.34)$$

$$V_l = I - \varrho_l y_l d_l^T \quad (2.35)$$

and

$$d_l = \gamma_{l+1} - \gamma_l \quad (2.36)$$

$$y_l = \hat{g}_{l+1} - \hat{g}_l \quad (2.37)$$

Using this method, instead of storing the matrices \hat{H}_l , one stores a certain number of pairs, say m , of pairs s_l, y_l that define them implicitly. The product of $\hat{H}_l g_l$ is obtained by performing a sequence of inner products involving g_l and the m most recent vector pairs s_l, y_l to define the iteration matrix. Line minimization is needed for determining γ in equation 2.32.

2.4 An overview on DCSM calibration

The task of calibration in the current application, is to reconstruct the physical processes under astronomical conditions as accurately as possible. The mathematical models always include some undetermined parameters that are only known with limited accuracy. These parameters are adjusted to ensure that the model represents the selected known cases with sufficient accuracy. The adjustments of uncertain parameters must be physically acceptable, since the aim of model calibration is obtaining the model that can be used in the forecast mode.

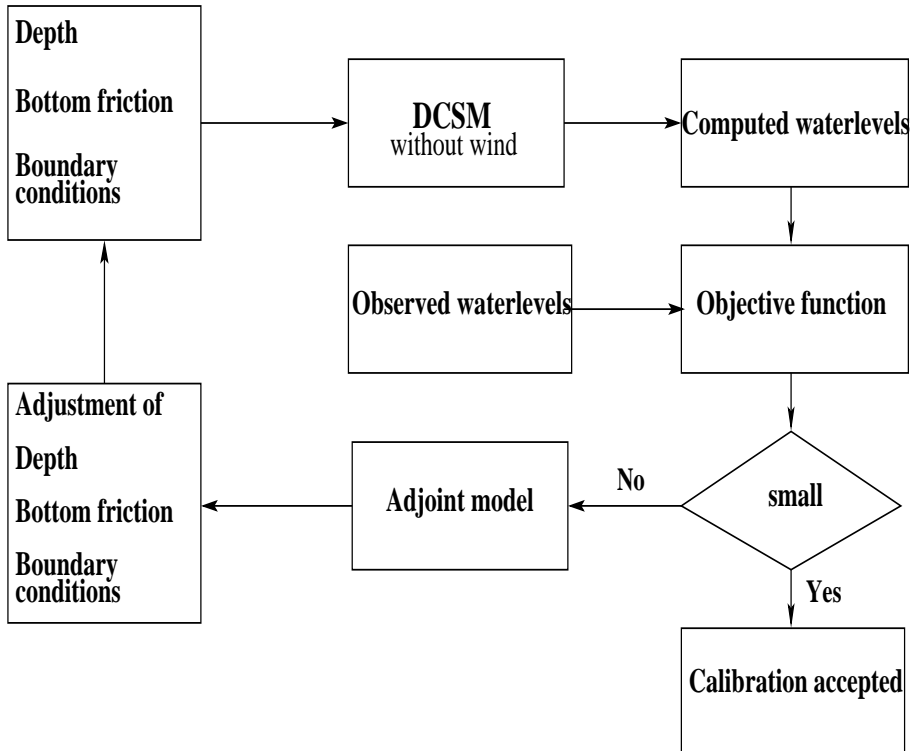


Figure 2.3: Block diagram of calibration process using adjoint method [2]

The objective of DCSM calibration is to make best estimates of the uncertain pa-

rameters, i.e. find parameters such that the water levels computed by the DCSM are as close as possible to the observed water levels. Bathymetry of the DCSM was obtained from nautical charts. Since the main purpose of these charts is to allow ships a safe course in any tidal situation, they usually give details of shallow rather than deep water areas. Hence, depth is a parameter on which model can be calibrated. At the open boundaries the water level is described in terms of harmonic components. These tidal components were estimated on the basis of encompassing models matched with nearby coastal and pelagic tidal data. Hence, the amplitudes and phases of these harmonic components are also the parameters on which the DCSM can be calibrated. The bottom friction in DCSM is described in terms of the Chezy coefficient, which is largely empirical model parameter.

In the early development of DCSM the calibration was done using sensitivity studies and making changes manually [93]. Later, automated procedures based on a variational method were developed for this purpose [86], [2]. Few efforts were also made in calibrating DCSM using the adjoint method [97]). Local changes were made in the model to compensate for the lack of resolution. This was done mainly near narrow channels between islands where a small misrepresentation of the bathymetry may lead to large changes in the flow. The calibration of the DCSM was also performed to assess the potential impact of satellite altimeter sea level observations [63], [61]. Although the impact of the satellite observations proved to be small with regard to the amount of data and level of accuracy available, it has provided some improvements to the calibration of the DCSM. Figure 2.3 presents flowchart of the calibration process with the adjoint method for DCSM model.

The difficulty in the use of adjoint method is the implementation and maintenance of adjoint model code. The implementation of adjoint model code requires significant programming effort that hampers its application. Moreover, every time the forward model code is altered the adjoint code also needs to be updated. So for the adjoint model it is necessary to keep track of all these alterations and modifications. In this thesis we have developed a method for the estimation of uncertain parameters in the model DCSM. The advantage of this method is that it efficiently estimates the uncertain parameters without the implementation of the adjoint code.

2.5 Operational forecasts using DCSM

The Dutch coast is divided in several zones, where each zone has its own warning threshold according to tidal amplitude and time of high water. Table 2.1 presents the warnings and alarm levels for different zones along Dutch coast. As described earlier, the forecasts regarding storm surges are made using the DCSM at KNMI. If the water levels in such forecasts exceed certain thresholds the SVSD is informed. If the SVSD hydrologists on duty decide that one or more warning level may be reached in response to these forecasts, the SVSD office is staffed. These warnings are made at least 6 hours ahead to provide required time which is necessary for preparation.

For operational correction of forecasts with observations a steady-state Kalman filter was developed [42]. This filter assimilates the selected water level observations from tide gauge stations along the British and Dutch coasts for selected set of stations.

Zone	Reference station	Warning (cm)	Alarm level (cm)
Schelde	Vlissingen	330	370
West Holland	Hoek van Holland	220	280
Den Helder	Den Helder	190	260
Harlingen	Harlingen	270	330
Delfzijl	Delfzijl	300	380

Table 2.1: Warning and alarm levels used by SVSD (level relative to normal Amsterdam level (NAP))

Since the shallow water equations for water much deeper than vertical tidal range are nearly linear, the Kalman filter can approximate in a very efficient manner. This requires only 10% additional computational cost. The filter can correct the forecasts up to 12 hours ahead if the forecasts deviate from the available water level observations.

The DCSM model including the steady state Kalman filter is installed in the KNMI for daily operational forecasting [25], where it is the part of automatic production line (APL). This APL was developed to produce regular numerical forecasts with a minimum of human intervention. The heart of APL is a limited area atmospheric circulation model (HIRLAM), which currently provides an analysis of the state of the atmosphere and a forecast for up to 48 h ahead four times per day with a resolution of approximately 22 km.

Using POD and BPOD within variational data assimilation ¹

3.1 Introduction

Proper Orthogonal Decomposition (POD) method provides an efficient means of deriving the reduced basis for high order nonlinear systems. This technique essentially identifies the most energetic modes in a time dependent system thus provides a means of obtaining a low dimensional description of the system dynamics. The POD method has been extensively used in recent years and applied successfully to different fields [13], [49], [51], [60], [3], [57], [100], [14]. A detailed description of POD research can be found in [40].

Another obvious application of POD is four dimensional variational data assimilation (4DVAR) which can be seen as an optimal control problem [88]. The variational (adjoint) method aims at adjusting a number of unknown control parameters on the basis of given data. The control parameters might be model initial conditions or model parameters ([88], [89]). Memory storage requirements impose a severe limitation on the size of assimilation studies. The POD has recently been applied successfully in data assimilation for the state and parameter estimation [67], [26], [102], [15]. Another drawback of the adjoint method is the computation of the gradient which requires the implementation of adjoint model code. The implementation and maintenance of the adjoint model is very expensive and thus hampers its applications.

The POD method however, can yield unpredictable results as it neglects low energy modes that may be important to the dynamics and POD modes are sensitive to the empirical data used and the choice of inner product [73]. An alternative POD method (referred as Balanced proper orthogonal decomposition (BPOD)) is to combine the concepts from both POD and balanced truncation methods. The balanced truncation method uses both controllable and observable subspaces while obtaining a low rank subspace for the reduced model. The balanced truncation methods are designed for

¹This chapter is a slightly revised version of [4]

stable and linear systems [104]. The balanced truncation of stable linear system is stable, but POD models of nonlinear systems may be unstable even if the nonlinear system is linearly stable at the origin ([79]). The main goal of BPOD is to achieve approximate balanced truncation with computational cost as in POD. It has already been applied by [104], [52] and recently by [73], [46].

The main motivation for the work presented in this chapter is inspired by the work done by [73], [99]. Vermeulen and Heemink [99] proposed a method based on POD, which shifts the minimization into lower dimensional space and avoids the implementation of the adjoint of the tangent linear approximation of the original nonlinear model. In their approach, an ensemble of snapshot vectors of forward model simulations is used to determine an approximation of the covariance matrix of the model variability and a small number of leading eigenvectors of this matrix are used to define a model subspace. By projecting the original model onto this subspace, an approximate linear reduced model is obtained. Due to the linear character of the reduced model, its adjoint can be implemented easily and the minimizing problem is solved completely in reduced space with very low computational cost. This process is repeated several times while generating new set of snapshots (ensemble) which is closer to the new estimated parameters.

Rowley [73] computed the balancing transformation directly from snapshots of empirical Gramians using singular value decomposition (SVD), without needing to compute the Gramians themselves. The method allows the empirical observability Gramian to be computed efficiently even when the number of outputs is large. The work presented here, uses POD and BPOD in the context of variational data assimilation. Employing the examples of POD and BPOD techniques, we demonstrate the usefulness of these techniques to parameter estimation and presents experiments with the 2D-advection diffusion model for the transport of pollutant. We have shown that the use of model reduction allows the optimization to be much efficient than the one with the original model.

In the next section we briefly describe the projection based model reductions techniques used in this chapter and Section 3.3 explains the procedure for collecting snapshots and the basis vectors (patterns) required for the simulation of reduce model in the context of variational data assimilation. The numerical results with 2D advection diffusion model to estimate space varying diffusion are presented in Section 3.4. Conclusions are given in Section 3.5.

3.2 Model reduction

Model reduction represents the solution to a problem as a truncated series of known basis function and independent coefficients. Roughly speaking, the goal of model reduction methods is to replace the initial data by data that are optimal in terms of storage capacity. This is achieved by suppressing redundant data that exists within multi-dimensional datasets. Two of the most well-known methods of model reduction used in control are the proper orthogonal decomposition (POD) and the method of balanced truncation. Balanced proper orthogonal decomposition (BPOD) is a model reduction method which combines the ideas and methodology from both of these methods. This

Section briefly explains the methodology of these model reduction methods.

3.2.1 Proper Orthogonal Decomposition

The proper orthogonal decomposition (POD), also known as principle components analysis, has been widely used for a broad range of applications. The main idea is, given a set of data that lies in a vector space V , to find a subspace V_r of fixed dimension r , such that the error in the projection onto the subspace is minimized.

We start by collecting the set of n^s snapshots of some physical process. Each sample of snapshots X_i which is defined on a set of n node X stands for an n dimensional vector X_j . i.e.,

$$X_i = \{X_{1i}, X_{2i}, \dots, X_{ni}\}', i \in \{1, 2, \dots, s\} \quad (3.1)$$

The elements within a snapshot represent the signal for a specific location in the model, possibly for multiple quantities. Define the vector X_b of background state and correct each snapshot vector so that

$$E_i = X_i - X^b, i \in \{1, 2, \dots, s\} \quad (3.2)$$

These corrected snapshots are arranged in matrix E , which denote the new ensemble. The covariance matrix Q can be constructed from the ensemble E of the snapshots by taking the outer product:

$$Q = EE^t \quad (3.3)$$

The dimension n often exceeds 10^4 , so direct solution of eigenvalue problem is not feasible. To shorten the calculation time necessary for solving the eigenvalue problem for this high dimensional covariance matrix, we define a covariance matrix G as the inner product. In the method of snapshots ([78]), one then solves the $n^s \times n^s$ eigenvalue problem

$$GZ_i = E^t E Z_i = \lambda_i Z_i, i \in \{1, 2, \dots, s\} \quad (3.4)$$

with λ_i are the eigenvalues of the above eigenvalue problem. The eigenvectors Z_i may be chosen to be orthonormal and the POD modes p_i are then given by

$$p_i = E Z_i / \sqrt{\lambda_i} \quad (3.5)$$

i.e., in matrix form;

$$P = E Z \Lambda^{-1/2} \quad (3.6)$$

Where $Z = \{Z_1, Z_2, \dots, Z_s\}$ and $P = \{p_1, p_2, \dots, p_s\}$ and Λ is a diagonal matrix containing the eigenvalues.

A physical explanation of POD modes is that they maximize the average energy in the projection of data onto subspace spanned by the modes. The eigenvalues λ_i provide a measure (ψ_i) for the relative energy associated with corresponding POD modes p_i :

$$\psi_i = \frac{\lambda_i}{\sum_{l=1}^s \lambda_l} \cdot 100\%, i = \{1, 2, \dots, s\} \quad (3.7)$$

We collect p_r ($r < s$) modes such that $\psi_1 > \psi_2 > \dots > \psi_r$ and they totally explain at least the required variance ψ^e .

$$\psi^e = \sum_{l=1}^r \psi_l \quad (3.8)$$

3.2.2 Balanced Truncation

Given a linear system defined by

$$\frac{dX}{dt} = AX + Bu \quad (3.9)$$

$$Y = HX \quad (3.10)$$

where $A \in \mathfrak{R}^{n \times n}$; $B \in \mathfrak{R}^{n \times n^u}$ and $H \in \mathfrak{R}^{n^o \times n}$. The main idea behind balanced truncation is to consider both inputs and outputs of the system governed by equation 3.9 and 3.10 when determining which states to keep in the reduced model structure.

Reduction of the systems will be achieved by retaining only certain states in the representation. This is equivalent to defining a certain subspace within the state space. Two important subspaces are the controllable and observable subspaces. The controllable subspace is the space that can be obtained with zero initial state and a given input $u(t)$, whereas the observable subspace comprises those states that as initial conditions could produce a nonzero output $Y(t_i)$ with no external input. The controllability and observability Gramians W_c and W_o are each an $n \times n$ matrix whose eigenvectors span the controllable and observable subspaces, respectively. These matrices are defined for the linear system (3.9) and (3.10) as

$$W_c = \int_0^{\infty} e^{At} B B^* e^{A^* t} dt \quad (3.11)$$

$$W_o = \int_0^{\infty} e^{A^* t} H^* H e^{At} dt \quad (3.12)$$

where the asterisk denotes the complex conjugate transpose. The Gramians are symmetric, positive-semi definite matrices usually computed by solving the Lyapunov equations given by

$$AW_c + W_c A^* + B B^* = 0 \quad (3.13)$$

$$A^* W_o + W_o A + H^* H = 0 \quad (3.14)$$

In order to obtain balanced realization of the system (3.9) and (3.10), a state transformation T is chosen such that controllability and observability Gramians are diagonal and equal. This transformation is obtained by computing appropriate scaled eigenvectors of the product of both the Gramians W_{co}

$$W_{co} = W_c W_o = T \Lambda T^{-1} \quad (3.15)$$

where $T_i; i = \{1, \dots, n\}$ are the eigenvectors that describes the balancing transformation. here the eigenvalues λ_i are positive and real. The square root of these eigenvalues λ_i gives the Hankel singular values σ_i .

$$\sqrt{\lambda_i} = \sigma_i \quad (3.16)$$

The process of balanced truncation is completed by first finding the balancing transformation T and then truncating those controllable-observable states which have less effect on inputs and outputs.

Method of snapshots

Instead of solving the Gramians by solving Lyapunov equations (3.13) and (3.14), one may compute them from data by the numerical simulations. This was the original approach used by [62]. The quantity $e^{At}B$ in equation 3.11 is simply the impulse response of the single input / single output system. So if so we have n^u number of inputs can write W_c as

$$W_c = \int_0^\infty (X_1(t)X_1(t)^* + \dots + X_{n^u}(t)X_{n^u}(t)^*)dt \quad (3.17)$$

Note the similarity to the POD dataset of snapshots (3.1) and the above expression. The POD modes for this dataset of impulse responses are just the largest eigenvectors of W_c . If data from simulations is used to find the impulse responses, then it is usually given at discrete times t_1, \dots, t_m and the integral above becomes a quadrature sum as in (3.1). Thus we can write W_c as:

$$W_c = EE^T \quad (3.18)$$

For the observability Gramians, we need to consider the adjoint system

$$z' = A^*z + H^*u_o \quad (3.19)$$

If n^q is the number of outputs then the observability Gramians is given by

$$W_o = \int_0^\infty (z_1(t)z_1(t)^* + \dots + z_{n^q}(t)z_{n^q}(t)^*)dt. \quad (3.20)$$

One then form the data matrix F , as in (3.1) and observability Gramians becomes

$$W_o = FF^T \quad (3.21)$$

The method requires n^q integrations of the adjoint system, where q is the number of outputs. Thus it is not feasible to use the method with large number of outputs.

3.2.3 Balanced POD

Balanced POD (BPOD) is an approximation to balanced truncation. The main goal of BPOD is to achieve approximate balanced truncation with computational cost comparable to POD. The BPOD method presented here has two components:

1. compute the balancing transformation directly from snapshots of empirical Gramians using singular value decomposition (SVD), without needing to compute the Gramians themselves.
2. enable traceable computation even when the number of outputs is large.

As described in previous section, the balancing transformation is obtained from the dominant eigenvectors of the product of two Gramians given by W_{co} . The Gramians are two $(n \times n)$ matrices. For large scale real time systems, where number of states are large, the idea of finding eigenvectors from these matrices is not feasible. Here the balancing modes are therefore obtained from the data matrices E and F by forming the SVD of the matrix $F^T E$

$$F^T E = U \Sigma V = (U_1 \quad U_2) \begin{pmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \quad (3.22)$$

where $\Sigma_1 \in \mathbb{R}^{r \times r}$ is invertible, r is the rank of matrix $F^T E$ and $U_1^T U_1 = V_1^T V_1 = I_r$. The advantage in using this method is balancing transformation is obtained by SVD of matrix of dimension $(n^u \times n^q)$. We can now define the transformation matrices S and T as

$$S = \Sigma_1^{-1/2} U_1^T F^T, \quad (3.23)$$

$$T = E V_1 \Sigma_1^{-1/2}. \quad (3.24)$$

If $r = n$, then the matrix Σ_1 contains the Hankel singular values, T determines the balancing transformation and S is its inverse [73]. Moreover, if $r < n$, then the columns of T form the first r columns of the balancing transformation and the rows of S form the first r rows of the inverse transformation. The major advantage of using the above method for computing the balancing transformation is that the Gramians themselves are not computed. Only one SVD of matrix with dimension $n^s \times n^q$ are required. Thus overall computation time is similar POD except that here one also needs to compute adjoint snapshots, which are required in POD method.

Outer Projection

As explained earlier, using impulse responses to compute adjoint snapshots is not feasible with large number of outputs. The outer projection (OP) method can be used to compute adjoint snapshots in this case. The main idea is to project the output onto an appropriate subspace in such a way that the input-output behavior is almost unchanged [73]. Instead of the linear system given by (3.9) and (3.10), consider the related system

$$\frac{dX}{dt} = AX + Bu \quad (3.25)$$

$$Y = P_r H X \quad (3.26)$$

where P_r is an orthogonal projection with rank r . The projection P_r will allow us to compute the observability Gramians W_o using only r simulations of the adjoint system instead of n^d simulations. We choose P_r such that the input-output behavior of (3.25-3.26) is almost similar to the input-output behavior of (3.9-3.10). Now we can write the adjoint system as

$$\dot{z} = A^* z + H^* P_r u_o \quad (3.27)$$

and the corresponding observability Gramians W_o becomes

$$W_o = \int_0^\infty e^{A^* t} H^* P_r P_r^T H e^{A t} dt \quad (3.28)$$

Thus, the following steps are performed in the BPOD method using outer projection.

1. Generate an ensemble of snapshots E and project these snapshots on observational space to get $\{H e_1, \dots, H e_s\}$.
2. Compute r dominant POD modes P_r of this matrix.
3. Integrate solution $z_1(t), \dots, z_r(t)$ of the adjoint system $\dot{z} = A^* z$ with initial conditions $z_k(0) = H^* P_k$. This will give the data matrix F .
4. Compute the SVD of $F^T E$ and the BPOD modes are given by (3.23) and (3.24).

Relation to POD

There are deep connections between the POD methods and the BPOD procedure explained here. BPOD may be viewed as bi-orthogonal decomposition instead of orthogonal decomposition given by POD. One of the difficulties with the POD method is that the inner product used for computing POD modes is arbitrary, specially for the compressible flow, the inner product is not obvious and different choices can give different results ([74]). BPOD may be viewed as a special case of POD using impulse responses and observability Gramians as an inner product. A useful consequence of using the observability Gramians as an inner product is that in this case the reduced order model preserves the stability of an equilibrium point at the origin, even if the full model is nonlinear ([73]). The balanced truncation of stable linear system is stable, but POD models of nonlinear systems may be unstable even if the nonlinear system is linearly stable at the origin ([79]).

3.3 Inverse modeling using reduced models

The discrete model for the evaluation of dynamical system from time t_i to time t_{i+1} can be described by an equation of the form

$$X(t_{i+1}) = M_i[X(t_i), \gamma] \quad (3.29)$$

where $X(t_{i+1}) \in \mathfrak{R}^n$ denotes the state vector at time t_{i+1} and γ is the vector of the uncertain parameters which needs to be determined. M_i is the nonlinear and deterministic dynamics operator that includes inputs. Suppose now that we have imperfect

observations $Y(t_i) \in \mathfrak{R}^{n^q}$ of the dynamical system (3.29), that are related to the model state at time t_i through

$$Y(t_i) = HX(t_i) + \eta(t_i) \quad (3.30)$$

where $H : R^n \rightarrow R^q$ being a linear observation operator that maps the model fields on observation space and $\eta(t_i)$ is an unbiased, random Gaussian error vector with covariance matrix R_i .

We assume that the difference between data and simulation results is only due to measurement errors and incorrectly prescribed model parameters. The problem of the estimation is then solved by directly minimizing the objective function J

$$J(\gamma) = \sum_i [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] \quad (3.31)$$

with respect to the parameters γ , satisfying the discrete nonlinear forecast model (3.29).

The efficient minimization of the objective function require the computation of the gradient of the objective function (3.31). The gradient vector (∇J) gives information about the direction (positive or negative) and the size of adjustments for each individual parameter. The adjoint method computes the exact gradient efficiently. The principle of the adjoint method is based on the systematic use of the chain rule of differentiation. Regardless of the number of parameters, the time required to compute the gradient using adjoint technique is more or less identical and is comparable with the computational time needed for a single simulation run of the nonlinear model (??). It requires one forward simulation with the original nonlinear model (3.29) and a second additional simulation backward in time with the adjoint model:

$$v(t_i) = \left(\frac{\partial M_i}{\partial X(t_i)} \right)^T v(t_{i+1}) - 2HR_i^{-1} [Y(t_i) - H(\mathbf{X}(t_i))] \quad (3.32)$$

where $v(t_i)$ represents the solution of the adjoint model. The gradient ∇J of the objective function J with respect to each component γ_k of the uncertain parameters vector γ is given by:

$$\nabla J_k = \sum_i - [v(t_{i+1})]^T \left[\frac{\partial M_i[X(t_i), \gamma]}{\partial \gamma_k} \right], k = \{1, \dots, n^p\} \quad (3.33)$$

The adjoint method is flexible as the number of parameters can easily be changed. The main hurdle in the use of adjoint method is its implementation, especially when the forward model contains nonlinearities. For the shallow water flow computations, the original model is very complicated and it is difficult to implement the adjoint for these type of models.

3.3.1 Linearization and reduced model formulation

The classical adjoint problem for a general model is a nonlinear constrained optimization problem which is difficult to solve. The problem can be simplified with the hypothesis that the objective function J can be made quadratic by assuming that the nonlinear dynamics operator M_i can be linearized. The linearization of nonlinear

high-order model (3.29) using the first order Taylor's formula around the background parameter γ^b gives

$$\Delta\bar{X}(t_{i+1}) = \frac{\partial M_i[X^b(t_i), \gamma^b]}{\partial X^b(t_i)} \Delta\bar{X}(t_i) + \sum_k \frac{\partial M_i[X^b(t_i), \gamma^b]}{\partial \gamma_k} \Delta\gamma_k \quad (3.34)$$

where \bar{X} is linearized state vector, X^b is the background state vector with the prior estimated parameters vector γ^b and $\Delta\bar{X}$ is a deviation of the model from background trajectory.

A model can be reduced if the incremental state $\Delta\bar{X}(t_{i+1})$ can be written as linear combination:

$$\Delta\bar{X}(t_i) = P\xi(t_{i+1}) \quad (3.35)$$

where $P = \{p_1, p_2, \dots, p_r\}$ is a projection matrix such that $P^T P = I_r$ and ξ is a reduced state vector given by

$$\xi(t_{i+1}) = \tilde{M}_i \xi(t_i) + \sum_k \frac{\partial \tilde{M}_i}{\partial \gamma_k} \Delta\gamma_k \quad (3.36)$$

or in matrix form

$$\begin{pmatrix} \xi(t_{i+1}) \\ \Delta\gamma \end{pmatrix} = \begin{pmatrix} \tilde{M}_i & \tilde{M}_i^\gamma \\ 0 & I \end{pmatrix} \begin{pmatrix} \xi(t_i) \\ \Delta\gamma \end{pmatrix} \quad (3.37)$$

Here $\Delta\gamma$ is the control parameter vector, \tilde{M}_i and \tilde{M}_i^γ are simplified dynamics operators which approximate the full Jacobians $\frac{\partial M_i}{\partial X^b}$ and $\frac{\partial M_i}{\partial \gamma_k}$ respectively:

$$\tilde{M}_i = P^T \frac{\partial M_i}{\partial X^b(t_i)} P \quad (3.38)$$

$$\tilde{M}_i^\gamma = P^T \left(\frac{\partial M_i}{\partial \gamma_1}, \dots, \frac{\partial M_i}{\partial \gamma_{n^p}} \right) \quad (3.39)$$

The Jacobian $\frac{\partial M_i}{\partial X^b}$, is obtained by approximating the nonlinear dynamics operator M_i by linearizing it with respect to background state X^b . Instead of computing this huge Jacobian by approximating the partial differential with finite difference by perturbing the nonlinear operator M_i in the direction of each node, we perturb along the direction of $p_h : h = \{1, \dots, r\}$ only

$$\frac{\partial M_i}{\partial X^b(t_i)} p_h = \frac{M_i[X^b(t_i) + \varepsilon p_h, \gamma^b] - M_i[X^b(t_i), \gamma^b]}{\varepsilon}, h = \{1, \dots, r\} \quad (3.40)$$

with ε being the size of the perturbation. The reduced dynamics operator \tilde{M}_i can now be computed by pre multiplying the above formula by P^T :

$$\tilde{M}_i = P^T \left(\frac{\partial M_i}{\partial X^b(t_i)} p_1, \dots, \frac{\partial M_i}{\partial X^b(t_i)} p_r \right) \quad (3.41)$$

Notice also that only the original model simulations are needed here. The reduced model requires less computational time as it simulates a reduced state within the dimension r instead of the original dimension n where $r < n$. The dimension on which the reduced model operates is $(r + n^p) \times (r + n^p)$ with n^p being the number of estimated parameters.

3.3.2 Collection of the snapshots and the reduced basis

Since the reduced model is used here to estimate uncertain parameters, the snapshots should be able to represent the behavior of the system for these parameters. Therefore the snapshot vectors $E_i \in \mathfrak{X}^s$ are the perturbed model simulations $\frac{\partial M_i}{\partial \gamma_k}$ with respect to each estimated parameter γ_k to get a matrix:

$$E = \{E_1, \dots, E_s\}; i = \{1, 2, \dots, s\} \quad (3.42)$$

The dimension of this ensemble matrix E is $s = u \times n^s$, where n^s is the number of snapshot collected for each individual parameter γ_k . The operators P used in the previous subsection are obtained from this ensemble of snapshot vectors by applying either POD or BPOD reduced order methodologies. In case of BPOD, we have an bi-orthogonal decomposition such that $P = T = S^{-1}$.

3.3.3 Approximate objective function and its adjoint

In reduced model approach, we look for an optimal solution of the (3.29) to minimize the approximate objective function (\hat{J}) in an incremental way:

$$\hat{J}(\Delta\gamma) = \sum_i \{[Y(t_i) - H(X^b(t_i))] - \hat{H}\xi(t_i, \Delta\gamma)\}^T R_i^{-1} \{[Y(t_i) - H(X^b(t_i))] - \hat{H}\xi(t_i, \Delta\gamma)\} \quad (3.43)$$

The value of the approximate objective function \hat{J} is obtained by correcting the observations $Y(t_i)$ for background state $X^b(t_i)$ which is mapped on the observational space through a mapping H and to the reduced model state $\xi(t_i, \Delta\gamma)$ which is mapped to the observational space through mapping \hat{H} , with $\hat{H} = HP$.

Since the reduced model has linear characteristics, it is easy to build an approximate adjoint model for the computation of gradient of the approximate objective function (3.43). The gradient of \hat{J} with respect to $\Delta\gamma$ is given by:

$$\frac{\partial \hat{J}}{\partial (\Delta\gamma)} = \sum_i -[\hat{v}(t_{i+1})]^T \frac{\partial \xi(t_{i+1})}{\partial (\Delta\gamma)} \quad (3.44)$$

where $\hat{v}(t_{i+1})$ is the reduced adjoint state variable. Once the gradient has been computed, the process of minimizing the approximate objective function \hat{J} is done along the direction of the gradient vector in the reduced space.

After the minimization process the initial parameters γ are updated and new set of updated parameters γ^{up} is obtained

$$\gamma^{up} = \gamma + \Delta\gamma \quad (3.45)$$

This process of minimization is repeated several times by constructing new reduced model with new set of updated parameters γ^{up} to get optimal parameters. Figure 3.1 presents the flowcharts of the parameter estimation procedure with classical adjoint method and two projection based model reduction methods.

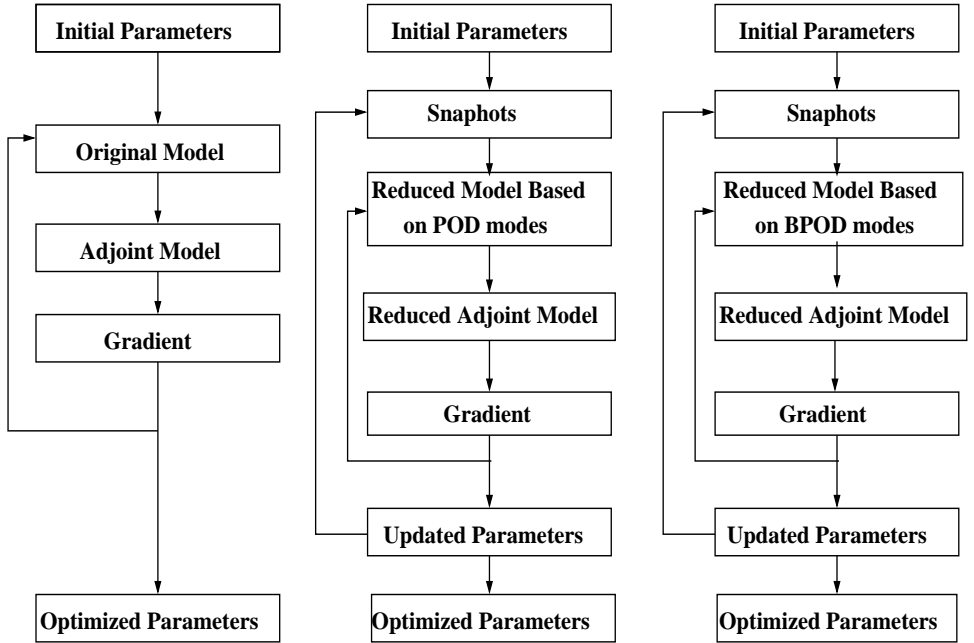


Figure 3.1: Flowcharts of the parameter estimation procedure with the classical adjoint method (left) and the reduced model approaches (middle (POD) and right (BPOD))

Convergence Criterion

To decide what will be optimal values of estimated parameter, we can use the criterion ρ . The outer iteration cycle is aborted when the terminal value of ρ is obtained

$$\rho = |J_{\beta_i} - J_{\beta_{i-1}}| \leq \kappa, \quad (3.46)$$

where β_i stands for the i^{th} outer iteration, κ is the terminal value. We have chosen $\kappa = 10^{-5}$ for all the numerical experiments.

3.4 Numerical experiments

3.4.1 The Model

We consider the advection-diffusion of concentration $c(x, t)$ for the transport of pollutant in two space dimensions. The evaluation of c gives

$$\frac{\partial c}{\partial t} + u \frac{\partial c}{\partial x} + v \frac{\partial c}{\partial y} = \frac{\partial}{\partial x} v_d \frac{\partial c}{\partial x} + \frac{\partial}{\partial y} v_d \frac{\partial c}{\partial y} + S \quad (3.47)$$

with the square domain $[0, d] \times [0, d]$, here v_d contains diffusion coefficient, $[u, v]$ is the velocity field and S , the source term. The experiments are performed on the 20×20

grid. In analyzing the system equation, we have assumed that the velocity field $[u, v]$ is known and constant with respect to time (see Figure 3.2) and the diffusion tensor is also constant. A solution to the partial differential equation is obtained by imposing boundary conditions and applying Euler time stepping to approximate the time derivative, the second derivatives are approximated with the central finite difference, while the upwind scheme is used for the first order spatial derivatives. Initially the concentration is zero for the whole model domain. A uniform source term is introduced at two grid points during the course of simulation.

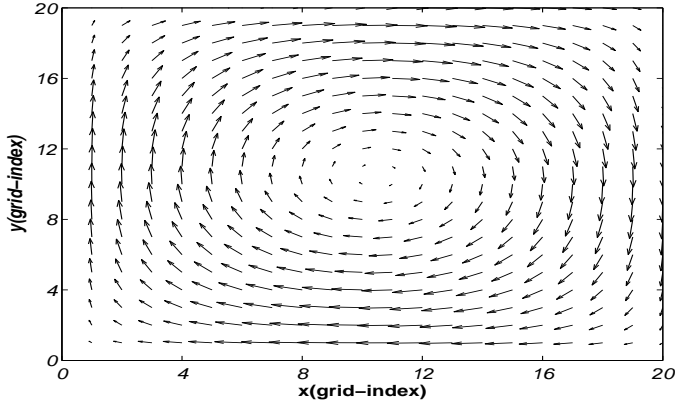


Figure 3.2: Velocity Field

3.4.2 Experiment 1

The POD based reduced model approach described above was used here to estimate the diffusion coefficient ν_d for the 2D advection diffusion model defined by 3.47. In experiment 1 three cases were considered to estimate ν_d according to the flow rate and complexity with respect to number of estimated parameters. The original model was simulated for 50 timesteps with $dt = 1s$ and a 20×20 grid with $dx = dy = 1m$ constitutes the model domain. The estimated parameters γ were effected by the parameter ν_d . A set of simulated observations $Y(t_i); i \in \{1, \dots, 50\}$ were obtained for each case.

Case 1:

The POD based model reduction method works well for diffusive systems [11]. Thus, diffusion was considered to be more dominant than advection in the first instance. The number of parameters estimated for this case were two. The numerical domain was divided into two regions for each γ_k . The true value for both the parameters was $\gamma^t = 0.18$. A set of simulated observations $Y(t_i); i \in \{1, \dots, 50\}$ with true values of the parameters were obtained at 2 different grid points one in each region of the model

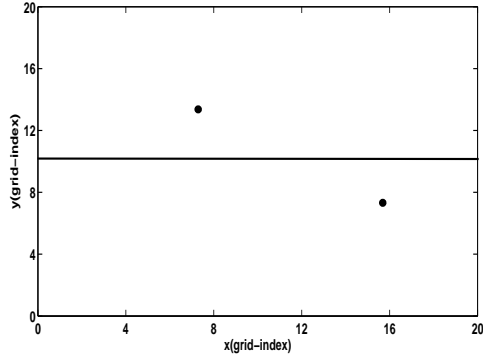


Figure 3.3: The 2D pollution model domain; The locations of the selected data points (circles) and straight line divides the model domain in two regions one for each estimated parameter

domain as shown in Figure 3.3. Initially $\gamma_1 = 0.12$ and $\gamma_2 = 0.20$. For the initial values of the parameters, a set consisting of 100 snapshot vectors was generated and we were able to form a basis consisting of only 8 dominant eigenvectors that captured 99.99% of the relative energy (see Figure 3.4). So a reduced model was built using these basis vectors which finally operates on the dimension \mathcal{X}^{8+2} .

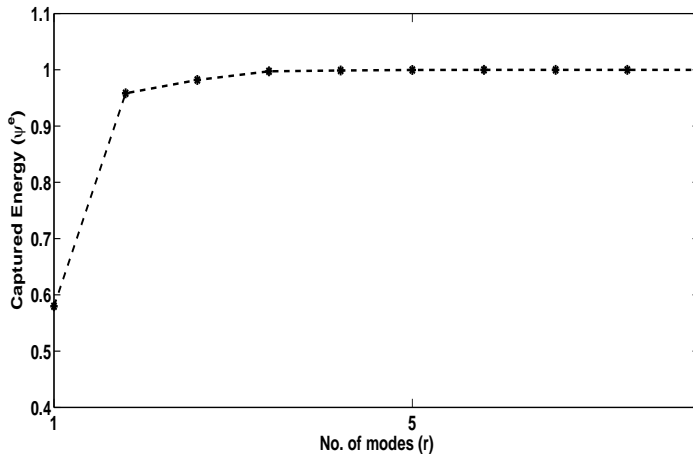


Figure 3.4: The POD modes captured energy for an ensemble of 100 snapshot vectors

With this reduced model approximate objective function \hat{J} was minimized and

new values of estimated parameters were found at $\gamma_1^{\mu p} = 0.1641$ and $\gamma_2^{\mu p} = 0.1944$. A significant improvement was observed in the objective function J which was reduced by more than 90% after the outer iteration β_1 . The minimization process was repeated several times with the new values of updated parameters γ_{up} by first constructing a new reduced model and then minimizing the approximate objective function \hat{J} in the reduced space. Table 3.1 shows the results of the minimization of two estimate variables.

(β)	γ_1	γ_2	J	\hat{J}	α
1	0.12	0.2	0.8434	0.0504	4
2	0.1641	0.1944	0.0865	0.0172	4
3	0.1751	0.1899	0.0167	0.0056	4
4	0.1830	0.1783	0.0035	4.5×10^{-4}	4
5	0.1811	0.1792	5.12×10^{-4}	9.11×10^{-5}	4
6	0.18039	0.17944	9.99×10^{-5}	3.79×10^{-5}	4
7	0.17998	0.17969	1.08×10^{-5}	4.77×10^{-6}	4

Table 3.1: The results of the estimation of two parameters γ_1 and γ_2 . Here α and β are the number of inner and outer iterations respectively.

With classical adjoint method, after **11** gradient computation, the objective function J reached a minimum value 7.7472×10^{-9} with $\gamma_1 = 0.18000121$ and $\gamma_2 = 0.18000219$.

Two separate reduced models were also constructed that captured 95% and 99% of the relative energy respectively. Using these reduced models, the approximate objective function \hat{J} was minimized in reduced space and the new values of the estimated parameters $\gamma^{\mu p}$ are found. Figure 3.5 shows the reduction of objective function J in the POD based calibration approach after each outer iteration β separately for each of the reduced models. It can be seen from Figure 3.5 that reduced model converged in all the cases but the rate of convergence is slow with the reduced models that captured 95% and 99% of the relative energy as compared to the reduced model that captured 99.99% relative energy. The basis for the reduced model must therefore be chosen intelligently.

Case 2:

To see the effect of dominant advection, the flow rate was increased in this case. The initial values for both the parameters were $\gamma_1 = 0.12$ and $\gamma_2 = 0.24$, and the true value for both parameters was $\gamma^t = 0.18$. A set of simulated observations $Y(t_i)$; $i \in \{1, \dots, 50\}$ with true values of the parameters were obtained at 9 different grid points of the model domain as shown in Figure 3.6.

For the above initial parameters values, a reduced model was constructed that captured 99.99% of the relative energy with only 10 basis vectors. With this reduced model of dimension \mathfrak{R}^{10+2} , the approximate objective function \hat{J} was minimized and new values of the estimated parameters were found at $\gamma_1^{\mu p} = 0.1773$ and $\gamma_2^{\mu p} = 0.1974$.

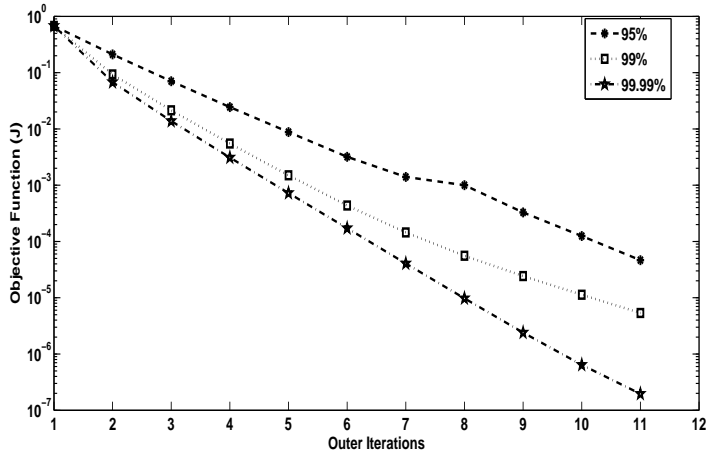


Figure 3.5: The value of the objective function J at successive outer iterations for reduced models with different relative energy attained

A significant improvement was again observed in the objective function J , which was again reduced by more than 90% after the outer iteration β_1 . The process was repeated several times and updated values of the estimated parameters were used as initial values to construct the reduced model in each outer iteration β . Table 3.2 shows the minimization results and the convergence of two estimated parameters. It is clear from the results that although the flow rate and the observation points were increased, the estimated parameters γ_1 and γ_2 converged to true parameter value $\gamma^j = 0.18$ with similar efficiency.

(β)	γ_1	γ_2	J	\hat{J}	α
1	0.12	0.24	20.8776	1.7673	4
2	0.1773	0.1974	1.2699	0.0894	4
3	0.1787	0.1848	0.1061	0.0070	4
4	0.1795	0.1812	0.0072	4.49×10^{-4}	4
5	0.17982	0.18027	4.28×10^{-4}	2.56×10^{-5}	4
6	0.17993	0.18006	2.59×10^{-5}	2.21×10^{-6}	4

Table 3.2: Results of the estimation of two parameters γ_1 and γ_2 with the reduced model that captured 99.99% of the relative energy.

Case 3:

The validity of the reduced model also depends on the number of estimated parameters. The parameters estimated in this case were four. The numerical domain was

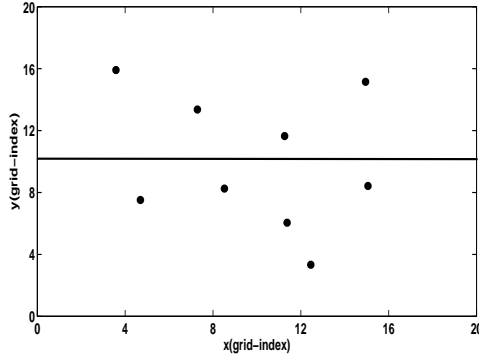


Figure 3.6: The locations of the selected data points (circles) and straight line divides the model domain in two regions one for each estimated parameter

divided into four regions for each γ_k . The true value for all the four parameters was $\gamma^t = 0.18$. A set of simulated observations were obtained with true values of the parameters at 12 different grid points in the model domain as shown in Figure 3.7. The initial values for the estimated parameters were $\gamma_1 = 0.12$, $\gamma_2 = 0.20$, $\gamma_3 = 0.14$ and $\gamma_4 = 0.22$.

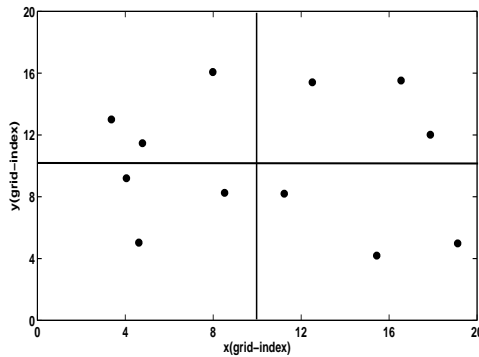


Figure 3.7: The locations of the selected data points (circles) and the four subdomains used used in case 3 to estimate four parameters

As in the previous cases, the initial values of the estimated parameters *gamma* were used to collect a set of 200 snapshot vectors and a basis consisting of 6 dominant eigenvectors was used to construct a reduced model that captured 98% of the relative energy (see Figure 3.8(a)). Here the reduced model was built using only few dominant modes to get the initial direction of the update of the estimated parameters γ and once

the direction is known the reduced model can be built using updated parameters γ_{up} with more relative energy attained for fast convergence.

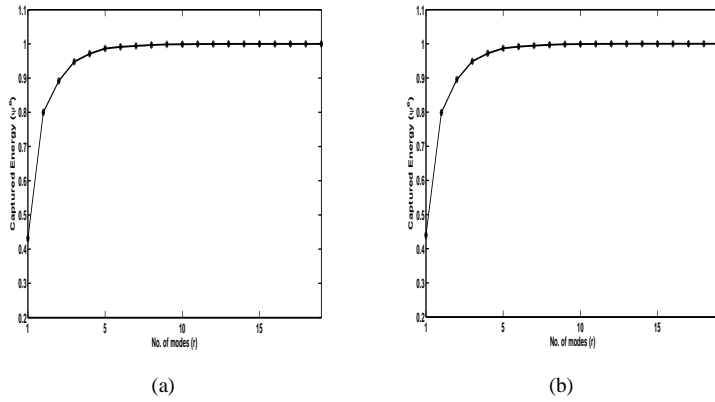


Figure 3.8: The POD modes captured energy for an ensemble of 200 snapshots in the outer iterations a) β_1 b) β_2

With this reduced model of dimension \mathfrak{R}^{6+4} , the approximate objective function \hat{J} was minimized and new values of the estimated parameters were found at $\gamma_1 = 0.176$, $\gamma_2 = 0.188$, $\gamma_3 = 0.164$ and $\gamma_4 = 0.206$. Although only 6 dominant POD modes were used to build the reduced model, the objective function J was reduced by more than 75% after the outer iteration β_1 .

The updated parameters obtained after outer iteration β_1 were used again to construct a new reduced model that captured 99.99% of the relative energy with 15 basis vectors (see Figure 3.8(b)). Table 3.3 shows the minimization results and the convergence of four estimated parameters. The results from Table 3.3 shows that the POD based estimation procedure efficiently minimizes the objective function J and all the estimated parameters converge to true parameter value $\gamma^t = 0.18$ after few outer iterations.

β	γ_1	γ_2	γ_3	γ_4	J	\hat{J}
1	0.12	0.20	0.14	0.22	4.3277	0.6073
2	0.176	0.188	0.164	0.206	0.9485	0.1101
3	0.18285	0.185388	0.17701	0.188588	0.1637	0.0227
4	0.185666	0.182784	0.180184	0.184586	0.0446	0.0027
7	0.180158	0.180092	0.180164	0.18014	6.5×10^{-5}	4.1×10^{-6}

Table 3.3: Results of the estimation of four parameters γ_1 , γ_2 , γ_3 and γ_4 with the POD based estimation procedure.

3.4.3 Experiment 2

In this experiment we have applied the BPOD procedure to get the basis of reduced model and compare results with POD based reduced-order model. In experiment 2 two cases were considered to estimate u_d with dominant advection as in the 2nd case of Experiment 1 according complexity with respect to the number of observations.

Case 1

The number of parameters estimated for this case were two. The numerical domain was divided into two regions for each γ_k . The true value for both the parameters was again $\gamma^t = 0.18$. A set of simulated observations $Y(t_i)$; $i \in \{1, \dots, 50\}$ with true values of the parameters were obtained at 20 different grid points scattered in the model domain as shown in Figure 3.9. Initially $\gamma_1 = 0.12$ and $\gamma_2 = 0.20$. For the initial

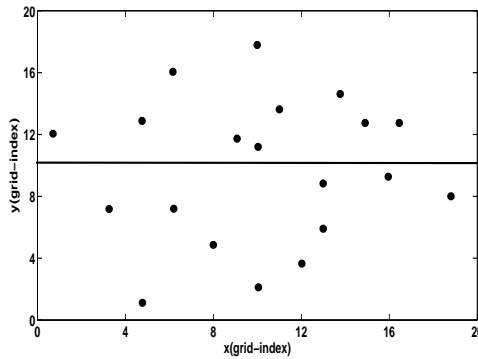


Figure 3.9: The 2D pollution model domain; The locations of the selected data points (circles) and straight line divides the model domain in two regions one for each estimated parameter

values of the parameters, a set consisting of 100 snapshot vectors was generated and the BPOD method was applied to get reduced basis for the reduced model. The first 10 balancing modes were used to construct the reduced model. With this reduced model of dimension \mathfrak{R}^{10+2} , the approximate objective function \hat{J} was minimized and new values of the estimated parameters were found at $\gamma_1^{up} = 0.19$ and $\gamma_2^{up} = 0.194$. The process was repeated several times and updated values of the estimated parameters were used as initial values to construct the reduced model in each outer iteration β .

Figure 3.10 shows the minimization of objective function J for BPOD and POD reduced models of order \mathfrak{R}^{10+2} . The Figure 3.10 demonstrates that the optimization using BPOD based reduced model performs efficiently in the first few outer iterations β of the minimization process as compared to the POD based reduced model. Moreover, the same minimum value of the objective function J is achieved with both the reduced order models after β_8 of the minimization process.

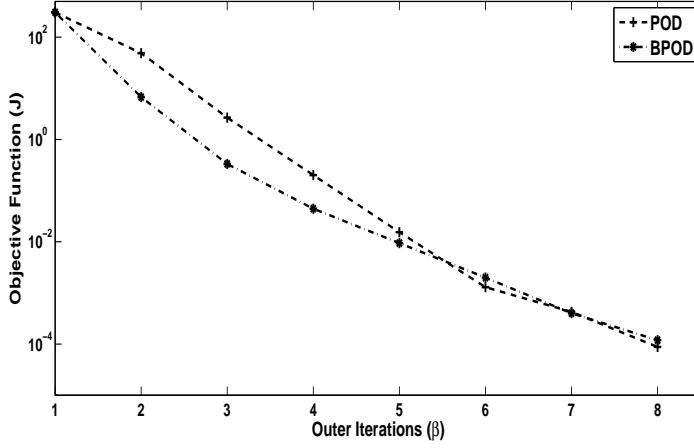


Figure 3.10: The successive outer iterations β of the objective function (J) for the POD and the BPOD based reduced order models of order \mathfrak{X}^{10+2}

Case 2

The only difference here from the previous case is the number of selected observations. The experiment was designed to compare the results from the BPOD method with and without outer projection procedure. Since outer projection procedure is useful when the number of outputs (observations) are large, therefore a set of simulated observations $Y(t_i); i \in \{1, \dots, 50\}$ with true values of the parameters were obtained at every other grid point in the model domain. So in total 200 observation points were selected.

To quantify the performance of both the reduce models we have used the following metric, i.e. relative root mean square error (RRMSE) [100] of the difference between the full order and the reduced order simulation. This was obtained by first taking the full order approximation results and the corresponding reduced order results within the inner iteration of the variational data assimilation.

$$RRMSE = \frac{1}{m} \sum_{i=1}^m \sqrt{\frac{\|Y(t_i) - \hat{Y}(t_i)\|^2}{\|Y(t_i)\|^2}} \quad (3.48)$$

where m is the number of time steps. Table 3.4 shows the comparison of the RRMSE in the full and reduced order approximations obtained by using the POD method, the BPOD method and the BPOD method with outer projection procedure respectively with respect to the size of the reduced order models. Here, the outer projection procedure was used by selecting a projection matrix P_r of 12 dominant POD modes as explained in Section 3.2.3. So instead of 200 simulations with the adjoint system, only 12 simulations with the adjoint system were needed to compute the balancing modes. The Table 3.4 demonstrates that there are no significant differences in the RRMSE errors for the BPOD with and without outer projection. Although the BPOD method

performs better as compared to the POD method, this difference is not significant if number of modes selected for the reduced model are more than 10.

(Modes)	POD	BPOD	OP(12)
2	0.5071	0.3658	0.3728
3	0.4965	0.3617	0.3704
4	0.2191	0.1383	0.1388
5	0.2197	0.1375	0.1380
6	0.1187	0.1012	0.1009
7	0.1178	0.0947	0.0995
8	0.0845	0.0777	0.0766
9	0.0717	0.0659	0.0660
10	0.0534	0.0527	0.0530
15	0.0267	0.0258	0.0257
20	0.0267	0.0247	0.0247
25	0.0225	0.0209	0.0213

Table 3.4: RRMSE in the forward simulation with different model reduction methods

Figure 3.11 shows the minimization of the objective function J with the outer projection and the POD based reduced models of order \mathcal{R}^{10+2} . The Figure 3.11 demonstrates that the optimization based on both types of reduced model performs efficiently and converge to almost similar value of the objective function J in each outer iteration β . This means that although lots of effort had been made in constructing a reduced model using outer projection procedure but the results of the optimization with the POD method and the BPOD method are the same.

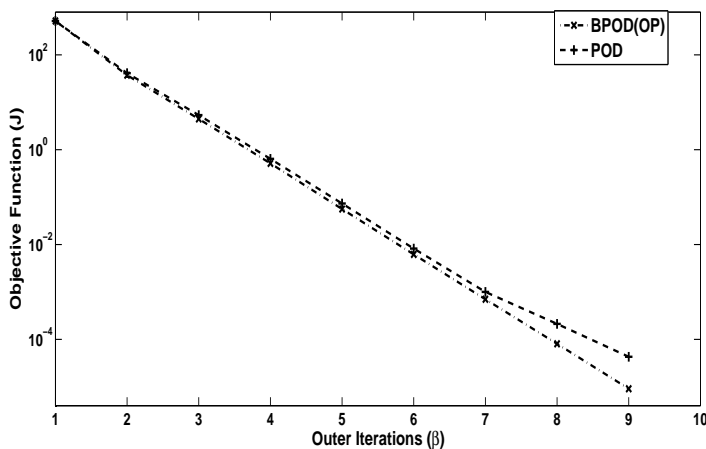


Figure 3.11: The successive outer iterations β of the objective function (J) for the POD and the outer projection based reduced order models of order \mathcal{R}^{10+2}

3.5 Conclusions

In variational data assimilation it is necessary to implement the adjoint model for the computation of gradient of the objective function J which is difficult and laborious for large scale real time systems. In this work we have applied a method which simplifies this problem using a projection based POD model reduction approach. The method approximate the full dynamical system while retaining its properties. An ensemble of forward model simulations is used to determine the approximation of the covariance matrix of the model variability and the dominant eigenvectors of this matrix are used to define a model subspace. An approximate linear reduced model is obtained by projecting the original model onto this reduced subspace. The adjoint of the tangent linear model is replaced by the adjoint of this linear reduced forward model. The minimization process is carried out in reduced subspace and hence reduces the computational costs. Numerical experiments were performed in a pollution model of concentration $c(x,t)$ with the POD based estimation method to estimate diffusion coefficients. The results demonstrates that the POD based estimation approach successfully estimate the diffusion coefficient for both advection dominated problems as for diffusion dominated problems.

BPOD is a model reduction method which combines the ideas and methodology of two model reduction methods, i.e. POD and balanced truncation. The balanced truncation method uses both controllable and observable subspaces while obtaining a low rank subspace for the reduced model. A state transformation is chosen such that controllability and observability Gramians are diagonal and equal. The BPOD method presented here allows computation of the balancing transformation directly from snapshots of empirical Gramians using SVD, without needing to compute the Gramians themselves. The method also allows the empirical observability Gramian to be computed efficiently even when the number of outputs is large.

Identical twin experiments were performed to estimate diffusion coefficients in a pollution model with both POD and BPOD methods. The results demonstrated that the reduced model obtained from BPOD performs better than POD based reduced model in the beginning of the minimization but both the methods converged to similar value after 6th outer iteration of the minimization process. An experiment was also performed to compare the two BPOD procedures when the number of outputs is large. We were able to form a reduced model with similar accuracy with 12 simulations of the adjoint system using outer projection procedure instead of 200 simulations with the adjoint system using usual BPOD method. Although lot of efforts had been put in constructing a reduced order model by BPOD method, the minimization results demonstrated that both the POD and the BPOD methods performed similarly.

Preliminary results shows the validity of the POD based model reduction methods for parameter estimation and now the method needs to be tested for large scale application which is the topic of the next chapter.

Inverse shallow water flow modeling using model reduction ¹

4.1 Introduction

Variational data assimilation has often been used for model calibration (e.g. [87], [53], [92], [43]). The method aims at adjusting a number of unknown parameters on the basis of given data. One first defines a scalar function which, for any model solution over the assimilation interval, measures the “distance” or “misfit” between that solution and the available observations. The so-called *objective (cost) function* is typically a sum of squared differences between the observations and the corresponding model values. One then looks for the model solution that minimizes the objective function. To obtain a computationally efficient procedure this objective function is minimized with a gradient-based algorithm where the gradient is determined by solving the adjoint problem. Variational data assimilation usually requires the implementation of an adjoint model. Research has recently been carried out on automatic generation of computer code for the adjoint and adjoint compilers have now become available (see [47]). Even with the use of these compilers coding an adjoint is a laborious programming effort that hampers new applications of the method.

In the last years the studies of complex systems have taken strong advantage of the development of mathematical methods coming from the theory of non-linear dynamical systems. Proper orthogonal decomposition (POD) is a model reduction method considered as an application of the singular value decomposition (SVD) to the approximation of general dynamical systems [11]. POD is the technique leading to the system of low-dimensional ordinary differential equations that approximate the model formulated in terms of partial differential equations, allowing to take full advantage of dynamical systems theory. The method was originally developed by Karl Pearson (see [68]) and has application in many fields like image processing, signal processing, data compression, oceanography, chemical engineering and fluid mechanics (see e.g.

¹This chapter is a slightly revised version of [5]

[49], [69], [3], [14]). For a detailed description on POD scope and research see [40].

Another application of POD in operational oceanography and weather forecasting is four dimensional variational data assimilation (4DVAR), which is based on an optimal control theory [88]. POD has recently been applied successfully in 4DVAR (e.g. [54], [22], [32], [31]). Previous experiences show that the POD is valuable in parameter estimation (e.g. [26]), especially the recent work by [102] shows that POD based model reduction technique can be successfully applied for inverse modeling of 3D groundwater flow.

In the POD model reduced approach, an ensemble of snapshot vectors of forward model simulations is used to determine an approximation of the covariance matrix of the model variability and a small number of the leading eigenvectors of this matrix is used to define a model subspace. By projecting the original model onto this subspace an approximate linear reduced model is obtained. Once the reduced model is available its adjoint can be implemented easily and the minimization problem is solved completely in reduced space with very low computational cost. If necessary this process of minimization is repeated several times by generating a new set of snapshots (ensemble) which is closer to the new estimated parameters.

Compared with the classical variational method the adjoint of the tangent linear model is replaced by the adjoint of the linear reduced forward model. While the adjoint of the tangent linear approximation of the original model produces the exact local gradient, the reduced order approach is based on statistically linearised model and hence produces an averaged gradient. As a result the model reduced approach can be less sensitive for local minima for certain applications [99].

The present work is mainly inspired by the recent work done in reduced order modeling by [99], [54]. These methods are based on deriving approximate low-order data assimilation system in the context of incremental 4DVAR procedure for parameter and state estimation respectively. In the present work we consider 1) a new application of reduced-order calibration approach. The method has been used for a model based on shallow water equations which is different in behaviour from the one used in [99]. We also consider 2) usage of extended time horizon for parameter estimation. The generation of an ensemble involves running the forward model several times. The computational cost of the method is dominated by the generation of this ensemble. In this study it is found that if the dynamics of the system does not change significantly then a smaller simulation period can be chosen to generate an ensemble of forward model simulations for an optimization problem over larger period.

A shallow water model of the continental shelf, the Dutch continental shelf model (DCSM) is used in the Netherlands to forecast the storm surges in the North Sea. Accurate predictions of storm surges are of vital importance to the Netherlands. The decision whether or not to close the storm surge barriers is based on these predictions. A number of twin experiments is performed with DCSM to evaluate the performance of proposed approach. This allows us to evaluate the results by comparing them to the truth.

The chapter is organized as follows. Section 4.2 explains classical inverse modeling methods. Procedure required for the construction of POD projection based reduced model is described in Section 4.3. In Section 4.4 the methodology of POD projection based reduced method for the calibration is explained. Section 4.5 contains

results from the twin experiments with the operational model for storm surge prediction, the DCSM, to estimate the water depth. The chapter concludes in Section 4.6 by discussing the results.

4.2 Inverse modeling

Consider the data assimilation problem for a general nonlinear dynamical system. The discrete system equation for the state vectors $X(t_{i+1}) \in \mathfrak{X}^n$ is given by;

$$X(t_{i+1}) = M_i[X(t_i), \gamma], \quad (4.1)$$

where M_i is nonlinear and deterministic dynamics operator that includes inputs and propagates the state from time t_i to time t_{i+1} , γ is vector of uncertain parameters which needs to be determined. Suppose now that we have imperfect observations $Y(t_i) \in \mathfrak{X}^q$ of the dynamical system (4.1) that are related to model state at time t_i through

$$Y(t_i) = HX(t_i) + \eta(t_i), \quad (4.2)$$

where $H : \mathfrak{X}^n \rightarrow \mathfrak{X}^q$ is linear observation operator that maps the model fields on observation space and $\eta(t_i)$ is unbiased random Gaussian error vector with covariance matrix R_i .

The idea of parameter estimation is to identify the values of uncertain model parameters γ . We assume that the difference between data and simulation results occurs only due to measurement errors and incorrectly prescribed model parameters. A most commonly used measure that determines this difference is the weighted sum of squared residuals. The problem of estimation is then solved by directly minimizing the cost function J

$$J(\gamma) = \sum_{i=1}^m [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))], \quad (4.3)$$

with respect to the parameters γ , satisfying the discrete nonlinear forecast model (4.1).

The minimization of the cost function J is often based on quasi-Newton methods. These methods require the computation of the gradient of the cost function. The gradient vector ∇J gives information about the direction (positive or negative) and the size of adjustments for each individual parameter. The adjoint method [18] computes the exact gradient efficiently. The principle of the adjoint method is based on the systematic use of the chain rule of differentiation. Regardless of the number of parameters, the time required to compute the gradient using adjoint technique is more or less identical and is comparable with the computational time needed for a single simulation run of the nonlinear model (4.1). It requires one forward simulation with the original the nonlinear model (4.1) and a second additional simulation backward in time with the adjoint model:

$$v(t_i) = \left(\frac{\partial M_i}{\partial X(t_i)} \right)^T v(t_{i+1}) - 2HR_i^{-1}[Y(t_i) - H(X(t_i))], \quad (4.4)$$

where $v(t_i)$ represents the solution of the adjoint model. The gradient ∇J of the cost function J with respect to each component $\gamma_k, k = \{1, \dots, n^p\}$, of the uncertain parameter vector γ is given by

$$\nabla J_k = \sum_{i=1} - [v(t_{i+1})]^T \left[\frac{\partial M_i[X(t_i), \gamma]}{\partial \gamma_k} \right]. \quad (4.5)$$

The main hurdle in the use of adjoint method is its implementation. Even with the use of adjoint compilers that have become available these days this is a huge programming effort that hampers new applications of the method. Moreover, the adjoint equation needs to be integrated backward in time and therefore the states of the forward model have to be stored at each grid point for all time steps. The memory access will therefore be huge for large scale problems.

4.3 Reduced order modeling

The problem of obtaining a lower-dimensional approximation to a high-dimensional system is known as model reduction. The method used in this paper fall in the category of projection methods, which involve projecting the system equations onto a subspace of the original phase space.

4.3.1 Proper Orthogonal Decomposition

The proper orthogonal decomposition, also known as principle components analysis, has been widely used for a broad range of applications. POD analysis yields a set of empirical eigenfunctions which describes the dominant behavior or dynamics of the given problem. POD can be described as the method that examines the linear relationship between variables with the aim of reducing the dimensionality of the problem.

The main idea is the following. Given a set of data that lies in a vector space V , find a subspace V_r of fixed dimension r such that the error in the projection onto the subspace is minimized. We start by collecting the set of s snapshots of some physical process. Each sample of snapshots X_i which is defined on a set of n nodes X stands for an n dimensional vector X_j . i.e.,

$$X_i = \{X_{1i}, X_{2i}, \dots, X_{ni}\}', i \in \{1, 2, \dots, s\}, \quad (4.6)$$

The elements within a snapshot represent the signal for a specific location in the model, possibly for multiple quantities. Define the vector X^b of background state and correct each snapshot vector so that

$$E_i = X_i - X^b, i \in \{1, 2, \dots, s\}, \quad (4.7)$$

These corrected snapshots are arranged in matrix E , which denotes the new ensemble. The covariance matrix Q can be constructed from the ensemble E of the snapshots by taking the outer product

$$Q = EE^T, \quad (4.8)$$

The dimension n often exceeds 10^4 , so direct solution of eigenvalue problem is not feasible. To shorten the calculation time necessary for solving the eigenvalue problem for this high dimensional covariance matrix, we define a covariance matrix G as the inner product. In the method of snapshots [78], one then solves $s \times s$ eigenvalue problem

$$GZ_i = E^T E Z_i = \lambda_i Z_i, i \in \{1, 2, \dots, s\}, \quad (4.9)$$

where λ_i are the eigenvalues of the above eigenvalue problem. The eigenvectors Z_i may be chosen to be orthonormal and the POD modes p_i are then given by

$$p_i = EZ_i / \sqrt{\lambda_i}. \quad (4.10)$$

A physical explanation of POD modes is that they maximize the average energy in the projection of data onto subspace spanned by the modes. The eigenvalues λ_i provide a measure ψ_i for the relative energy associated with the corresponding POD modes p_i :

$$\psi_i = \frac{\lambda_i}{\sum_{l=1}^s \lambda_l} \cdot 100\%, i = \{1, 2, \dots, s\} \quad (4.11)$$

We collect p_r ($r < s$) modes such that $\psi_1 > \psi_2 > \dots > \psi_r$ and they totally explain at least the required variance ψ^e :

$$\psi^e = \sum_{l=1}^r \psi_l. \quad (4.12)$$

4.4 Inverse modeling using reduced models

An approximate linear reduced model in variational data assimilation presented here is based on the principle of POD model reduction technique. An ensemble of snapshot vectors is generated from the original model. The reduced model operates on the space defined by the dominant eigenvectors of the generated ensemble.

4.4.1 Linearization and reduced Basis

Linearization of non-linear high-order model (4.1) with respect to parameter k gives

$$\bar{X}(t_{i+1}) = M_i[X^b(t_i), \gamma^b] \bar{X}(t_i) + \sum_{k=1} \frac{\partial M_i[X^b(t_i), \gamma^b]}{\partial \gamma_k} \Delta \gamma_k, \quad (4.13)$$

with \bar{X} is linearized state, X^b is the background state for which the corresponding parameter γ^b are linearized. The partial derivatives $\frac{\partial M}{\partial \gamma_k}$ can be computed using perturbation method with respect to each parameter γ_k

$$\frac{\partial M_i}{\partial \gamma_k} \approx \frac{M_i[X^b(t_i), \gamma_k^b + \Delta \gamma_k] - M_i[X^b(t_i), \gamma_k^b]}{\Delta \gamma_k}, \quad (4.14)$$

where $\Delta \gamma_k$ is the perturbation. The snapshot vectors $E_i \in \mathfrak{R}^s$ are the perturbed model simulations $\frac{\partial M}{\partial \gamma_k}$ with respect to each parameter γ_k . These snapshots are collected in

matrix $E = [E_1, \dots, E_s]$. We then simplify the eigenvalue problem as explained in Section 4.3 to obtain the POD basis (modes) $P = [p_1, \dots, p_r]$ of r dominant eigenvectors. The total number of eigenmodes r in the basis P depends on the required accuracy of the reduced model.

4.4.2 Reduced model formulation

A model can be reduced if state $\hat{X}(t_i)$ can be written as linear combination

$$\hat{X}(t_{i+1}) = X^b(t_{i+1}) + P\xi(t_{i+1}), \quad (4.15)$$

where \hat{X} is the approximate linearized state, ξ is a reduced time-varying state vector given by

$$\begin{pmatrix} \xi(t_{i+1}) \\ \Delta\gamma \end{pmatrix} = \begin{pmatrix} \tilde{M}_i & \tilde{M}_i^\gamma \\ 0 & I \end{pmatrix} \begin{pmatrix} \xi(t_i) \\ \Delta\gamma \end{pmatrix} \quad (4.16)$$

, here \tilde{M}_i and \tilde{M}_i^γ are simplified dynamics operators which approximate the full Jacobians $\frac{\partial M_i}{\partial X^b}$ and $\frac{\partial M_i}{\partial \gamma_k}$ respectively:

$$\tilde{M}_i = P^T \frac{\partial M_i}{\partial X^b(t_i)} P, \quad (4.17)$$

$$\tilde{M}_i^\gamma = P^T \left(\frac{\partial M_i}{\partial \gamma_1}, \dots, \frac{\partial M_i}{\partial \gamma_{n^p}} \right). \quad (4.18)$$

The Jacobian $\frac{\partial M_i}{\partial X^b}$ is obtained by linearizing the nonlinear dynamics operator M_i with respect to background state X^b . Instead of computing this huge Jacobian with finite difference method (i.e. by perturbing the nonlinear operator M_i in the direction of each node), we perturb along the direction of POD modes only:

$$\frac{\partial M_i}{\partial X^b(t_i)} P_h = \frac{M_i[X^b(t_i) + \varepsilon p_h, \gamma^b] - M_i[X^b(t_i), \gamma^b]}{\varepsilon} \quad (4.19)$$

with ε being the size of the perturbation. The reduced dynamics operator \tilde{M}_i can now be computed by premultiplying the above formulae by P^T

$$\tilde{M}_i = P^T \left(\frac{\partial M_i}{\partial X^b(t_i)} p_1, \dots, \frac{\partial M_i}{\partial X^b(t_i)} p_r \right). \quad (4.20)$$

The dimension of the subspace on which reduced model operates depends on the number of eigenmodes r selected in the POD basis and the number of estimate variables n^p i.e. $(r + n^p) \times (r + n^p)$.

4.4.3 Approximate objective function and its adjoint

The value of the approximate objective function \hat{J} is obtained by correcting the observations $Y(t_i)$ for the background state $X^b(t_i)$ which is mapped on the observational

space through a mapping H and for the reduced model state $\xi(t_i)$ which is mapped on the observational space through mapping \hat{H} with $\hat{H} = HP$:

$$\hat{J}(\Delta\gamma) = \sum_{i=1}^m [\{Y(t_i) - H(X^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)]^T R_i^{-1} [\{Y(t_i) - H(X^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)]. \quad (4.21)$$

As the dimension of the reduced model is smaller than that of original model and reduced model has linear characteristics, it is easy to build an approximate adjoint model for the computation of gradient of the approximate objective function (4.21). The gradient of \hat{J} with respect to parameter $\Delta\gamma$ is given by:

$$\frac{\Delta\hat{J}}{\Delta\gamma} = \sum_i -[\hat{v}(t_{i+1})]^T \frac{\partial \xi(t_{i+1})}{\partial \Delta\gamma}, \quad (4.22)$$

where $\hat{v}(t_{i+1})$ is the reduced adjoint state variable (see Appendix A). Once the gradient is computed, the process of minimizing the approximate objective function \hat{J} is performed along the direction of the gradient vector in the reduced space.

4.4.4 Procedural flow with reduced model

In order to perform the whole parameter estimation process the following steps are executed.

1. Outer Iteration.

- Background initial parameters γ^b are used to generate an ensemble of forward model simulations.
- A POD reduced model and its adjoint model are established using this ensemble.
- Inner Iteration. Perform optimization iterations in reduced space to obtain the optimal solution of the approximate objective function \hat{J} .
- The background initial parameters γ^b are updated and new set of initial parameters ($\gamma^{\mu p}$) is obtained.

$$\gamma^{\mu p} = \gamma^b + \Delta\gamma. \quad (4.23)$$

- ##### 2. Return to step 1 with new set of initial parameters until optimality condition is achieved.

Convergence criterion for inner and outer iterations

We have defined two criteria both for the inner and outer iterations of the optimization process. We stop the present inner iteration and switch to a new outer iteration with new set of parameters following the criterion μ , which is defined as

$$\mu = \sum_{k=1} \|\nabla \hat{J}_{\alpha_k}\| / \|\nabla \hat{J}_{\alpha_1}\| \leq \delta, \quad (4.24)$$

where $\nabla \hat{J}_{\alpha_1}$ is the value of the gradient at start of inner iteration, $\nabla \hat{J}_{\alpha_i}$ is the value of the gradient after each inner iteration. The value of δ is chosen considering that the gradient should decrease by at least three orders of magnitude from the initial gradient [15]. The outer iteration cycle is said to converge when the optimal value α of the objective function J is achieved:

$$\rho = \|J_{\beta_i} - J_{\beta_{i-1}}\| \leq \kappa, \quad (4.25)$$

where β_i is the i^{th} outer iteration. We have chosen $\kappa = 0.5$ for all the numerical experiments.

4.4.5 Computational cost

The computational costs of the reduced model approach are dominated by the generation of the ensemble of forward model simulations. If the dynamics of the system does not change significantly during the course of simulation then a smaller simulation period can be chosen for the generation of ensemble. Using this ensemble the optimization problem can then be solved over the whole period of model simulation. The efficiency of optimization process is also influenced by the ensemble size. A large ensemble size leads to a huge eigenvalue problem. It is possible to include only those snapshots in the ensemble where data is available.

The method needs to be updated in each outer iteration β by constructing a new POD model by generating a new ensemble of forward model simulations. The number of outer iterations β is influenced by the chosen abortion criterion κ . It should not be chosen too small as this causes jumping of objective function J since it is possible that reduced model overestimates γ_k due to the process of linearisation.

4.5 Application: The Dutch Continental Shelf Model

The Dutch continental shelf model (DCSM) is an operational storm surge model used in the Netherlands for real-time storm surge prediction in North Sea. Accurate predictions of the storm surges are of vital importance to the Netherlands since large areas of the land lie below sea level. Accurate forecasts at least six hours ahead are needed for proper closure of the movable storm surge barriers in Eastern Scheldt and the New Waterway. The governing equations used in DCSM are the non-linear 2-D shallow water equations. The shallow water equations, which describe large scale water motions, are used to calculate the movements of the water in the area under consideration. These equations are

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + g \frac{\partial h}{\partial x} - f v + \frac{g u \sqrt{u^2 + v^2}}{H C_{2D}^2} = \frac{1}{\rho_w} \frac{\tau_x}{H} - \frac{1}{\rho_w} \frac{\partial p_a}{\partial x}, \quad (4.26)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + g \frac{\partial h}{\partial y} + f u + \frac{g v \sqrt{u^2 + v^2}}{H C_{2D}^2} = \frac{1}{\rho_w} \frac{\tau_y}{H} - \frac{1}{\rho_w} \frac{\partial p_a}{\partial y}, \quad (4.27)$$

$$\frac{\partial h}{\partial t} + \frac{\partial H u}{\partial x} + \frac{\partial H v}{\partial y} = 0, \quad (4.28)$$

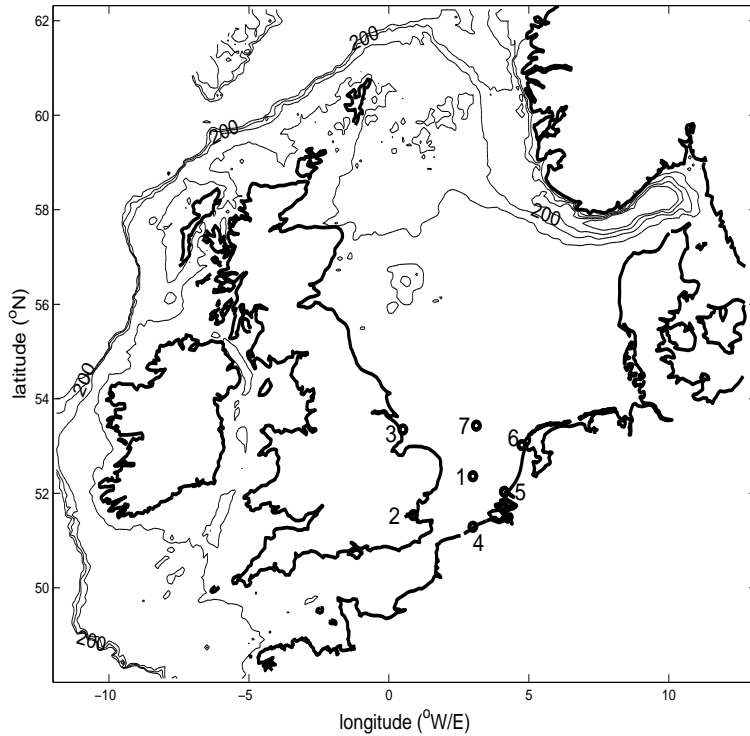


Figure 4.1: DCSM area and assimilation stations: 1. N51, 2. Southend, 3. Innerdowsing, 4. Oostende, 5. H.v.Holland, 6. Den Helder, 7. N4

where

- x, y = Cartesian coordinates in horizontal plane
- t = time coordinate
- u, v = depth-averaged current in x and y direction respectively
- h = water level above reference plane
- D = water depth below the reference plane
- H = total water depth ($D+h$)
- f = coefficient for the Coriolis force
- C_{2D} = 2D Chezy coefficient
- τ_x, τ_y = wind stress in x and y direction respectively
- ρ_w = density of sea water
- p_a = atmospheric pressure
- g = acceleration of gravity

These equations are discretized using an alternating directions implicit (ADI) method and the staggered grid that is based on the method by [55] and improved by [82]. In the implementation the spherical grid is used instead of rectangular (see e.g. [94]). Boundary conditions are applied at both closed and open boundaries. At closed boundaries the velocity normal to the boundary is zero. So no inflow and outflow can occur through these boundaries. At open boundaries the water level is described in terms of ten harmonic components ($M_2, S_2, N_2, K_2, O_1, K_1, Q_1, P_1, U_2, L_2$) as follows

$$h(t) = h_0 + \sum_{j=1}^{10} f_j H_j \cos(\omega_j t - \theta_j) \quad (4.29)$$

where

- h_0 mean water-level
- $f_j H_j$ amplitude of harmonic constituent j
- ω_j angular velocity of j
- θ_j phase of j

The DCSM covers an area in the north-east European continental shelf, i.e. $12^\circ W$ to $13^\circ E$ and $48^\circ N$ to $62^\circ N$ as shown in Figure 4.1. The resolution of the spherical grid is $1/8^\circ \times 1/12^\circ$, which is approximately 8×8 km. With this configuration there are 201×173 grid blocks with 19809 computational grid points. The time step is $\Delta t = 10$ minutes. All open boundaries of the model are located in deep water (more than 200m), see Figure 4.1. This is done in order to model explicitly the non-linearities of the surge-tide interaction.

4.5.1 Estimation of depth

The bathymetry for a model is usually derived from nautical maps. One of the purposes of these maps is to guide large ships safely through shallow waters. Therefore these maps usually give details of shallow rather than deep-water areas. If we use these maps to prescribe the water depth, it is reasonable to assume that such prescription of the bathymetry is erroneous. So depth can be a parameter on which model can be calibrated. In the early years of the developments of the DCSM the changes to bathymetry were made manually. Later automated calibration procedures based on variational methods were developed starting from the work by [87], [63]. The complete description on the development of these calibrated procedures for DCSM can be found in [98].

The experiments are performed to assimilate data near the Dutch coast, i.e. for domain Ω (see Figure ??). Obviously, not every depth point $D_{x,y}$ can be conceived as an unknown parameter of the simulation model. Including too many parameters, identifiability will become a problem [97]. The parameter groups should be selected in accordance with the physical properties of the model. It seems logical to avoid parameter groups in which the flow varies widely [43]. The numerical domain Ω is divided into subdomains $\Omega_k, k = 1, \dots, 7$. These subdomains Ω_k are chosen based on uniformity of the depth in Ω_k . For each subdomain Ω_k a correction parameter γ_k^b is defined that is related to $D_{x,y}$ by

$$D_{x,y}^{new} = D_{x,y} + \gamma_k^b, \text{ if } (x, y) \in \Omega_k, \quad (4.30)$$

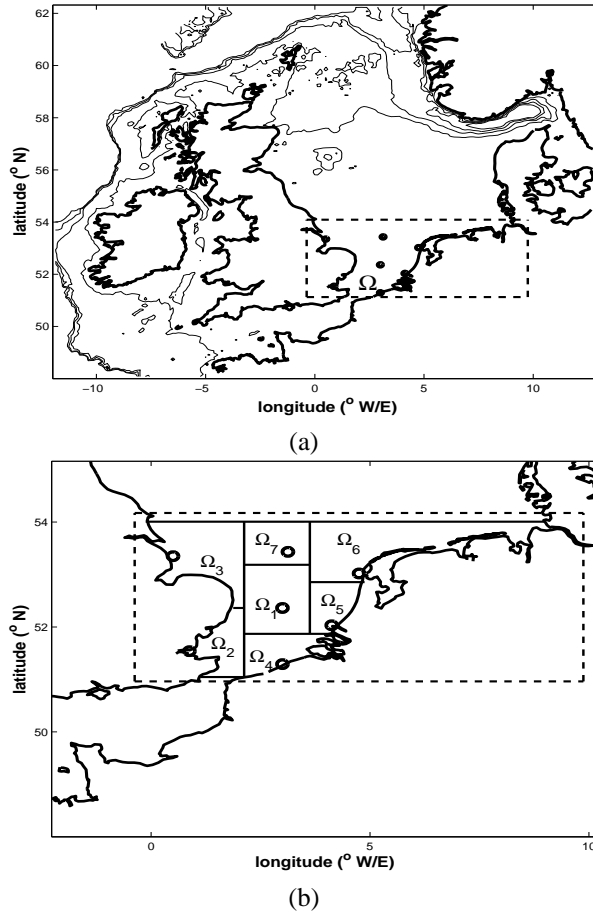


Figure 4.2: (a) The domain Ω (dashed rectangle) of DCSM; (b) The subdomains Ω_1 , Ω_2 , Ω_3 , Ω_4 , Ω_5 , Ω_6 and Ω_7

The parameters γ_k^b are treated as unknown parameters that are to be estimated. They act as a correction for the mean level of $D_{x,y}$ in a subdomain Ω_k and leave the spatial dependence inside Ω_k unaltered.

Experiment 1

Seven observation points are included in the assimilation, two of which are located along the east coast of the UK, two along the the Dutch coast and one at the Belgian coast (see Figure ??). The truth model is run for a period of six days from 13 December 1997 00:00 to 18 December 1997 24:00 with the specification of water depth $D_{x,y}$ as used in the operational DCSM to generate artificial data at the assimilation stations. The first two days are used to properly initialize the simulations. The set of observa-

tions Y of computed water levels h is collected for last four days with an interval of every ten minutes at seven selected assimilation grid points, which coincide with the points where data are observed in reality. We have assumed that the observations are perfect. This assumption is made in order to see how close is our estimate to the truth.

$5(m)$ is added to $D_{x,y}$ at all the grid points in domain Ω to get the initial adjustments γ_k^b . With this specification of the background initial parameters γ_k^b , an ensemble E of 210 snapshot vectors is collected for the period where data is available, i.e. from 15 December 1997 00:00 to 18 December 1997 24:00. The snapshots are chosen with an equal interval of 20 timesteps, so 30 snapshots are collected for each γ_k .

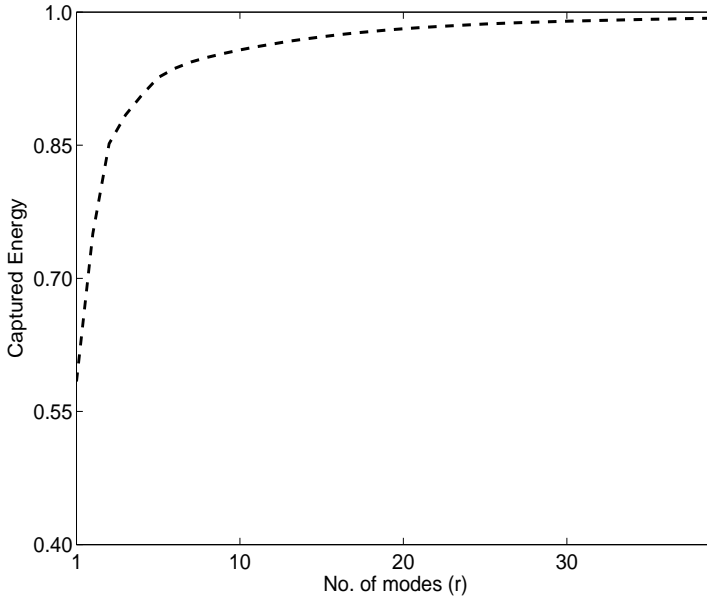


Figure 4.3: The energy captured by POD modes for an ensemble of 210 snapshots of the water level h , velocities u and v

Each snapshot vector consists of predicted water levels h , velocities u and v . Therefore it is necessary to scale the snapshot vectors before solving the eigenvalue problem (4.9). The state vector should be scaled such that all state variables become equally observable. For the shallow water equations the scaling based on energy produced at the output can be used to find a practical scaling method [95]. The potential and kinetic energies for one grid cell are

$$E_h = 1/2gh^2\rho_w\Delta x\Delta y, \quad (4.31)$$

$$E_{u,v} = 1/2(u^2 + v^2)D\rho_w\Delta x\Delta y, \quad (4.32)$$

Assume one measures surface elevations. Through propagation of the model kinetic energy may become potential energy. Since the model is dissipative the sum of the two

can only decrease or at most remain the same. This suggests that scaling state variables according to the energy they represent creates approximately equal observability if the dissipation is small. In this case, the water levels h should be scaled with \sqrt{g} and the velocities u and v with \sqrt{D} [95].

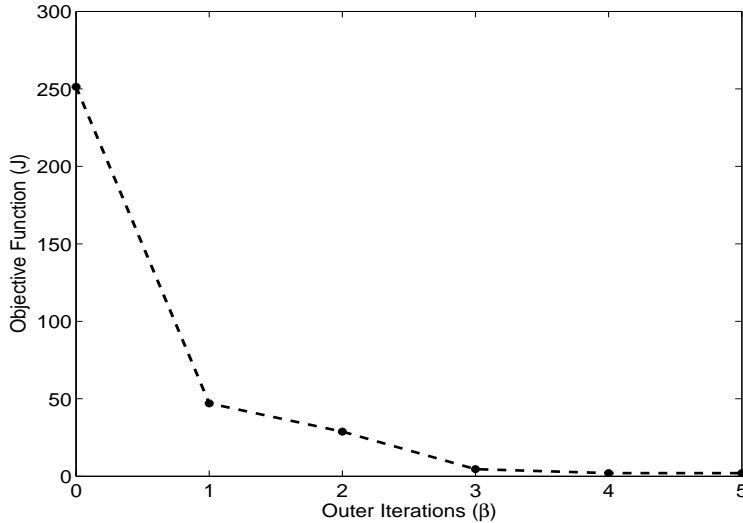


Figure 4.4: Value of the objective function J versus the number of outer iterations β

After applying the above scaling to each snapshot vector in an ensemble E we are able to form a basis consisting of only 15 dominant eigen modes that captured 97% relative energy. Figure 4.3 shows energy captured by POD modes for 210 snapshot vectors. So a reduced model is built using these 15 modes and finally operates on \mathfrak{R}^{15+7} . The low-dimensional model is defined by assuming that the matrix \bar{M} remains stationary throughout the experiment.

The numerical solution of the optimal control problem is obtained by quasi-Newton method with LBFGS (limited memory Broyden Fletcher Goldfard Shanno) updating formula. With this reduced model approximate objective function \hat{J} is minimized and new values of estimate variables are found. The objective function J is reduced by more than 80% after the inner minimization. We have stopped the inner minimization process and switched to a new outer iteration with the new set of parameters following the criterion μ , that the gradient should decrease by at least three orders of magnitude from the initial gradient value. A new POD model is required in the outer iteration if the old POD model cannot substantially reduce the objective function J . Figure 4.4 shows the history of the minimization of the objective function J in the POD based estimation approach with respect to number of outer iterations β . It is clear from Figure 4.4 that after four outer iterations the objective function J is approximately equal to its optimal value.

In the beginning of minimization process there is a significant change in param-

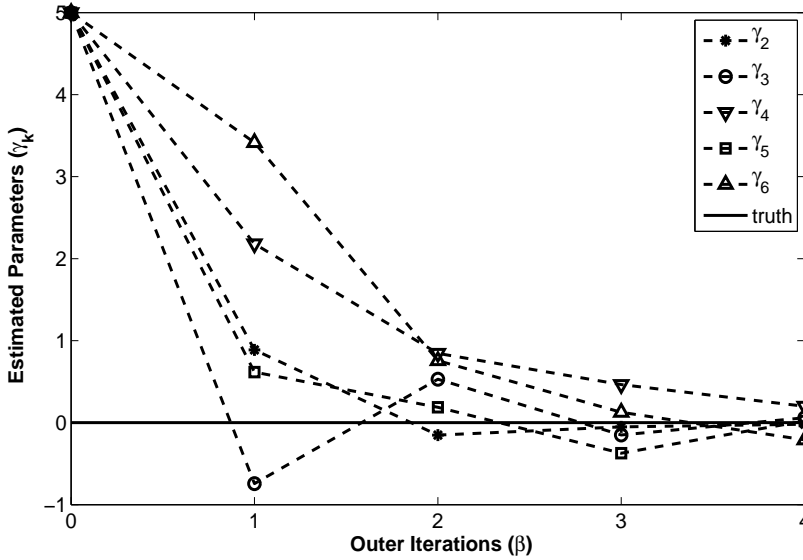


Figure 4.5: Convergence of the depth parameters that coincide along the UK, Dutch and Belgium coasts

eters for regions coinciding with the UK, Dutch and Belgian coast, but there is not much improvement in deep water regions Ω_1 and Ω_7 . The subdomains containing deep areas are less sensitive as compared to the subdomains containing shallow areas, so it is difficult to estimate γ_k^b in these areas. However in the third and fourth outer iterations, we have found improvement in the deep water regions Ω_1 and Ω_7 . Figure 4.5 shows the convergence of parameters that coincide with coastal areas.

Figure 4.6 presents water level timeseries at Den Helder and Hoek van Holland respectively for the period of 16 December 1997 00:00 - 18 December 1997 24:00. These timeseries refer to water levels obtained from simulated data (truth), timeseries using deterministic model without data assimilation, and timeseries with data assimilation after four outer iterations respectively. These figures demonstrate that the differences between the timeseries with data assimilation and the truth are always smaller as compared to the difference between the timeseries without data assimilation and the truth. It is also clear from the figures that both phase shift and correct amplitude are compensated.

To quantify the performance of the POD based estimation method at assimilation stations we use root-mean-square-error (RMSE) metric of the water level errors over the whole simulation period

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (h^t(t_i) - \hat{h}(t_i))^2} \quad (4.33)$$

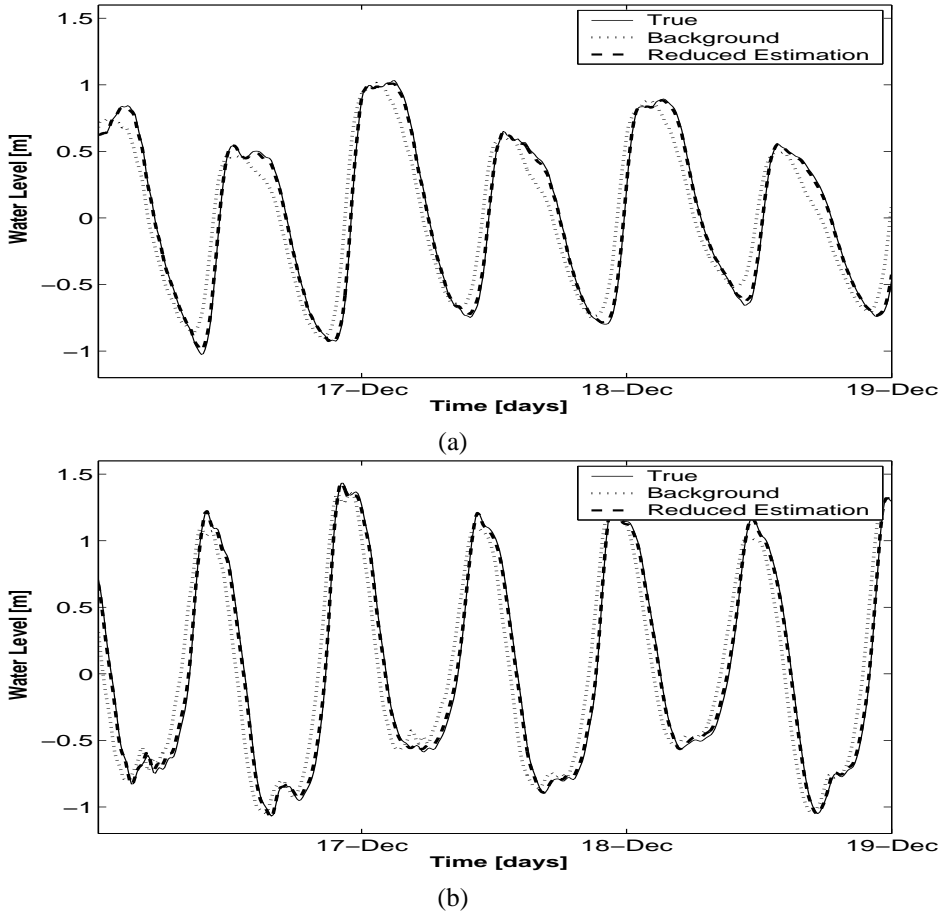
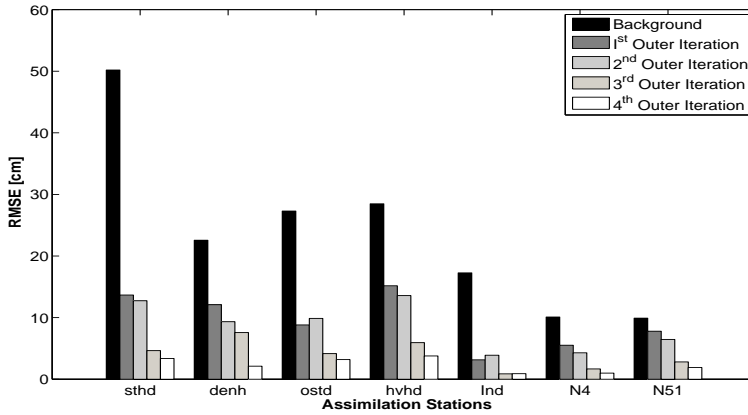


Figure 4.6: Water level timeseries for the period from 16 December 1997 00:00 - 18 December 1997 24:00 for simulated data (truth), deterministic model without assimilation (background), deterministic model with data assimilation after four outer iterations (reduced estimation) at a) Den Helder; b) Hoek van Holland

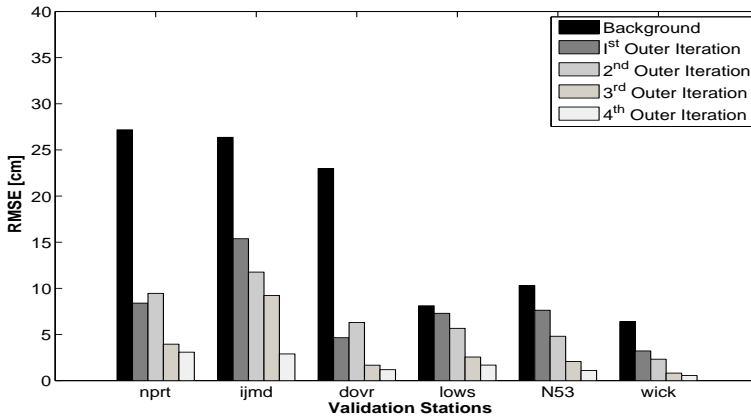
where h^t is the true water level and \hat{h} is the analysed water level. To check whether the data assimilation works at different locations, the RMSE of water level innovations is also computed for some validation stations where no assimilation was performed. Six validation stations are selected for this purpose. Figure 4.7 shows RMSE of water levels for assimilation and validation stations respectively. The POD based estimation procedure reduces the RMSE at both assimilation and validation stations.

In order to get an idea of the required computational cost we express it in terms of number of simulations with the original model. In this experiment seven parameters are estimated, so eight model simulations for a period of 6 days are performed to generate an ensemble. The snapshots are collected for last four days only, since

observations are available for this period. As the number of snapshots chosen is only 210, the computational time to solve eigenvalue problem and to construct the reduced model is much less than that of running the full model. Combined with seven parameters to be estimated, the reduced model operated eventually in R^{22} instead of $\sim R^{6 \times 10^4}$. Simulating the POD reduced model and its adjoint model over the whole simulation period requires approximately 1/50 of the time for simulating the full model. The number of outer iterations β in this experiment is four. Therefore the time required to estimate seven parameters with POD based reduced estimation procedure is equal to approximately 40 simulations with the original model. The same amount of time is required in case of the classical adjoint method see [85]. Thus the POD calibration



(a)



(b)

Figure 4.7: RMSE for water level observations at a) assimilation stations and b) validation stations

method offers an efficiency comparable to the classical adjoint method without the burden of implementation of the adjoint.

Experiment 2

In this experiment we have computed POD reduced model from a short simulation and used it for calibration over a longer period. For boundary forcing and water depth $D_{x,y}$ specified as in the previous experiment, a truth model is run for 15 days, i.e. from 13 December 1997 00:00 to 27 December 1997 24:00, to generate artificial data (observations) for the last 13 days. These observations Y are again the computed water levels for the same assimilation stations.

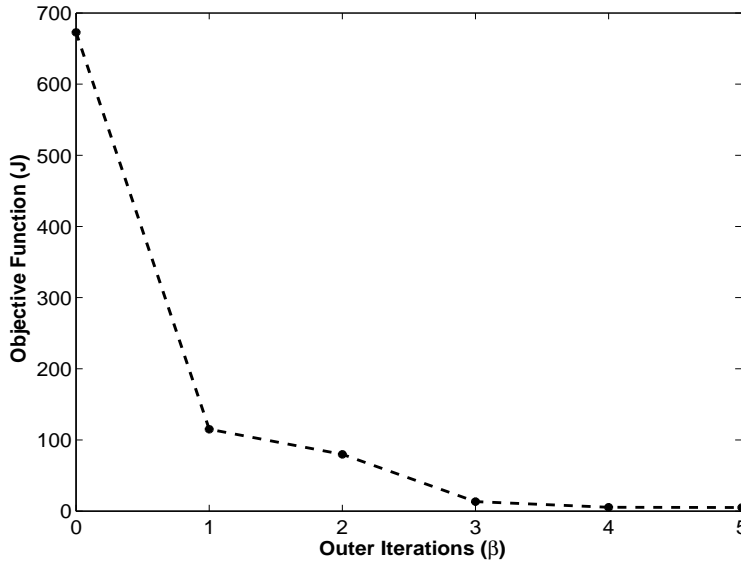


Figure 4.8: Value of the objective function J versus the number of outer iterations β for the calibration over 15 days

The specification of the background initial parameters γ_k^b is the same as in the previous experiment. We have used the same POD reduced order model which was constructed from an ensemble E of 210 snapshot vectors of forward model simulations for the period from 15 December 1997 00:00 to 18 December 1997 24:00 (4 days) as in the previous experiment. The calibration period is now 15 days, i.e. 13 December 1997 00:00 to 27 December 1997 24:00.

The approximate objective function \hat{J} reduces its value with similar magnitude. Figure 4.8 shows the minimization of the objective function J in the experiment with respect to the number of outer iterations β . The existence of the same trend in this graph as compared to the minimization of objective function J in calibration over 6 days indicates that the POD reduced model constructed over the small time period can be used for calibration over a much larger period.

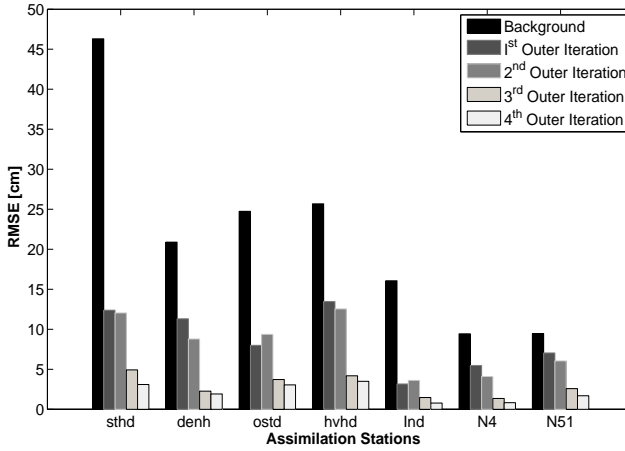


Figure 4.9: RMSE for water level observations at assimilation stations for calibration over 15 days

Figure 4.9 shows RMSE for water level at the assimilation stations. The estimation procedure reduces the RMS values of the water level errors at all assimilation stations. Figure 4.10 gives an idea about the model performance after calibration. The figure presents the water-level time series at Hoek van Holland for the period of 24 December 1997 00:00 - 24 December 1997 24:00 for the initial setting of the calibration parameters, the true calibration parameters and the calibration parameters after four outer iterations with the reduced estimation procedure.

In this experiment seven parameters are estimated during a calibration over 15 days. Seven model simulations for a period of 4 days and one model simulation with background initial parameters for a period of 15 days are performed to generate an ensemble. The total number of outer iterations β in this experiment is four, therefore the time required to estimate seven parameters is now approximately 20 simulations with the original model. Thus for the current experiment POD reduced model required 1/2 of the computational time of the classical adjoint method. The use of POD based estimation procedure has significantly reduced the computational cost.

4.6 Conclusions

The adjoint method is a powerful tool for sensitivity analysis and model calibration, but it is laborious to implement adjoint model for the computation of the gradient for large scale systems. The POD based model reduction approach presented here is used to simplify this problem using a projection based POD model reduction method. An optimal order-reduction approach to model calibration must capture accurately the properties of full dynamical model. The presented approach is designed to approx-

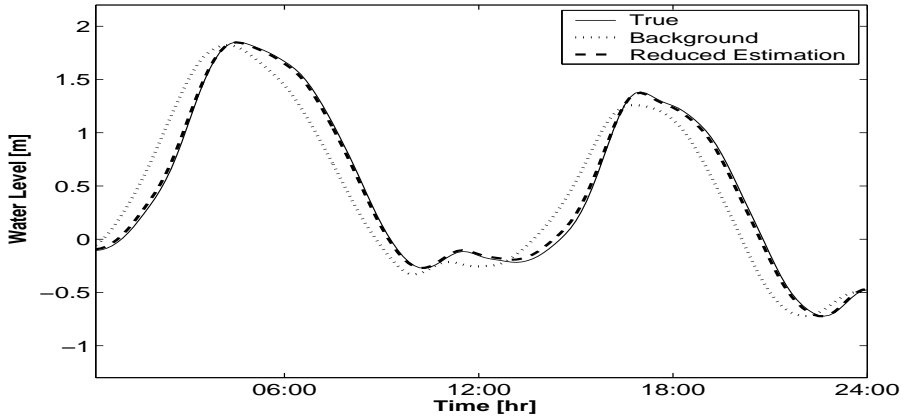


Figure 4.10: Water level timeseries at Hoek van Holland for the period from 24 December 1997 00:00 - 24 December 1997 24:00 for simulated data (truth), the initial setting of the calibration parameters (background) and deterministic model after calibration (reduced estimation)

imate the data assimilation system in a restricted space while retaining its essential properties. The method fits into the theory of incremental variational data assimilation by using restriction and prolongation operators.

The method has been used to calibrate the operational model for storm surge prediction, the DCSM. Twin experiments have been performed to estimate the water depth. The results show that the calibration method performs very efficiently. A POD reduced model of much smaller size R^{15} has been constructed instead of original model of size ($\sim R^{6 \times 10^4}$). After four outer iterations β the objective function J has reduced significantly and is very close to optimal value. The RMSE at both assimilation and validation stations have improved significantly.

Considering the model periodicity in time an experiment has been done to calibrate the model for longer period with a reduced order model over shorter period. The same trend in the minimization of objective function J has been observed for the calibration over 15 days from the same POD model that is used for the calibration over six days. This result demonstrates the potential usage of the method to calibrate DCSM model for much longer period. Moreover, the POD calibration offers a relatively efficient method compared to the classical adjoint method without the burden of implementation of the adjoint.

The classical method employs the adjoint of tangent linear model which is replaced here by the adjoint of linear reduced model. Compared to the classical adjoint method the minimization in reduced space converges faster due to better condition number of the reduced Hessian. The method has a limitation that it has to be updated at each outer iteration by constructing a new POD model by generating an ensemble of forward model simulations. Moreover, the quality of ensemble and the process of generating ensemble is crucial for a reduced order procedure to be effective.

Case Study: Identification of uncertain parameters in a large scale tidal model¹

5.1 Introduction

Accurate sea water level forecasting is crucial in the Netherlands. This is mainly because large areas of the land lie below sea level. Forecasts are made to support storm surge flood warning system. Timely water level forecasts are necessary to support the decision on closure of the movable storm surge barriers in the Eastern Scheldt and the New Waterway. Moreover, forecasting is also important for harbor management, as the size of some ships has become so large that they could only enter the harbor during high water period. Storm surge warning service (SVSD) in close cooperation with royal Netherlands meteorological institute (KNMI) is responsible for these forecasts. The surge is predicted by using a numerical hydrodynamic model, the Dutch continental shelf model (DCSM) (see [82], [94]). Performance of the DCSM regarding the storm surges is influenced by its performance in forecasting the astronomical tides. Using inverse modeling techniques, these tidal data can be used to improve the model results.

The adjoint method is a well-known approach to inverse modeling. The method aims at adjusting a number of unknown control parameters on the basis of given data. The control parameters might be model initial conditions or model parameters [88], [89]. An objective function is defined which measures the misfit between the solution and the available data for any model solution over the assimilation interval. This objective function is typically a sum of squared differences between the data and the corresponding model values. One then looks for the model solution that minimizes this objective function. To obtain a computationally efficient procedure this objective

¹This chapter is a slightly revised version of [9]

function is minimized with a gradient-based algorithm where the gradient is determined by solving the adjoint problem. The adjoint approach is computationally very efficient because one gradient calculation requires just a single simulation of the forward model and a single simulation of the adjoint model backward in time, irrespective of the number of parameters. The adjoint method has been used and applied successfully to many types of inverse problems in ground water flow studies (e.g. [16]), in meteorology (e.g. [20]), in oceanography (e.g. [91]) and in shallow water flow models (e.g. [87], [53], [92], [43]).

One of the drawbacks of the adjoint method is the programming effort required for the implementation of the adjoint model. Research has recently been carried out on automatic generation of computer codes for the adjoint, and adjoint compilers have now become available (see [47]). Even with the use of these adjoint compilers developing an adjoint model is often a significant programming effort that hampers new applications of the method. Courtier [21] had proposed an incremental approach, in which the forward solution of the nonlinear model is replaced by a low resolution approximate model. Reduced order modeling can be used in an incremental approach to obtain an efficient low order approximate linear model [54].

Proper orthogonal decomposition (POD) is a model reduction method considered as an application of the singular value decomposition (SVD) to the approximation of general dynamical systems [11]. It is a data driven projection based method originally developed by Karl Pearson [68]. Karhunen [48] and Loeve [56] had used it as statistical tool to analyze random process data. Lumley [59] gave the name POD, and used the method to study turbulent flow. The POD method has application in many fields like image processing, signal processing, data compression, oceanography, chemical engineering and fluid mechanics (see [40]). In the POD method the projection subspace is determined by processing data obtained from numerical simulations of high dimensional model which is expected to provide information about the dynamical behavior of the system. The high dimensional equations are projected onto the low dimensional subspace resulting in a low dimensional model. In this way, the POD method reduces the CPU time of model simulation but does not change the complexity of the problem and consequently does not solve the implementation problem of the adjoint model. The method has recently been investigated by (e.g. [15], [22], [32]). The POD method has also been applied successfully for parameter estimation (e.g. [26]).

Vermeulen and Heemink [99] proposed a method based on POD which shifts the minimization into lower dimensional space and avoids the implementation of the adjoint of the tangent linear approximation of the original nonlinear model. In their approach, an ensemble of snapshot vectors of forward model simulations is used to determine an approximation of the covariance matrix of the model variability and a small number of leading eigenvectors of this matrix is used to define a model subspace. By projecting the original model onto this subspace an approximate linear reduced model is obtained. Due to the linear character of the reduced model its adjoint can be implemented easily and the minimization problem is solved completely in reduced space with very low computational cost. The method has recently been successfully applied to the 2D-DCSM to estimate water depth [5]. Several synthetic cases were used to show that the depth parameters were correctly identified in the selected regions of the

model domain. The generation of an ensemble in the POD method involves running the forward model several times. The computational cost of the method is dominated by the generation of this ensemble. It was also found in the study that if the dynamics of the system does not change significantly then a smaller simulation period can be chosen to generate an ensemble of forward model simulations for an optimization problem over larger period [5, 8].

The first version of DCSM was developed in the 1980's and has been through numerous improvements since then. Among the developments was the increased resolution of the wind forecast input from 55 to 22 km, which could slightly increase the forecast quality in storm situations. Another attempt was to refine further the resolution of the wind input to 11 km. However case studies comparing resolution 22 km and 11 km input for DCSM showed that the 11 km resolution could not improve the prediction. Besides works on improving the wind input, a lot of studies had also been devoted to model calibration ([87], [63], [43]). The last calibration was performed in 1998 using the adjoint method [61]. It was assumed that with the limited model resolution of approximately 8 km and certain quality of the bathymetry information further calibration was not worthwhile. For further details on the development of the DCSM see [98].

DCSM version 6 is the recently designed large-scale spherical grid based water level model for the northwest European continental shelf (around 10^6 computational grid points). It covers a much larger deep water area than the operational version of DCSM (see Figure 5.1) and has a spatial resolution that is a factor 5 finer in both latitudinal and longitudinal directions. The flow exchanges with the non-modeled part of the Atlantic Ocean are modeled by prescribing the tidal water level variation along the open boundary of the domain. One of the objectives of this new development is to extend the time horizon of the water level forecasts in the near future and to make these forecasts for a denser distribution of locations along the Dutch coast. At this moment the forecasts are made for just the five 5 main stations which are taken as representative for a specific coastal section.

In this chapter, the POD based model reduced approach is used for the estimation of the water depth and space varying bottom friction coefficient values in large-scale DCSM model. This is the first application of the method to a very large-scale model and real data. The computational costs of the method are dominated by the generation of an ensemble of forward model simulations. The simulation period of the ensemble is equivalent to the timescale of the original model. Here an accurate reduced model is obtained from an ensemble with a relatively short simulation period as compared to the calibration period. A new ensemble is usually required with the suboptimal estimated parameters until the convergence criterion is achieved. It is also found in the present study that a new ensemble is not always required for the new set of estimated parameters. Instead of defining a new model subspace of the leading eigenvectors by generating a new ensemble of the forward model simulations, we obtained a reduced model by projecting original model with the updated parameters onto the same subspace.

The chapter is organized as follows. Section 5.2 briefly describes the DCSM model used in this study. The methodology of POD projection based reduced method for the calibration is explained in Section 5.3. Section 5.4 discusses the experiments

with the newly developed DCSM to estimate the water depth and the bottom friction coefficient. Section 5.5 presents the conclusions.

5.2 The new Dutch continental shelf model

Since the mid-1980's, the storm surge forecasts in the Netherlands are based on a numerical hydrodynamic model DCSM. This model uses forecasts of the meteorological high-resolution limited area model (HIRLAM) as input. The model is based on shallow water equations. The use of a 2D model implies that flow related quantities are computed in depth-averaged form, which is a commonly used approximation for tidal modeling. The model is based on the work of [55] and improved by [82]. In the implementation the spherical grid is used instead of rectangular (see e.g. [94]).

The time between warning for dangerous high water and the actual occurrence of the high water is an important parameter in the planning of the response, e.g. decisions on the closure of the storm surge barriers, dike watch and potentially even an activation of an evacuation scenario. The objectives of this new development include obtaining better quality of the forecasts, forecasts for a denser distribution of locations along the Dutch coast and increasing the forecasts horizons.

5.2.1 Model computational grid of the new model

The newly developed DCSM covers an area in the northwest European continental shelf, i.e. $15^{\circ}W$ to $13^{\circ}E$ and $43^{\circ}N$ to $64^{\circ}N$, as shown in Figure 5.1. The spherical grid has a uniform cell size of $1/40^{\circ}$ in eastwest direction and $1/60^{\circ}$ in northsouth direction which corresponds to a grid cell size of about 2×2 km. With this configuration there are 1120 grid cells in eastwest direction and 1260 grid cells in north-south direction. The grid cells that include land are excluded from the model and the model finally contains 869544 computational grid points. The grid resolution of the spherical grid is factor five finer than of the operational DCSM.

5.2.2 Model bathymetry and bottom roughness

The bathymetry D for a model is usually derived from nautical maps. One of the purposes of these maps is to guide large ships safely through shallow waters. The bathymetry for the new DCSM is based on a NOOS gridded data set and for some areas in the model ETOPO2 bathymetry data are interpolated on the computational grid. The model bathymetry is presented in Figure 5.2. The North Sea is much shallower, with maximum depth around 200m. In southern North Sea (English Channel) the depths are mostly less than 50m. In the southwestern and northern parts of the model domain the depth exceeds 2000m.

To account for the bottom friction, the empirical 2D-Chezy coefficient is computed in the direction of the velocities u and v according to the following formulation:

$$C_{2D} = \frac{\{H\}^{\frac{1}{6}}}{c_m}, \quad (5.1)$$

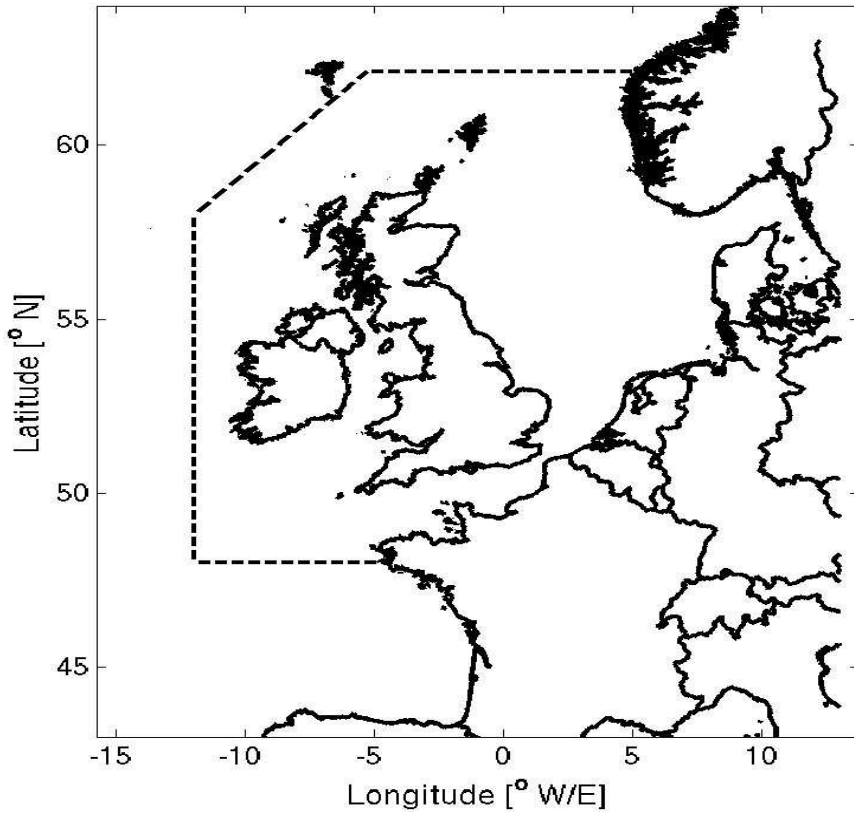


Figure 5.1: Newly developed hydrodynamic DCSM area. The dashed line represents the area of the operational DCSM extent

where

H total water depth

c_m the manning coefficient

A uniform value of $0.028s/m^{1/2}$ has been applied for the manning coefficient c_m . This initial value of the c_m has been obtained through manual calibration.

5.2.3 Model boundary conditions

Boundary conditions are applied at both closed and open boundaries. At closed boundaries the velocity normal to the boundary is zero. So no inflow and outflow can occur through these boundaries. The dashed line in Figure 5.1 shows the comparison of the new DCSM area with the operational DCSM. The model area of the new DCSM is extended significantly in order to ensure that the open boundary conditions are located further away in deep water. At the northern and western sides of the model domain

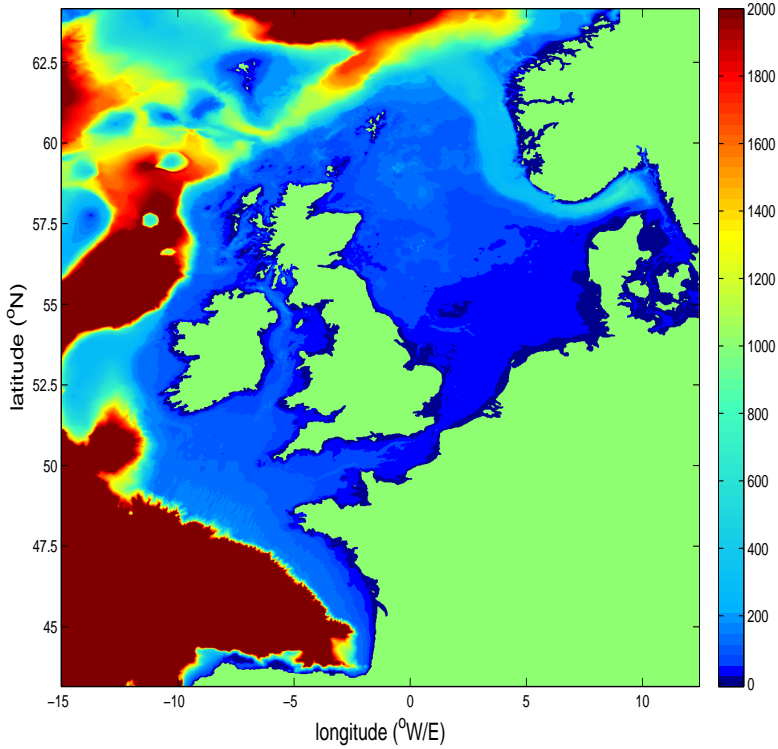


Figure 5.2: DCSM model bathymetry in meters. The bathymetry greater than 2000m is shown as 2000m

water levels are specified at 205 different locations along the open boundaries. These water levels h at the open boundaries are described in terms of the amplitude and phase of the ten harmonic components ($M_2, S_2, N_2, K_2, O_1, K_1, Q_1, P_1, U_2, L_2$) as follows:

$$h(t) = h_0 + \sum_{j=1}^{10} f_j H_j \cos(\omega_j t - \theta_j), \quad (5.2)$$

where

- h_0 mean water-level
- $f_j H_j$ amplitude of harmonic constituent j
- ω_j angular velocity of j
- θ_j phase of j

The tidal conditions of these ten main diurnal and semi-diurnal constituents have been derived by interpolation from a dataset that is obtained from the GOT00.2 global tidal model.

5.2.4 Initial conditions and computational time

A uniform initial water level of 0m mean sea level (MSL) is used. First four days of the model run are considered as the spin-up period. Zero flow conditions are prescribed for the initial velocity. The time zone of the model is GMT. A computational time step of 2 minutes is applied. So it takes more than 2 days to complete a one year model run on eight 3.6 MHz CPUs.

5.3 Inverse modeling using POD

The discrete model for the evaluation of shallow water system from time t_i to time t_{i+1} can be described by an equation of the form

$$X(t_{i+1}) = M_i[X(t_i), \gamma], \quad (5.3)$$

where state vector $X(t_{i+1}) \in \mathfrak{X}^n$ denotes the vector of water levels h , velocities u and v at time t_{i+1} and γ is the vector of the uncertain parameters which needs to be determined. M_i is nonlinear and deterministic dynamics operator that includes inputs. Suppose now that we have imperfect observations $Y(t_i) \in \mathfrak{X}^{n^q}$ of the dynamical system (5.3) that are related to the model state at time t_i through

$$Y(t_i) = HX(t_i) + \eta(t_i), \quad (5.4)$$

where $H : \mathfrak{X}^n \rightarrow \mathfrak{X}^{n^q}$ is a linear observation operator that maps the model fields on observation space and $\eta(t_i)$ is an unbiased, random Gaussian error vector with covariance matrix R_i .

We assume that the difference between data and simulation results is only due to measurement errors and incorrectly prescribed model parameters. The problem of the estimation is then solved by directly minimizing the objective function J

$$J(\gamma) = \sum_i [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] \quad (5.5)$$

with respect to the parameters γ satisfying the discrete nonlinear forecast model (5.3).

The efficient minimization of the objective function requires the computation of the gradient of the objective function (5.5). The gradient vector ∇J gives information about the direction (positive or negative) and the size of adjustments for each individual parameter. The adjoint method computes the exact gradient efficiently. The principle of the adjoint method is based on the systematic use of the chain rule of differentiation. Regardless of the number of parameters, the time required to compute the gradient using adjoint technique is more or less identical and is comparable with the computational time needed for a single simulation run of the nonlinear model (5.3). It requires one forward simulation with the original nonlinear model (5.3) and a second additional simulation backward in time with the adjoint model

$$\nu(t_i) = \left(\frac{\partial M_i}{\partial \mathbf{X}(t_i)} \right)^T \nu(t_{i+1}) - 2HR_i^{-1} [\mathbf{y}(t_i) - H(\mathbf{X}(t_i))], \quad (5.6)$$

where $v(t_i)$ represents the solution of the adjoint model. The gradient ∇J of the objective function J with respect to each component γ_k of the uncertain parameters vector γ is given by

$$\nabla J_k = \sum_i - [v(t_{i+1})]^T \left[\frac{\partial M_i[\mathbf{X}(t_i), \gamma]}{\partial \gamma_k} \right], k = \{1, \dots, n^p\}. \quad (5.7)$$

The adjoint method is flexible as the number of parameters can easily be changed. The main hurdle in the use of adjoint method is its implementation, especially when the forward model contains nonlinearities. For the shallow water flow computations the original model is very complicated and it is difficult to implement the adjoint for this type of models.

5.3.1 Linearization and reduced model formulation

The classical adjoint problem for a general model is a nonlinear constrained optimization problem which is difficult to solve. The problem can be simplified with the hypothesis that the objective function J can be made quadratic by assuming that the nonlinear dynamics operator M_i can be linearized. The linearization of nonlinear high-order model (5.3) using the first order Taylor's formula around the background parameter γ_k^b gives

$$\Delta \bar{X}(t_{i+1}) = \frac{\partial M_i[X^b(t_i), \gamma^b]}{\partial X^b(t_i)} \Delta \bar{X}(t_i) + \sum_k \frac{\partial M_i[X^b(t_i), \gamma^b]}{\partial \gamma_k} \Delta \gamma_k \quad (5.8)$$

where \bar{X} is linearized state vector, X^b is the background state vector with the prior estimated parameters vector γ^b and $\Delta \bar{X}$ is a deviation of the model from background trajectory.

A model can be reduced if the incremental state $\Delta \bar{X}(t_{i+1})$ can be written as linear combination

$$\Delta \bar{X}(t_i) = P \xi(t_{i+1}), \quad (5.9)$$

where $P = [p_1, p_2, \dots, p_r]$ is a projection matrix such that $P^T P = I_r$ and ξ is a reduced state vector given by

$$\begin{pmatrix} \xi(t_{i+1}) \\ \Delta \gamma \end{pmatrix} = \begin{pmatrix} \tilde{M}_i & \tilde{M}_i^\gamma \\ 0 & I \end{pmatrix} \begin{pmatrix} \xi(t_i) \\ \Delta \gamma \end{pmatrix} \quad (5.10)$$

Here $\Delta \gamma$ is the control parameter vector, \tilde{M}_i and \tilde{M}_i^γ are simplified dynamics operators which approximate the full Jacobians $\frac{\partial M_i}{\partial \mathbf{X}^b}$ and $\frac{\partial M_i}{\partial \gamma_k}$ respectively:

$$\tilde{M}_i = P^T \frac{\partial M_i}{\partial \mathbf{X}^b(t_i)} P, \quad (5.11)$$

$$\tilde{M}_i^\gamma = P^T \left(\frac{\partial M_i}{\partial \gamma_1}, \dots, \frac{\partial M_i}{\partial \gamma_{n^p}} \right), \quad (5.12)$$

The Jacobian $\frac{\partial M_i}{\partial \mathbf{X}^b}$, is obtained by approximating the nonlinear dynamics operator M_i by linearizing it with respect to background state X^b . Instead of computing this huge

Jacobian by approximating the partial differential with finite difference by perturbing the nonlinear operator M_i in the direction of each node, we perturb along the direction of p_h only:

$$\frac{\partial M_i}{\partial X^b(t_i)} p_h = \frac{M_i[X^b(t_i) + \varepsilon p_h, \gamma^b] - M_i[X^b(t_i), \gamma^b]}{\varepsilon}, h = \{1, \dots, r\}, \quad (5.13)$$

with ε being the size of the perturbation. The reduced dynamics operator \tilde{M}_i can now be computed by premultiplying the above formulae by P^T :

$$\tilde{M}_i = P^T \left(\frac{\partial M_i}{\partial X^b(t_i)} p_1, \dots, \frac{\partial M_i}{\partial X^b(t_i)} p_r \right). \quad (5.14)$$

Notice also that only the original model simulations are needed here. The reduced model requires less computational time as it simulates a reduced state within the dimension r instead of the original dimension n where $r < n$. The dimension on which the reduced model operates is $(r + n^p) \times (r + n^p)$ with n^p being the number of estimated parameters.

5.3.2 Collection of the snapshots and POD basis

The POD method is used here to obtain an approximate low-order formulation of the original tangent linear model. POD is an optimal technique of finding a basis which spans an ensemble of data (snapshots) collected from an experiment or a numerical simulation of a dynamical system. The POD modes are optimal at approximating a given dataset. Since the reduced model is used here to estimate uncertain parameters (depth D and manning coefficient c_m), the snapshots should be able to represent the behavior of the system for these parameters. Therefore the snapshot vectors $e_i \in \mathfrak{R}^s$ are obtained from the perturbations $\frac{\partial M_i}{\partial \gamma_k}$ along each estimated parameter γ_k to get a matrix

$$E = \{e_1, \dots, e_s\}; i = \{1, 2, \dots, s\}. \quad (5.15)$$

The dimension of this ensemble matrix E is $s = n^p \times n^s$, where n^s is the number of snapshot collected for each individual parameter γ_k . The covariance matrix Q can be constructed from the ensemble E of the snapshots by taking the outer product

$$Q = EE^T \quad (5.16)$$

The projection matrix P used in the previous section is based on the dominant eigenvectors (POD modes) of this covariance matrix which are obtained as explained in Section ??.

5.3.3 Approximate objective function and its adjoint

In reduced model approach, we look for an optimal solution of the (5.3) to minimize the approximate objective function \hat{J} in an incremental way:

$$\hat{J}(\Delta\gamma) = \sum_i [\{Y(t_i) - H(X^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)]^T R_i^{-1} [\{Y(t_i) - H(X^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)]. \quad (5.17)$$

The value of the approximate objective function \hat{J} is obtained by correcting the observations $Y(t_i)$ for background state $X^b(t_i)$ which is mapped on the observational space through a mapping H and for the reduced model state $\xi(t_i, \Delta\gamma)$ which is mapped to the observational space through mapping \hat{H} with $\hat{H} = HP$.

Since the reduced model has linear characteristics, it is easy to build an approximate adjoint model for the computation of gradient of the approximate objective function (5.17). The gradient of \hat{J} with respect to $\Delta\gamma$ is given by

$$\frac{\partial \hat{J}}{\partial (\Delta\gamma)} = \sum_i -[\hat{v}(t_{i+1})]^T \frac{\partial \xi(t_{i+1})}{\partial (\Delta\gamma)}, \quad (5.18)$$

where $\hat{v}(t_{i+1})$ is the reduced adjoint state variable (see Appendix A). Once the gradient has been computed, the process of minimizing the approximate objective function \hat{J} is done along the direction of the gradient vector in the reduced space.

After the minimization process the initial parameters γ are updated and new set of updated parameters γ^{up} is obtained:

$$\gamma^{up} = \gamma + \Delta\gamma. \quad (5.19)$$

This process of minimization is repeated several times by constructing new POD model with new set of updated parameters γ^{up} to get optimal parameters.

5.3.4 Workflow with POD algorithm

In order to perform the whole parameter estimation process, the following steps are executed.

1. Outer Iteration:
 - (a) Generate an ensemble of forward model simulations using initial parameters γ^b .
 - (b) Solve eigenvalue problem to get dominant eigenmodes p_i
 - (c) Establish a POD reduced model and its adjoint model using eigenmodes p_i .
2. Inner Iteration:
 - (a) Perform optimization iterations in the reduced space to obtain the optimal solution of the approximate objective function \hat{J} .
 - (b) Update the initial parameters γ^b after the minimization process obtain new set of updated parameters γ^{up} .
3. Return to step 1 with new set of updated parameters γ^{up} until optimality condition is achieved.

5.3.5 Convergence criterion for inner and outer iterations

The minimization is performed using a quasi-Newton optimization algorithm where the Hessian of the objective function is updated using the limited Broyden-Fletcher-Goldfarb-Shanno (LBFGS) method. The minimization algorithm requires convergence criteria to terminate. We have defined two criterions, one is for inner iterations and one is for outer iterations of the optimization process. We stop the present inner iteration α and switch to a new outer iteration β with updated parameters γ^{up} by criterion μ , which is defined as

$$\mu = \frac{|\hat{J}_{\alpha_{i+1}} - \hat{J}_{\alpha_i}|}{\max\{|\hat{J}_{\alpha_{i+1}}|, 1\}} < \epsilon, \quad (5.20)$$

where α_i represents the i^{th} inner iteration. The value of the ϵ is chosen such that the approximate objective function \hat{J} stops to change, i.e. $\epsilon = 10^{-4}$ (see [66]). The outer iteration cycle is aborted when the terminal value of ρ is obtained

$$\rho = \frac{|J_{\beta_i} - J_{\beta_{i-1}}|}{|J_{\beta_i}|} \leq \kappa, \quad (5.21)$$

where β_i stands for the i^{th} outer iteration, κ is the terminal value.

5.3.6 Computational efficiency of the algorithm

The computational efficiency of the model-reduced approach is influenced by three factors.

1. Ensemble generation: The computational costs of the reduced model approach are dominated by the generation of the ensemble of forward model simulations. If the dynamics of the system does not change significantly during the course of simulation then a smaller simulation period can be chosen for the generation of ensemble [5]. Using this ensemble the optimization problem can then be solved over the whole period of model simulation.

To achieve convergence, the POD method needs to be updated in each outer iteration β , so the ensemble E of snapshot vectors is required in each β . Instead of defining a new model subspace of the leading eigenvectors in each β by generating a new ensemble of the forward model simulations, it is possible to obtain the reduced model by projecting the original model with updated parameters onto the same subspace.

2. Ensemble size: The efficiency of optimization process is also influenced by the ensemble size. A large ensemble size leads to a huge eigenvalue problem. On the other hand, since the ensemble gives the representation of the model behavior with respect to each γ_k , it is important that the number of snapshot vectors included in the ensemble must give this representation. So the quality of ensemble is crucial for a reduced-order procedure to be effective. It is possible to include only those snapshots in the ensemble for the period where data is available.

3. Outer iteration: The convergence criterion ρ should be carefully chosen. It should not be chosen too small as this causes jumping of the updated parameters γ^{up} around the optimal global solution [99]. For the current application we have chosen $\kappa = 0.05$.

5.4 Parameter identification for the new DCSM

In this section, the POD based calibration approach described before is used for the calibration of the newly developed DCSM. The following parameters are estimated:

1. correction parameter for the depth,
2. correction parameter for the manning roughness coefficient.

In the early years of the developments of the DCSM these parameters were adapted manually. Later automated calibration procedures based on the variational methods were developed starting from the work of [87], [63]. An overview of the development of these calibration procedures for DCSM can be found in [98].

Both depth and bottom friction have to be prescribed at each grid cell of the model. Thus, theoretically it is possible to consider depth and bottom friction at each grid cell as a parameter to adapt. Practically it is not possible to take the adaptation values of every grid point as a parameter since far too many parameters would then have to be estimated in proportion to the available amount of data. Including too many parameters, identifiability will become a problem [97]. Here the rectangular areas are chosen, for which adaptation parameters are considered. These rectangular areas are chosen based on the previous calibrations of the DCSM [85] and the spatial correlations within the rectangular regions.

5.4.1 Measurement data

The model performance can be assessed by comparing it to the measured (observed) dataset. The available data used in this research consists of two datasets of the tide gauge stations are used, namely,

1. water level measurement data from the Dutch DONAR database,
2. BODC (British Oceanographic Data Center) offshore water level measurement data.

In the operational system the astronomical tide component of the observed data is replaced by the one predicted using the DCSM, obtained by running the DCSM without any wind input forcing [37]. The target of the calibration of the parameters (i.e. depth and bottom friction coefficient) is to optimize the model for its reproduction of the astronomical tide. The tide gauge data are therefore retrieved from the results of the harmonical analysis to exclude the meteorological influences.

stations that are used for the calibration are not included in the dataset that is used for the validation. The model validation with different dataset and time period prevents the model to be adjusted too much to one specific dataset and time period. Another validation dataset, a so-called Dutch dataset, is prepared (see Figure 5.5). It includes the stations from both calibration and validation datasets along the Dutch coast. This dataset is used as an extra control to closely monitor the improvement along the Dutch coast for the validation time period.

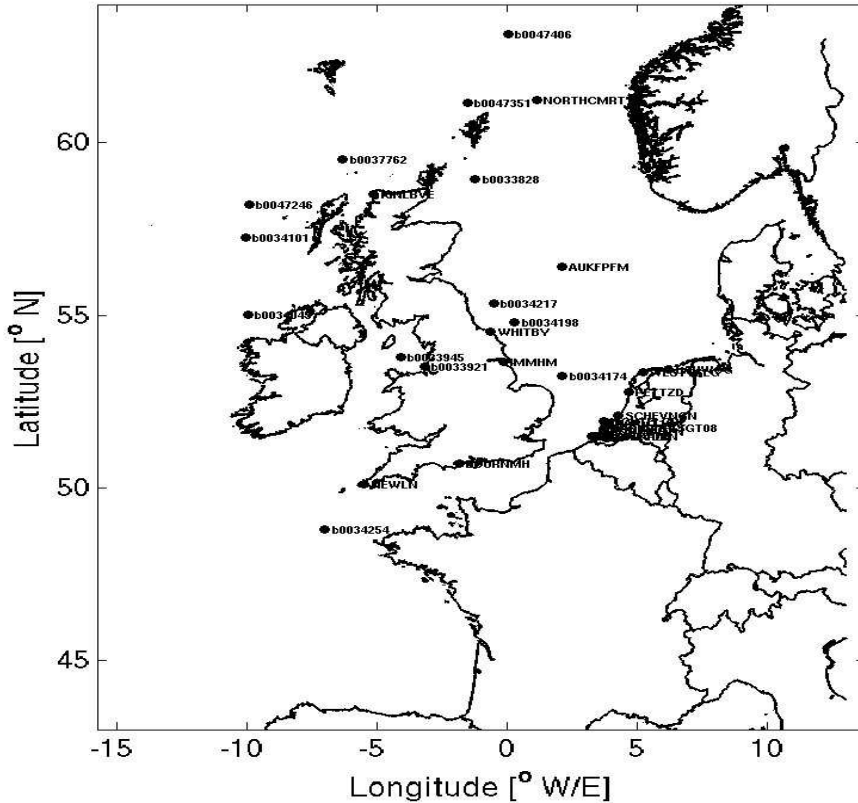


Figure 5.4: DCSM area with stations included in the validation dataset

5.4.3 Time and frequency domain analysis

The tidal predictions have been made using amplitudes and phases derived from the harmonic analysis of the DONAR and BODC measurements. In this way the water level variation resulting from the tides is the only signal. To measure the overall performance of the model in time domain, the root-mean-square-error (RMSE) is determined for each station over the calibration and validation periods. The RMSE metric

is also used to measure the quality of model simulation for different datasets used for calibration and validation periods.

Although the calibration is performed in time domain, it is usually more convenient to interpret the results in the frequency domain. For the frequency analysis a vector difference (VD) is defined [70]:

$$VD = [\{A_c \cos(G_c) - A_o \cos(G_o)\}^2 + \{A_c \sin(G_c) - A_o \sin(G_o)\}^2]^{\frac{1}{2}}, \quad (5.22)$$

where A_c and G_c represent the computed amplitude (in cm) and phase (in radian) respectively, while A_o and G_o represent the observed amplitude and phase respectively for the tide gauge stations. The RMS(VD) is also obtained in which the mean is taken over the datasets of tide gauge stations used for the calibration and the validation periods.

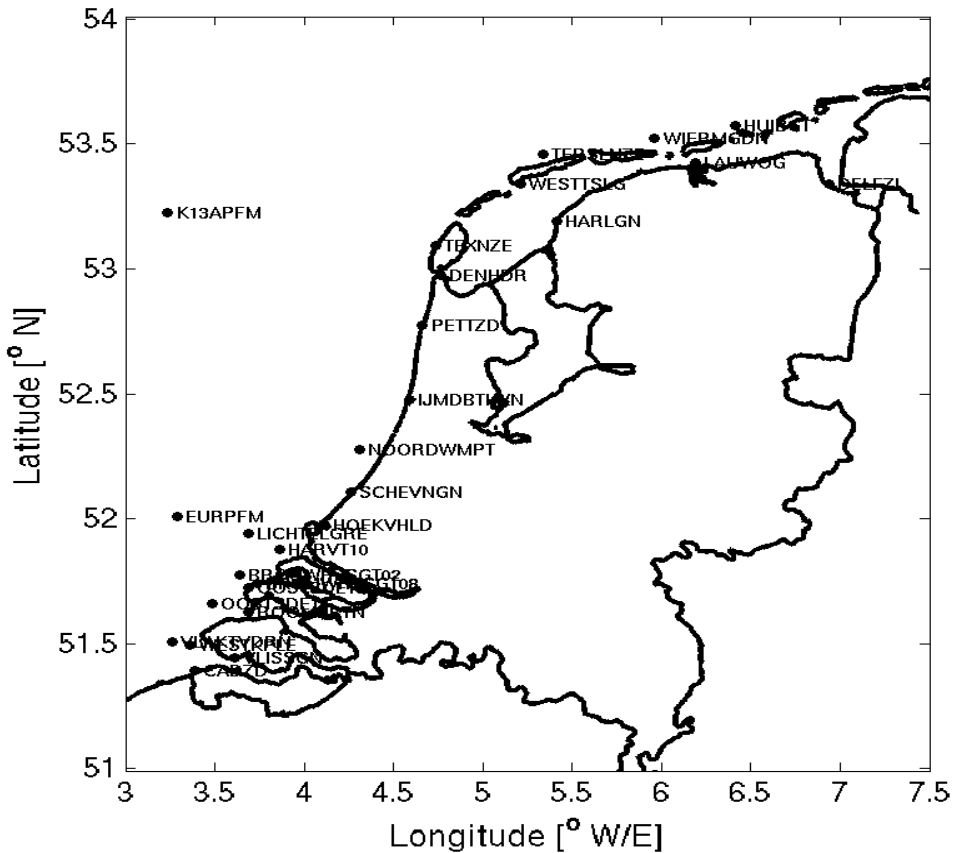


Figure 5.5: Stations along the Dutch coast included in the Dutch dataset for the validation period

5.4.4 Ensemble generation

As explained in Section 5.3.1, the generation of the ensemble involves perturbations $\frac{\partial M}{\partial \gamma_k}$ with respect to each parameter for the whole simulation period. If the dynamics of the system does not change significantly then a smaller simulation period can be chosen to generate an ensemble of forward model simulations for an optimization problem over larger period [?]. The ensemble E is generated using forward model simulations for a period from 28 December 2006 00:00 to 04 January 2007 24:00. The snapshot vectors in the ensemble are collected for the period where data is available, i.e. from 01 January 2007 00:00 to 04 January 2007 24:00. The snapshots are chosen with an equal interval of 03 hours, so 33 snapshots are collected for each γ_k .

5.4.5 Scaling

Each snapshot vector consists of predicted water level h , velocities u and v . Before solving the eigenvalue problem as explained in the section 5.3.2 to find dominant eigenmodes, it is necessary to scale the snapshot vectors. The state vector should be scaled such that all state variables become equally observable. One approach here is based on the energy. The potential energy of a surface elevation h above the reference plane for one grid cell is

$$E_h = 1/2gh^2\rho_w\Delta x\Delta y \quad (5.23)$$

and the kinetic energy of the grid cell is

$$E_{u,v} = 1/2(u^2 + v^2)D\rho_w\Delta x\Delta y, \quad (5.24)$$

where

- g the gravitational acceleration
- ρ_w the density of the water

Assume one measures surface elevations. Through propagation of the model kinetic energy may become potential energy, and because the model is dissipative, the sum of the two can only decrease or at most remain the same. This suggests that scaling the state variables according to the energy they represent creates approximately equal observability if the dissipation is small. In this case, the water levels should be scaled with \sqrt{g} and the velocities u and v with \sqrt{D} (see [95]).

5.4.6 Model calibration

The tidal calibration of the DCSM was carried out in steps. Three experiments are performed during the calibration process, two with the estimated parameter depth and one with the bottom friction.

Experiment 1

For this experiment, the numerical domain Ω is divided into four subdomains Ω_k , $k = 1, \dots, 4$ (see Figure 5.6). For each subdomain Ω_k a correction parameter γ_k^b is defined that is related to $D_{x,y}$ by

$$D_{x,y}^{new} = D_{x,y}(1 + \gamma_k^b), \text{ if } (x, y) \in \Omega_k. \quad (5.25)$$

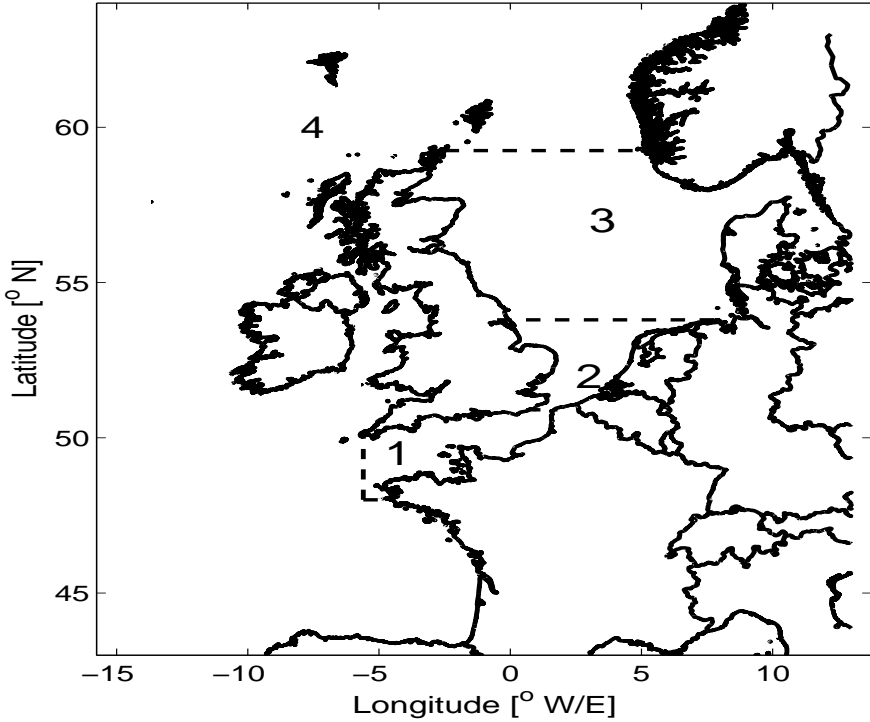


Figure 5.6: The four subdomains Ω_k of the DCSM used in Experiments 1 and 3

The parameters γ_k^b are treated as unknown parameters that are to be estimated. They act as correction for the $D_{x,y}$ in a subdomain Ω_k and leave the spatial dependence inside Ω_k unaltered. The adaptation in depth is relative. This is assumed keeping in mind that the model is quite sensitive for the shallow areas.

After the initial adjustments γ_k^b , an ensemble E of 132 snapshot vectors is generated using the forward model simulations. As explained in section 5.4.4, the snapshot vectors in the ensemble are collected for the period where data is available, i.e. from 1 January 2007 00:00 to 04 January 2007 24:00. After applying the scaling to each snapshot vector in an ensemble E , we are able to form a basis consisting of only 24 dominant eigenmodes that capture more than 97% of the relative energy. Figure 5.7 shows energy captured by POD modes for 132 snapshot vectors. So a reduced model is built using these 24 modes and finally operates on state space \mathfrak{R}^{24+4} . The low dimensional model is defined by assuming that the matrix \tilde{M} remains stationary throughout the experiment.

With this reduced model the approximate objective function \hat{J} is minimized in reduced space and the new values of the estimated variables γ^{op} are found. We have

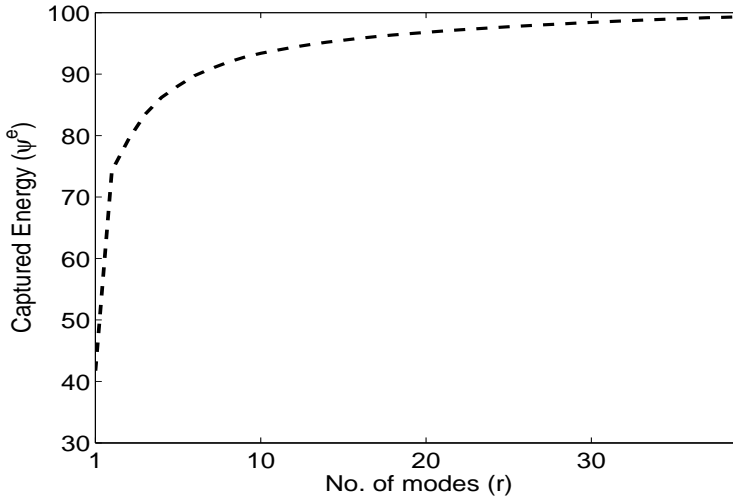


Figure 5.7: The energy captured by POD modes for an ensemble of 132 snapshots of the water level h , velocities u and v (Experiment 1)

stopped the inner minimization process and switched to a new outer iteration with the new set of parameters following the criterion μ . The objective function J is reduced by more than 50% with the updated parameters γ^{up} after the inner minimization (Figure 5.8). The RMSE for the calibration dataset has decreased from 21.75 cm to 14.74 cm after the 1st outer iteration β_1 of the minimization.

A new POD model is required in the outer iteration β_2 if the old POD model cannot substantially reduce the objective function J . Here the new POD model is constructed using γ^{up} . The POD modes however are the same as in the β_1 as explained in section 5.3.6. In this way, the generation of the new ensemble E and the solution of the eigenvalue problem to get the dominant eigenmodes p_i can be avoided. Again a significant improvement is observed in the objective function J after the inner minimization (see Figure 5.8) with the new values of γ^{up} . The RMSE for the calibration dataset has now decreased to 12.98 cm after the outer iteration β_2 of the minimization process.

Figure 5.9 shows the reduction of \hat{J} and the convergence of μ with respect to the inner iterations α for the outer iterations β_1 and β_2 . In the case of the quadratic objective function the quasi-Newton routine iterates approximately $(n^p + 1)$ times [33]. It can be seen from Figure 5.9b that the convergence criterion μ is achieved within the specified range of the quasi-Newton routine.

Three validation runs are performed, the 1st with the initial values for the calibration parameters γ^b and two with the updated parameters γ^{up} after each of the outer iterations β_1 and β_2 respectively. As explained in section 5.4.2, the validation experiments have to clarify whether these parameter adaptations really improve the model.

Table 5.1 shows the RMSE in the POD based calibration approach after each outer

iteration β_1 and β_2 for the tide gauge stations, separately for the different datasets used for the calibration and validation periods. The RMSE in Table 5.1 shows that significant improvements are found for different locations and different time periods. Table 5.1 also demonstrates that after β_2 the RMSE for the stations near Dutch coast is reduced to 12.05 cm as compared to the 27.15 cm with the initial parameter values γ^b .

<i>OuterIterations</i> (β)	<i>Calibration</i>	<i>Validation</i>	<i>Dutch</i>
<i>Initial</i>	21.75	19.94	27.15
β_1	14.74	13.22	15.24
β_2	12.98	11.72	12.05

Table 5.1: RMSE results for the minimization process with the 97% relative energy after 1st and 2nd outer iteration (Experiment 1)

Another reduced model is constructed that captures 90% of the relative energy. 15 POD modes are required in this case to build a reduced model that finally operates on state space \mathfrak{R}^{15+4} . Using this reduced model, the approximate objective function \hat{J} is minimized in reduced space and the new values of the estimated parameters γ^{up} are found. Table 5.2 shows the RMSE in the POD based calibration approach that captured 90% relative energy after each outer iteration β for the tide gauge stations, separately for each of the datasets used for the calibration and validation periods. The RMSE is again reduced significantly for both the calibration and the validation

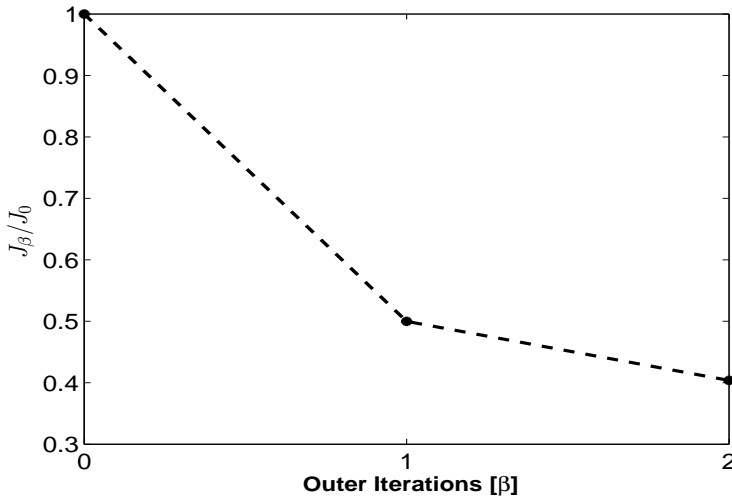


Figure 5.8: The reduction of the value of the objective function J at successive outer iterations β_1 and β_2 (Experiment 1)

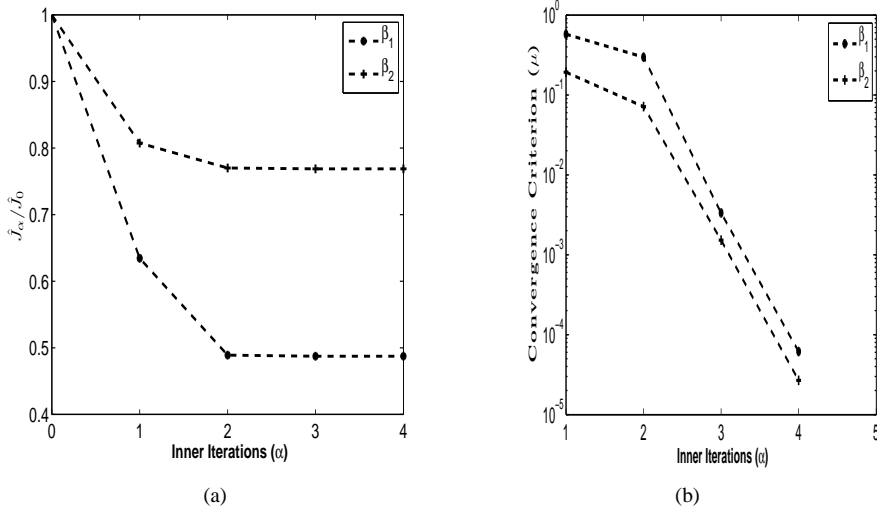


Figure 5.9: The values of a) $\hat{J}_\alpha / \hat{J}_0$ and b) convergence criterion μ at successive inner iterations α for the outer iterations β_1 and β_2 (Experiment 1)

<i>OuterIterations(β)</i>	<i>Calibration</i>	<i>Validation</i>	<i>Dutch</i>
<i>Initial</i>	21.75	19.94	27.15
β_1	15.44	13.85	16.42
β_2	13.80	12.42	13.57

Table 5.2: RMSE results for the minimization process with the 90% relative energy after 1st and 2nd outer iteration (Experiment 1)

datasets and time periods as in the previous case. The reduced model that captured more relative energy has performed slightly better since the eigenmodes p_i that are not included in the second case describe the part of the model variance which is relevant to the observations. Therefore the accuracy of the minimization procedure is really dependent on the size of the reduced model and also importance of the selected eigenmodes p_i with respect to the observations included in the calibration dataset [99].

In order to compare the results, a new POD model is constructed with the updated parameters γ^{up} after the outer iteration β_1 . Here a new ensemble E of 132 snapshot vectors is generated using the forward model simulations. We are able to form a basis consisting of 28 dominant eigenmodes that captures 97% of the relative energy (see Figure 5.10). Based on 28 modes the approximate objective function \hat{J} is minimized and the new values of the estimate variables γ^{up} are found.

Table 5.3 shows the comparison of the minimization results obtained from the

	<i>Calibration</i>	<i>Validation</i>	<i>Dutch</i>
<i>NewEnsemble</i>	12.33	11.41	11.13
<i>SameEnsemble</i>	12.98	11.72	12.05

Table 5.3: Results after 2nd outer iteration obtained by generating new ensemble and using the same ensemble as in 1st outer iteration (Experiment 1).

POD model that uses the POD modes p_i based on a new ensemble E in the outer iteration β_2 with the one that uses the same POD modes p_i as in β_1 . It is clear from the Table 5.3 that the results obtained by generating new ensemble E with new set of updated parameters γ^{up} are slightly better. But the computational cost of generating this ensemble E is much higher especially when the number of parameters is large.

In this experiment, we have found that the largest adjustments in the depth are in the southern part of the North Sea with the channel area, i.e. the Dutch coast and the east-coast of UK. This is because the initial RMSE in subdomain Ω_2 is quite big as compared to the other parts of the model domain. Also the data observation points are concentrated in this area of the model domain (see Figure 5.3).

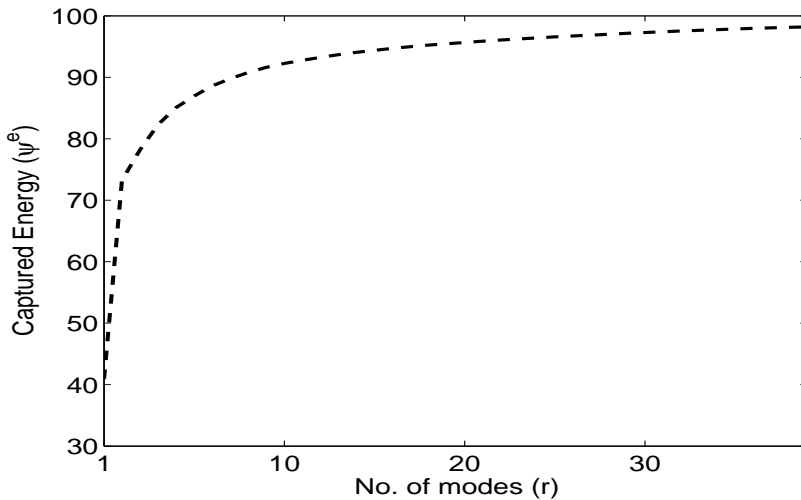


Figure 5.10: The energy captured by POD modes for a new ensemble of 132 snapshots of the water level h , velocities u and v in the outer iteration β_2 (Experiment 1)

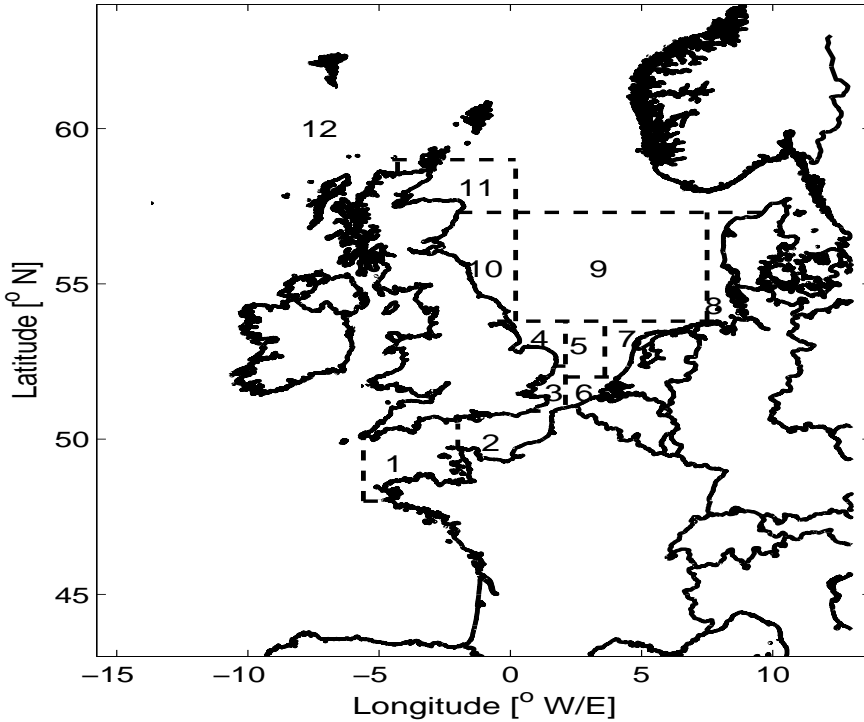


Figure 5.11: The 12 subdomains Ω_k of the DCSM used in Experiment 2

Experiment 2

In order to further improve the model results, the numerical domain Ω is divided into the 12 subdomains $\Omega_k, k = 1, \dots, 12$ (see Figure 5.11). The influence of the depth adjustments is quite significant and also depth has a strong local effect. This is one of the reasons for increasing the number of subdomains. Another reason for this subdivision is to separate the subdomains containing both deep and shallow areas (see Figure 5.11). As described earlier the data observation points are concentrated in the English Channel, so this region is divided into 5 subdomains to further improve the results by considering the local effects of the depth in each subdomain $\Omega_k, k = 3, \dots, 7$, in this area.

With this new selection of subdomains an ensemble E consisting of 396 snapshot vectors is generated for the same time period as in the previous experiment. Figure 5.12 shows energy captured by POD modes obtained from this ensemble E . A reduced model is then built using 49 dominant POD modes that captured 90% of the relative energy. The model finally operates on state space \mathcal{R}^{49+12} . These dominant POD modes are used in outer iterations β_3 and β_4 of the minimization process to reduce the approximate objective function \hat{J} .

<i>Outeriteration</i> (β)	<i>Calibration</i>	<i>Validation</i>	<i>Dutch</i>
<i>Initial</i> (β_2)	12.98	11.72	12.05
β_3	10.97	9.87	9.15
β_4	10.55	9.86	9.20

Table 5.4: RMSE results for the minimization process with the 90% relative energy after 3rd and 4th outer iterations (Experiment 2)

The initial values of the depth parameters used here are the values obtained after β_2 . Figure 5.13 shows the reduction of the objective function J in the outer iterations β_3 and β_4 . The objective function J is reduced by more than 30% after the β_4 of the minimization process as compared to the objective function J at the start of this experiment. The overall RMSE has decreased from 12.98 cm to 10.55 cm.

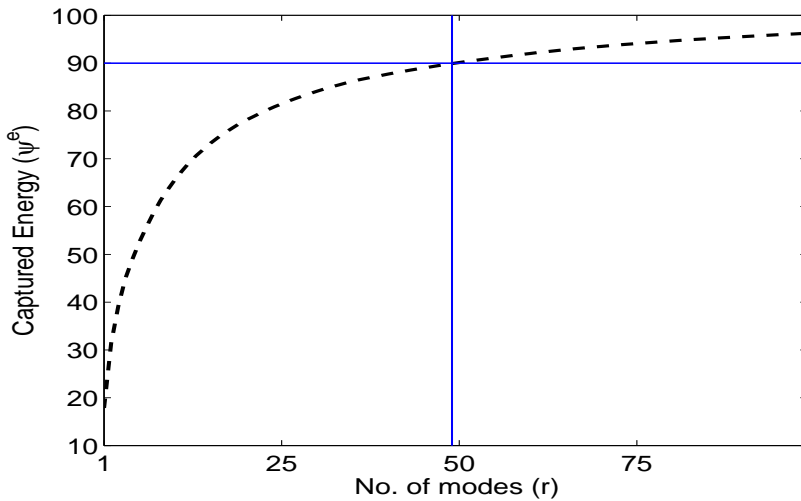


Figure 5.12: The energy captured by POD modes for an ensemble of 396 snapshots of the water level h , velocities u and v in the 3rd outer iteration β_3 (Experiment 2)

Table 5.4 shows the RMSE in the POD based calibration approach after the outer iterations β_3 and β_4 of the minimization process for the tide gauge stations, separately for the datasets used for the calibration and validation periods. The RMSE for the Dutch dataset of the tide gauge stations along the Dutch coast for the validation period is reduced to 9.20 after β_4 which is in acceptable range now.

The larger adjustments in the bathymetry are again found along the east-coast of the UK. The preliminary bathymetry along the UK coast is still a question. It is believed that the prescribed bathymetry in this area of the model is not satisfactory

and that it has also a strong effect on the model results. The adjustments along the southern Dutch coast (subdomain Ω_6) are larger as compared to the rest of the Dutch coast (subdomain Ω_7). Compared to the initial bathymetry an overall adjustments of around 8 – 11% are applied along the Dutch coast. It has also been noticed that the deep water regions are not adjusted significantly. Also from a physical point of view, the estimation of the depth in subdomains containing deep and shallow areas is more or less the estimation of the depth in the shallow areas [85].

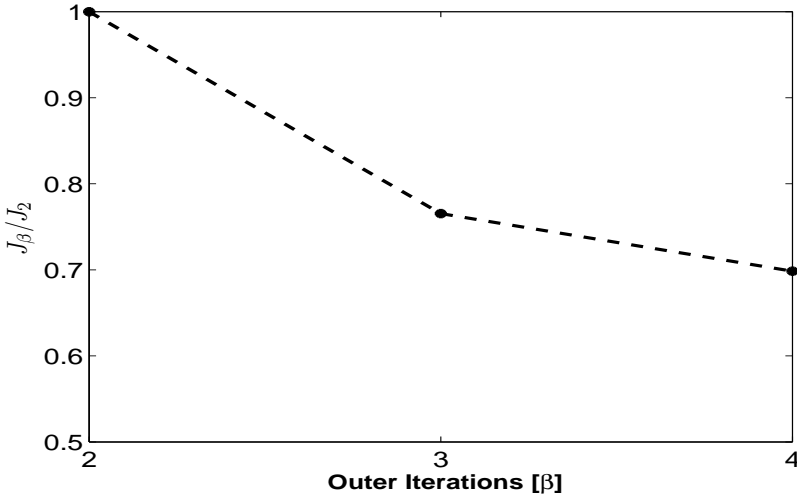


Figure 5.13: The reduction of value of the objective function J at successive outer iterations β_3 and β_4 (Experiment 2)

Experiment 3

As mentioned in section 2, a uniform value of $0.028s/m^{\frac{1}{3}}$ has been applied for the manning coefficient c_m . In this experiment we have calibrated the uniform manning coefficient. The numerical domain Ω is again divided into four subdomains $\Omega_k, k = 1, \dots, 4$, as in experiment 1 (see Figure 5.6). For each subdomain Ω_k a correction parameter γ_k^b is defined that is related to c_m^{new} by

$$c_m^{new} = c_m + \gamma_k^b, k = \{1, \dots, 4\}, \tag{5.26}$$

where c_m is the value used in the model.

With the above specification, an ensemble E of 132 snapshot vectors is generated from the perturbations $\frac{\partial M_i}{\partial \gamma_k}$ along the estimated parameters γ_k^b . Figure 5.14 shows energy captured by POD modes obtained from this ensemble E . A reduced model is then built using 32 dominant POD modes that capture 97% of the relative energy. The reduced model finally operates on state space \mathfrak{X}^{32+4} . These dominant POD modes are used in β_5 to reduce the approximate objective function \hat{J} .

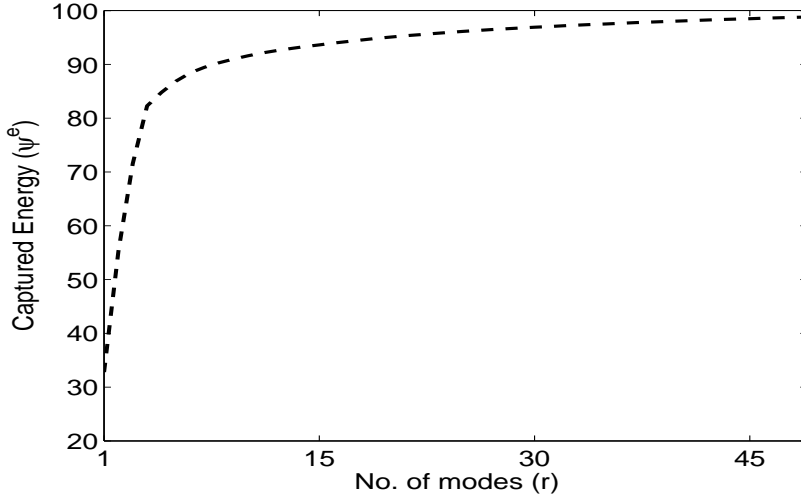


Figure 5.14: The POD modes captured energy for an ensemble of 132 snapshots of the water level h , velocities u and v in the outer iteration β_5 (Experiment 3)

<i>Outeriteration</i> (β)	<i>Calibration</i>	<i>Validation</i>	<i>Dutch</i>
<i>Initial</i> (β_4)	10.55	9.86	9.20
β_5	10.25	9.82	9.22

Table 5.5: RMSE results for the minimization process with the 97% relative energy after 5th outer iteration (Experiment 3)

Figure 5.15 shows the reduction of \hat{J} and the convergence of μ after each inner iteration α for the outer iteration β_5 . Although \hat{J} is reduced, the reduction is not very significant as compared to the experiments with depth parameters. This is because the influence of the depth adjustments is usually far greater than the adjustments to the bottom friction coefficient. Moreover, the manning coefficient is already being calibrated manually. Nevertheless any further reduction in objective function J is considered to be an improvement in the model results.

Table 5.5 presents the RMSE obtained in the β_5 of the POD based calibration approach, separately for the datasets used for the calibration and validation periods. The results show that the POD based calibration procedure reduces the RMSE for the datasets of the calibration period, but for the validation period there is no further improvement.

5.4.7 Discussion on results

Five outer iterations β have been performed during the calibration process. Figure 5.16 summarizes the RMSE after each outer iteration β for the datasets used for the calibration and the validation periods respectively. The POD based calibration approach, reduces the RMS values of the water levels h for the tide gauge stations for both the calibration and the validation periods with similar magnitude, except for the the outer iterations β_4 and β_5 where there is not much improvement for the datasets of the validation periods.

Figure 5.17 presents the mean value of the total depth adaptation after each outer iteration. The depth is adapted significantly in each outer iteration of the minimization process.

Figure 5.18 presents water levels h at the two tide gauge stations Delfzijl and Vlissingen along the north and south of the Dutch coast respectively for the period of 15 January 2007 00:00 - 16 January 2007 24:00. These time series refer to water levels obtained from observations, forecast using deterministic model without data assimilation and forecast with data assimilation after β_4 respectively. The figure demonstrates that the POD based calibration approach significantly reduces the differences between forecast time series with data assimilation and the observations as compared to the differences between the forecast time series without data assimilation and the observations. It is also clear from the Figure 5.18 that both the effects, phase shift and amplitude are compensated.

Figure 5.19 shows RMSE of water level at the selected stations along the Dutch coast for the validation period with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1, β_3 and β_5 respectively. As explained

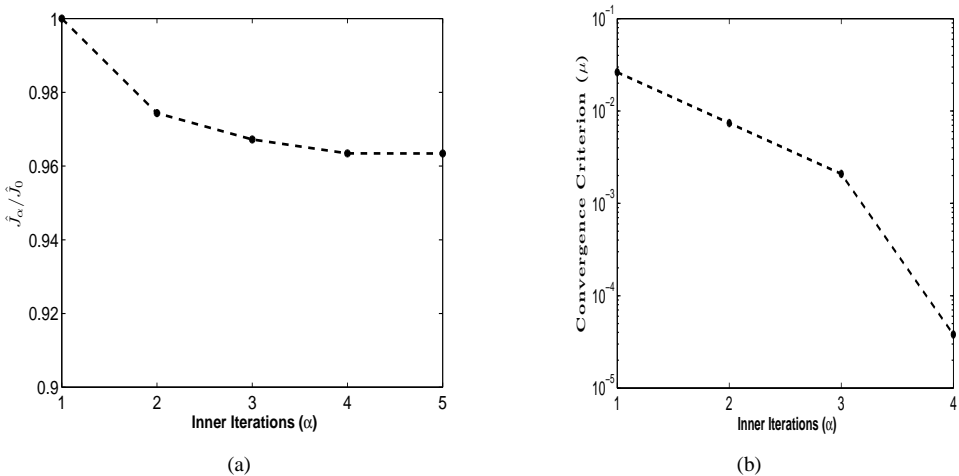


Figure 5.15: The values of a) $\hat{J}_\alpha / \hat{J}_0$ and b) convergence criterion μ at successive inner iterations α for the outer iteration β_5 (Experiment 3)

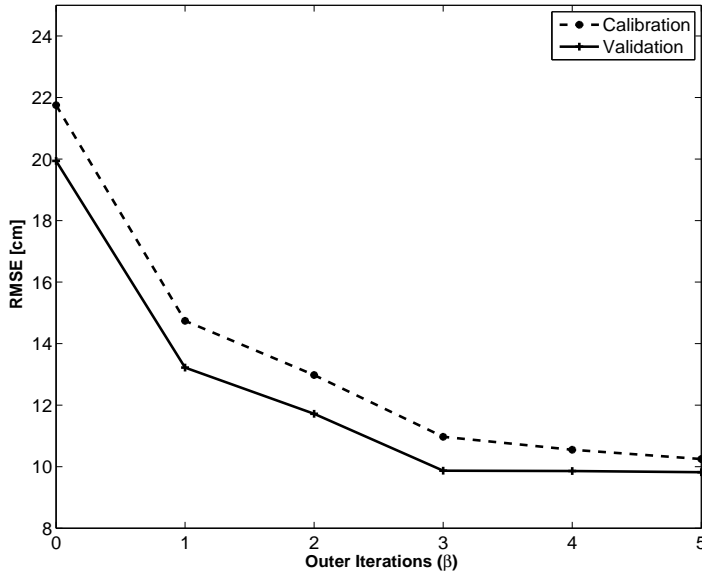


Figure 5.16: RMSE for the calibration and validation datasets after each outer iterations β of the minimization process

earlier, the target of the calibration of the depth and bottom friction parameters is to optimize the model for reproduction of the astronomical tide especially for the stations along Dutch coast. The Figure 5.19 demonstrates that the POD based calibration procedure significantly reduces the RMS values of the water level errors at all the tide gauge stations along the Dutch coast except for the station WESTTSLG where the improvement in RMSE is not significant.

Although the calibration is done in time domain, its important to analyze the performance of the model in frequency domain after calibration. Figure 5.20 illustrates the model performance after the calibration by means of the VD metric for the seven important constituents for the three selected locations used in the calibration for the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1, β_3 and β_5 respectively. The figure shows large improvements especially in the constituents M2 and M4. The model performance is further illustrated by Figure 5.21, in which the RMS(VD) is shown for the seven important constituents for the three datasets of the tide gauge stations used for the calibration and the validation periods for the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1, β_3 and β_5 respectively. The estimation procedure reduces the RMS values for all the constituents for all the datasets used for the calibration and validation periods.

The computational cost of the calibration experiments are expressed in terms of

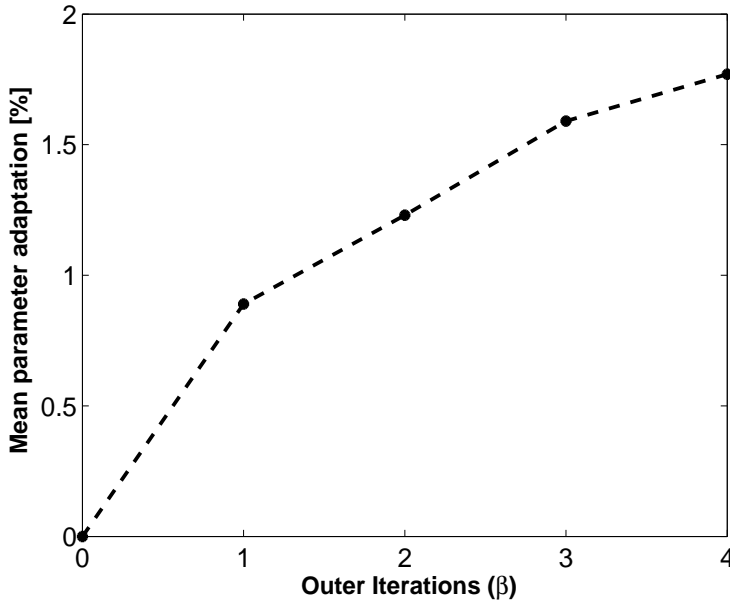
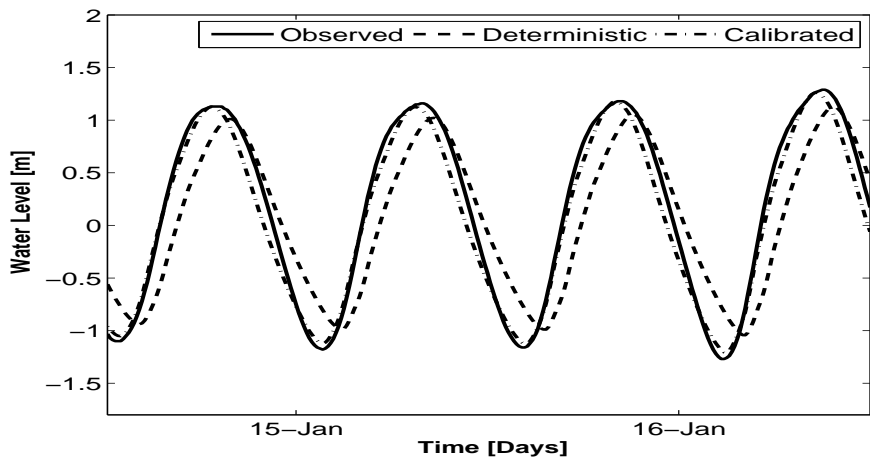


Figure 5.17: The mean value of the total depth adaptation after each outer iteration

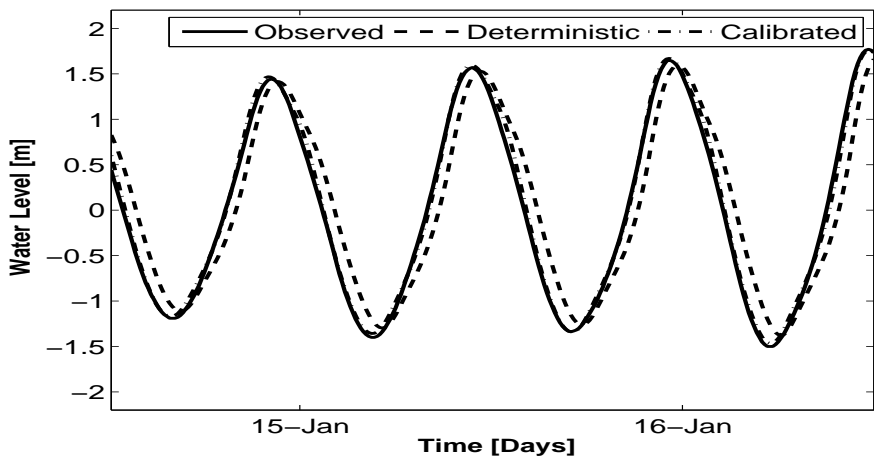
the number of simulations with the original model. Table 5.6 presents the computational costs of the calibration experiments. Four parameters are estimated during the outer iterations β_1, β_2 and β_5 (experiment 1 and 3), while 12 parameters are estimated during the outer iterations β_3 and β_4 (experiment 2) of the minimization process. In the outer iteration β_1 one forward model simulation is required for the calibration period, i.e. from 28 December 2006 to 30 January 2007, to obtain the initial value of the objective function J . As four parameters are estimated during β_1 , so four forward model simulations are performed from 28 December 2006 to 04 January 2007 to obtain an ensemble of the perturbations $\frac{\partial M_i}{\partial \gamma_k}$ along γ_k . The snapshots are collected for last four days only, since observations are available for this period. As the number of snapshots chosen is 132 (every three hours), the computational time to solve eigenvalue problem and to construct the reduced model is negligible. Combined with four estimated parameters, the reduced model simulates a reduced state within the dimension of a subspace \mathfrak{R}^{28} instead of the original state space $\sim \mathfrak{R}^{3 \times 10^6}$. Similarly the cost of optimization in the reduced space is negligible and eventually the time required to estimate 4 parameters in the outer iteration β_1 is equal to approximately 2.2 forward model simulations.

An ensemble E of the forward model simulations is obtained in the outer iterations β_1, β_3 and β_5 while in the outer iterations β_2 and β_4 the same ensembles are used as in the case of β_1 and β_3 respectively. In this way the computational cost of generating a new ensemble E and solving the eigenvalue problem to get the dominant POD modes

can be avoided in the outer iterations β_2 and β_4 as shown in Table 5.6. Table 5.6 also summarizes the computational cost for each outer iteration β of the POD based calibration approach. So approximately only 11 full forward model simulations are required to fine tune the model with respect to 16 depth and 4 bottom friction coefficients.



(a)



(b)

Figure 5.18: Water level timeseries for the period from 15 January 2007 00:00 - 16 January 2007 24:00 obtained from measurement data (observations), forecast using deterministic model without data assimilation and forecast with data assimilation (calibrated) after β_4 respectively at the two tide gauge stations a) Delfzijl and b) Vlissingen

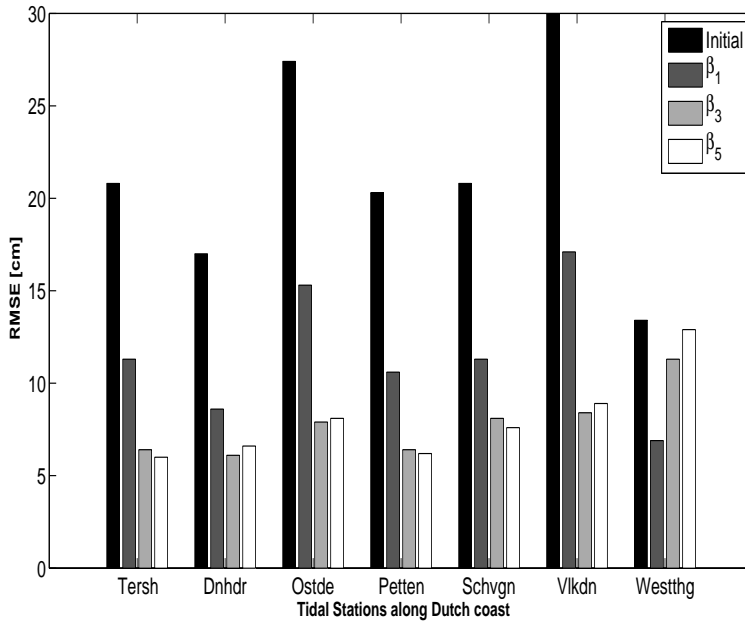


Figure 5.19: RMS of water level at the selected tide gauge stations along the Dutch coast with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1 , β_3 and β_5 respectively

Outeriteration(β)	Parameters	NewEnsemble	PODmodes	No.of simulations
β_1	4	Yes	24	~ 2.2
β_2	4	No	24	~ 1.1
β_3	12	Yes	49	~ 4.2
β_4	12	No	49	~ 1.1
β_5	4	Yes	32	~ 2.2
Total				~ 11

Table 5.6: Computational costs of the calibration experiments after each outer iteration β of the minimization process

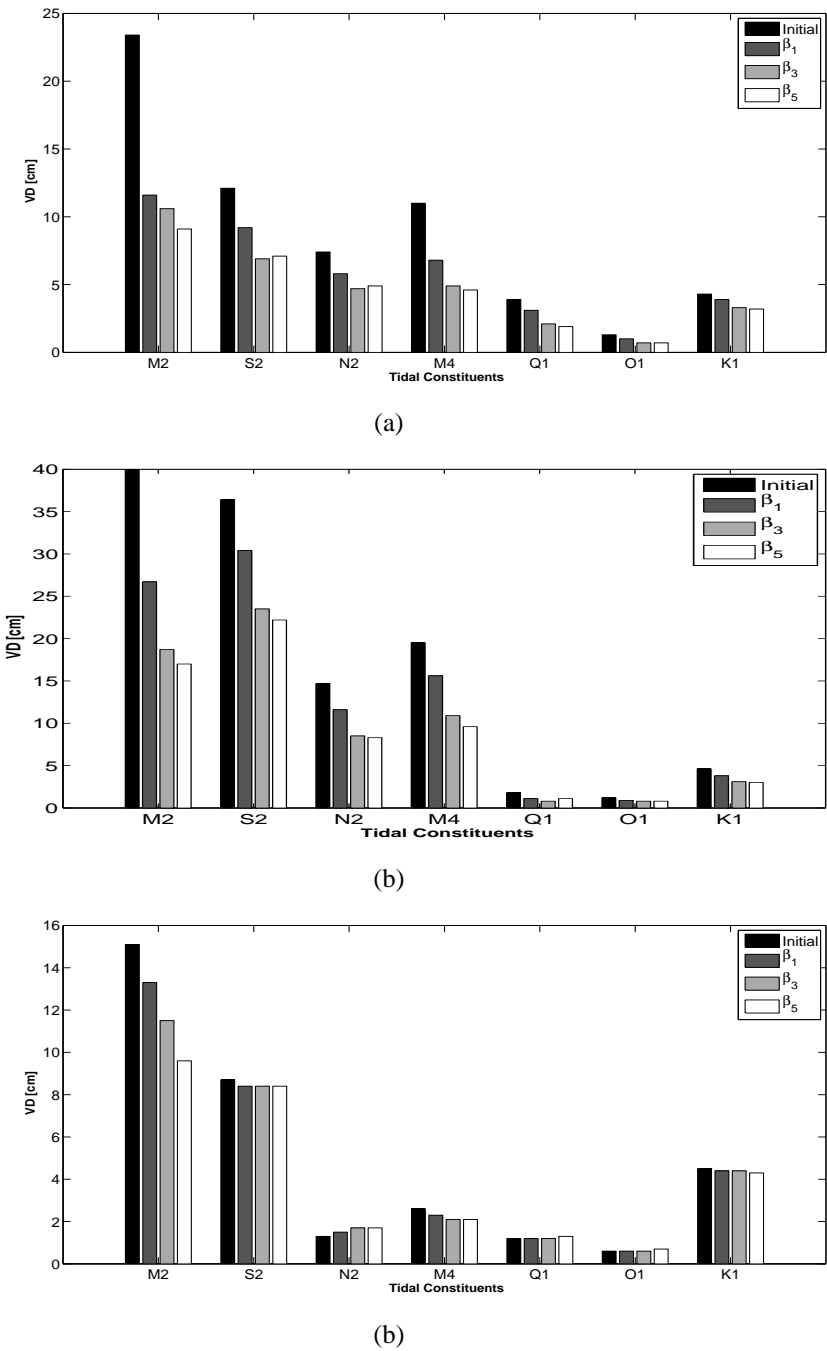
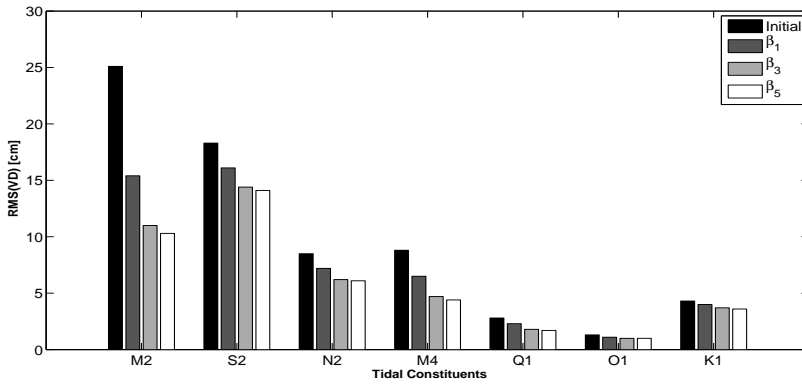
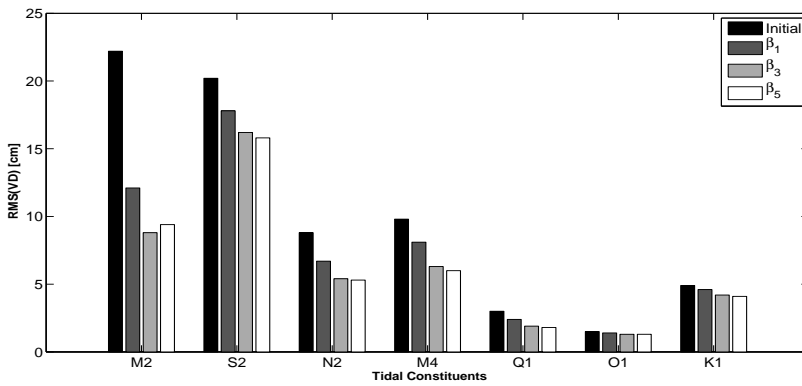


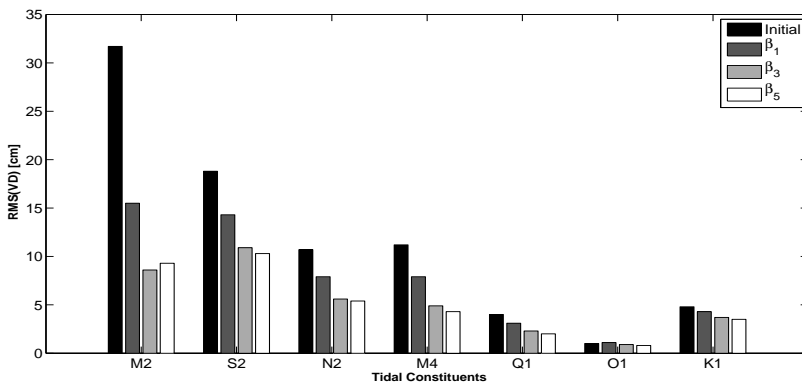
Figure 5.20: Model performance for a) Hoek van Holland; b) Dover; c) Wick expressed as vector differences of the 7 important constituents with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1 , β_3 and β_5 respectively



(a)



(b)



(c)

Figure 5.21: Model performance for the datasets a) Calibration; b) Validation; c) Dutch expressed as RMS(VD) for the 7 important constituents with the initial values of the parameters and with the updated parameters γ^{up} after the outer iterations β_1 , β_3 and β_5 respectively

5.5 Conclusions

The adjoint method is a powerful tool for sensitivity analysis and model calibration, but it is laborious to implement adjoint model for the computation of the gradient for large scale systems. The model-reduced approach presented here is used to simplify this problem using a projection based on POD model reduction method. The present approach is designed to approximate the data assimilation system in a restricted space while retaining its essential properties. By using restriction and prolongation operators the method fits into the theory of incremental variational data assimilation. Compared to the classical adjoint method, the minimization in reduced space converges faster due to better condition number of the reduced Hessian.

In this chapter, the POD based calibration approach has been used to calibrate the two-dimensional large-scale shallow water flow model, the new DCSM, defined over the entire European continental shelf. The method has been used to calibrate the new DCSM with respect to the bathymetry and a space-varying Chezy coefficient. The results show that the calibration method performs very efficiently. A POD reduced model of much smaller size is constructed instead of original model with state space $\sim \mathfrak{R}^{3 \times 10^6}$. The RMS errors for the tide gauge stations used for both calibration and validation periods have improved significantly with an overall improvement of more than 50% is observed after the calibration in comparison with the initial model.

The computational costs of the method are dominated by the generation of an ensemble of forward model simulations. The simulation period of the ensemble is equivalent to the timescale of the original model. Here an accurate reduced model is obtained from an ensemble with a relatively short simulation period of first four days that is used for calibration over the whole calibration period of one month.

To achieve the convergence, the method needs to be updated at each outer iteration β by constructing a new POD model by generating an ensemble of forward model simulations. We have found in the present study that it is not always required to use the new ensemble in each outer iteration β to construct the new POD based reduced-order model. Instead, the same ensemble can be used in several outer iterations. The results also demonstrate that in total only 11 full model simulations are required to calibrate the DCSM with 20 degrees of freedom. Thus the POD calibration method offers a very efficient minimization technique compared to the classical adjoint method without the burden of implementation of the adjoint.

Parameter Estimation using Simultaneous Perturbation¹

6.1 Introduction

Most efficient optimization algorithms require a gradient of the objective function. This usually requires the implementation of the adjoint code for the computation of the gradient of the objective function. The adjoint method aims at adjusting a number of unknown control parameters on the basis of given data. The control parameters might be model initial conditions or model parameters [88, 89]. A sizeable amount of research on adjoint parameter estimation was carried out in the last 30 years in fields such as meteorology, petroleum reservoirs and oceanography for instance by [77], [12], [19], [20], [91], [87], [53], [92], [43].

One of the drawbacks of the adjoint method is the programming effort required for the implementation of the adjoint model. Research has recently been carried out on automatic generation of computer code for the adjoint, and adjoint compilers have now become available (see [47]). Even with the use of these adjoint compilers, this is a huge programming effort, that hampers new applications of the method.

Vermeulen and Heemink [99] proposed a method based on Proper orthogonal decomposition (POD) which shifts the minimization into lower dimensional space and avoids the implementation of the adjoint of the tangent linear approximation of the original nonlinear model. Due to the linear character of the POD based reduced model its adjoint can be implemented easily and the minimization problem is solved completely in reduced space with very low computational cost. Recently [6, 9] applied this POD based calibration method for the estimation of depth values and bottom friction coefficients for a very large scale tidal model.

This chapter focuses on a method referred to as simultaneous perturbation stochastic approximation (SPSA) method. This method can be easily combined with any numerical model to do automatic calibration. For the calibration of numerical tidal

¹This chapter is a slightly revised version of [10]

model, the SPSA algorithm would require only the water level data predicted from the given model. SPSA is stochastic offspring of the Kiefer-Wolfowitz algorithm [50] commonly referred as finite difference stochastic approximation (FDSA) method. This algorithm uses objective function evaluations to obtain the gradient approximations. Each individual model parameter is perturbed one at a time and the partial derivatives of the objective function with respect to the each parameter is estimated by a divided difference based on the standard Taylor series approximation of a partial derivative. This approximation of each partial derivative involved in the gradient of the objective function requires at least one new evaluation of the objective function, thus this method is not feasible for automated calibration when we have large number of parameters.

The SPSA method uses stochastic simultaneous perturbation of all model parameters to generate a search at each iteration. SPSA is based on a highly efficient and easily implemented simultaneous perturbation approximation to the gradient. This gradient approximation for the central difference method uses only two objective function evaluation independent of the number of parameters being optimized. The SPSA algorithm has gathered a great deal of interest over the last decade and has been used for a variety of applications [45], [80], [81], [35], [34], [103]. As a result of the stochastic perturbation, the calculated gradient is also stochastic, however the expectation of the stochastic gradient is the true gradient [34]. So one would expect that the performance of the basic SPSA algorithm to be similar to the performance of steepest descent.

The gradient based algorithms are faster to converge than any objective function based gradient approximations such as SPSA algorithm when speed is measure in terms of the number of iterations. The total cost to achieve effective convergence depends not only on the number of iterations required, but also on the cost needed to perform these iterations, which is typically greater in gradient based algorithms. This cost may include greater computational burden and resources [5, 9], additional human effort required for determining and coding gradients.

Accurate sea water level forecasting is crucial in the Netherlands. This is mainly because large areas of the land lie below sea level. Forecast are made to support storm surge flood warning system. Timely water level forecasts are necessary to support the decision of the proper closure of the movable storm surge barriers in the Eastern Scheldt and the New Waterway. The surge is predicted by using numerical hydrodynamics model, the Dutch continental shelf model (DCSM) [82, 94]. Performance of the DCSM regarding to storm surges is influenced by its performance in forecasting the astronomical tides. Using inverse modelling techniques, these tidal data can be used to improve the model results. SPSA algorithm is applied here for the estimation of depth values in the tidal model DCSM of the entire European continental shelf. Experiments are performed with both simulated and real data to see the performance of the SPSA algorithm in terms of the accuracy of the estimates and computational efficiency.

The chapter is organized as follows. Section 2 describes the SPSA algorithm. The following section contains results from experiments with the model DCSM, to estimate the water depth. The paper concludes in section 4 by discussing the results.

6.2 Parameter estimation using SPSA

Consider a data assimilation problem for a general nonlinear dynamical system. The discrete system equation for the state vectors $X(t_{i+1}) \in \mathfrak{X}^n$ is given by;

$$X(t_{i+1}) = M_i[X(t_i), \gamma], \quad (6.1)$$

where M_i is nonlinear and deterministic dynamics operator that includes inputs and propagates the state from time t_i to time t_{i+1} , γ is vector of uncertain parameters which needs to be determined. Suppose now that we have imperfect observations $Y(t_i) \in \mathfrak{X}^q$ of the dynamical system (6.1) that are related to model state at time t_i through

$$Y(t_i) = HX(t_i) + \eta(t_i), \quad (6.2)$$

where $H : \mathfrak{X}^n \rightarrow \mathfrak{X}^{n^q}$ is linear observation operator that maps the model fields on observation space and $\eta(t_i)$ is unbiased random Gaussian error vector with covariance matrix R_i .

We assume that the difference between data and simulation results is only due to measurement errors and incorrectly prescribed model parameters. The problem of the estimation is then solved by directly minimizing the objective function J

$$J(\gamma) = \sum_i [Y(t_i) - H(X(t_i))]^T R_i^{-1} [Y(t_i) - H(X(t_i))] \quad (6.3)$$

with respect to the parameters γ satisfying the discrete nonlinear forecast model (6.1).

In the SPSA algorithm, we minimize the objective function $J(\gamma)$ using the iteration procedure

$$\gamma^{l+1} = \gamma^l - a_l \hat{g}_l(\gamma^l), \quad (6.4)$$

where $\hat{g}_l(\gamma^l)$ is a stochastic approximation of $\nabla J(\gamma^l)$, which denotes the gradient of the objective function with respect to γ evaluated at the old iterate, γ^l . if $\hat{g}_l(\gamma^l)$ is replaced by $\nabla J(\gamma^l)$, then 6.4 represents the steepest descent algorithm.

The stochastic gradient $\hat{g}_l(\gamma^l)$ in SPSA algorithm is calculated by the following procedure.

1. Define the n^p dimensional column vector Δ_l by

$$\Delta_l = [\Delta_{l,1}, \Delta_{l,2}, \dots, \Delta_{l,n^p}]^T, \quad (6.5)$$

and

$$\Delta_l^{-1} = [\Delta_{l,1}^{-1}, \Delta_{l,2}^{-1}, \dots, \Delta_{l,n^p}^{-1}]^T, \quad (6.6)$$

where $\Delta_{l,i}$, $i = 1, 2, \dots, n^p$ represents independent samples from the symmetric ± 1 Bernoulli distribution. This means that $+1$ or -1 are the only possible values that can be obtained for each $\Delta_{l,i}$. It also means that

$$\Delta_{l,i}^{-1} = \Delta_{l,i}, \quad (6.7)$$

and

$$E[\Delta_{l,i}^{-1}] = E[\Delta_{l,i}] = 0, \quad (6.8)$$

where E denotes the expectation.

2. Define a positive coefficient c_l and obtain two evaluations of the objective function $J(\gamma)$ based on the simultaneous perturbation around the current γ^l : $J(\gamma^l + c_l \Delta_l)$ and $J(\gamma^l - c_l \Delta_l)$.
3. A realization of the stochastic gradient is then calculated by using central difference approximation as

$$\hat{g}_l(\gamma^l) = \frac{J(\gamma^l + c_l \Delta_l) - J(\gamma^l - c_l \Delta_l)}{2c_l} \Delta_l^{-1} \quad (6.9)$$

Since Δ_l is a random vector, \hat{g}_l is also random vector. So by generating a sample of Δ_l , we generate a specific sample of \hat{g}_l . The FDSA algorithm involves computation of each component of ∇J by perturbing one model parameter at a time. If one does a one-sided approximation for each partial derivative involved in $\nabla J(\gamma^l)$, then computation of the gradient requires $n^p + 1$ evaluations of J for each iteration of the steepest descent algorithm. In contrast the SPSA requires only two evaluations of the objective function $J(\gamma^l + c_l \Delta_l)$ and $J(\gamma^l - c_l \Delta_l)$ at each iteration.

6.2.1 Choice of a_l and c_l

Returning to equations 6.4 and 6.9, we see that we have left to specify with a_l and c_l . These are specified here according to the guidelines given by Spall [80]. The relevant formulas for a_l and c_l are given by

$$a_l = \frac{a}{(A + l + 1)^{\hat{\alpha}}}, \quad (6.10)$$

and

$$c_l = \frac{c}{(l + 1)^{\hat{\beta}}}, \quad (6.11)$$

where a , c , A , $\hat{\alpha}$ and $\hat{\beta}$ are positive real numbers such that $0 < \hat{\alpha} \leq 1$, $\hat{\alpha} - \hat{\beta} < 0.5$ and $\hat{\alpha} > 2\hat{\beta}$. The given choices for $\hat{\alpha}$, $\hat{\beta}$ will ensure that the algorithm, equation (6.4) converges to a minimum of J in a stochastic sense (almost surely). The choice of a , c , A , $\hat{\alpha}$ and $\hat{\beta}$ is to some extent case dependent and it may require some experimentation to determine good values of these parameters. Although the asymptotically optimal values of $\hat{\alpha}$ and $\hat{\beta}$ are 1.0 and 1/6 respectively ([17]), but choosing smaller values e.g. $\hat{\alpha} = 0.602$ and $\hat{\beta} = 0.101$ ([80]) appear to be more effective in practice. One recommendation for A is to set A equal to 10% of the maximum number of iterations allowed.

The value of constant c should be chosen so that c is equal to the standard deviation of the noise in objective function J . If one has perfect objective function, then c should be chosen as small positive number.

6.2.2 Average stochastic gradient

One of the motivations for SPSA is that for a quadratic objective function such as J , the expectation of the stochastic gradient is the true gradient ([34]), i.e.

$$E[\hat{g}_l(\gamma^l)] = \overline{\hat{g}_l(\gamma^l)} = \nabla J(\gamma^l), \quad (6.12)$$

where $\overline{\hat{g}_l(\gamma^l)}$ is defined as

$$\overline{\hat{g}_l(\gamma^l)} = \frac{1}{N} \sum_{i=1}^N \hat{g}_l(\gamma^l), \quad (6.13)$$

with each $\hat{g}_l(\gamma^l)$ is obtained from equation 6.9 using N different samples of Δ_l . Due to the relationship given in equation 6.15 one would hope that SPSA would have convergence properties similar to those of steepest descent in terms of the number of iterations required to reduce the objective function J to a certain level. In this case SPSA could be much more efficient than the steepest descent algorithm.

6.3 Numerical experiments

The algorithm presented in Section 6.2 was tested in a calibration experiment using the model DCSM. The DCSM is an operational storm surge model, used in the Netherlands for real-time storm surge prediction in North sea.

6.3.1 Experiment 1

The DCSM model used in this experiment covers an area in the north-east European continental shelf, i.e. $12^\circ W$ to $13^\circ E$ and $48^\circ N$ to $62^\circ N$, as shown in Figure 6.1. The resolution of the spherical grid is $1/8^\circ \times 1/12^\circ$, which is approximately 8×8 km. With this configuration there are 201×173 grid with 19809 computational grid points. The time step is $\Delta t = 10$ minutes.

In chapter 4 several twin experiments were performed to estimate depth values in the English channel of the model DCSM (see Figure 6.2) using POD based calibration method. Similar twin experiment was performed as in Section 4.5.1 to estimate depth values using SPSA algorithm. The numerical domain Ω was divided into seven subdomains $\Omega_k, k = 1, \dots, 7$. For each subdomain Ω_k , a correction parameters γ_k^b is defined that is related to D_{n_1, n_2} by:

$$D_{n_1, n_2} = D_{n_1, n_2}^b + \gamma_k^b, \mathbf{if}(n_1, n_2) \in \Omega_k \quad (6.14)$$

with D_{n_1, n_2}^b , the initial value. The parameters γ_k^b were treated as unknown parameters. They act as a correction for the mean level of the D_{n_1, n_2} in a sub domain Ω_k and leave the spatial dependence inside Ω_k unaltered. Seven observation points were included in the assimilation, two of which are located along the east coast of the UK, two along the the Dutch coast and one at the Belgium coast (see Figure 6.1). The truth model was run for a period of 15 days from 13 December 1997 00:00 to 27 December 1997 24:00 with the specification of water depth D_{n_1, n_2}^b as used in the operational DCSM to generate artificial data at the assimilation stations. The first two days were used to properly initialize the simulations and set of observations Y of computed water levels h were collected for last 13 days at an interval of every ten minutes in seven selected assimilation grid points, which coincide with the points where data are observed in reality. The observations were assumed to be perfect. This assumption was made in order to see how close the estimate to the truth. 5[m] was added in D_{n_1, n_2}^b at all the grid points in domain Ω to get the initial adjustments γ_k^b .

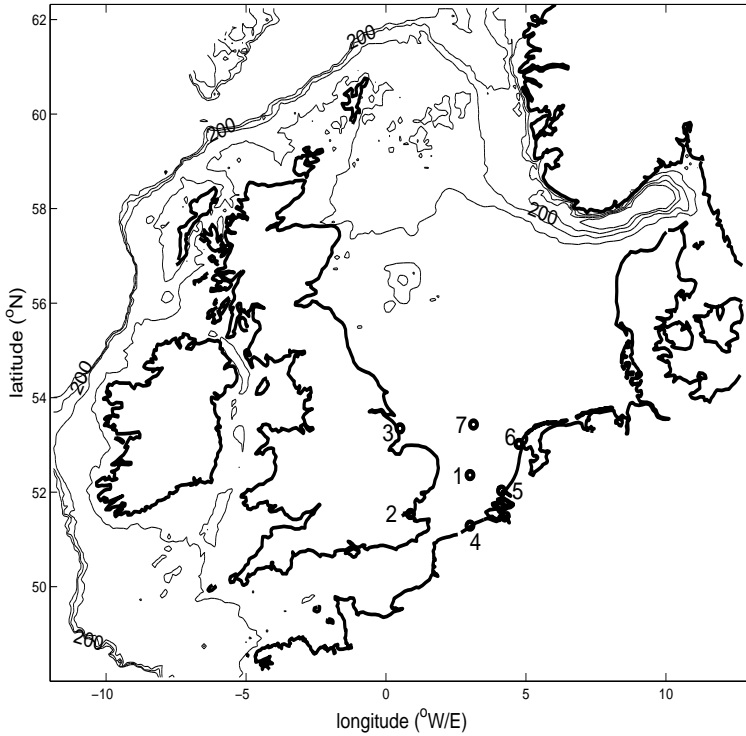


Figure 6.1: DCSM area with calibration stations: 1. N51, 2. Southend, 3. Innerdowsing, 4. Oostende, 5. H.v.Holland, 6. Den Helder, 7. N4

For the SPSA optimization algorithm, two methods were applied to calculate the stochastic gradient. In the first method, the stochastic gradient $\hat{g}_l(\gamma^l)$ was computed according to equation 6.9. In the second method the gradient was computed by equation 6.13 referred as Average SPSA where expectation is taken over two independent stochastic gradients.

The values of a , c , A , $\hat{\alpha}$ and $\hat{\beta}$ were obtained according to the guidelines given in Section 6.2.1. These values were determined as best from several forward model simulations. The iteration cycle for the SPSA algorithm was aborted when the value of the objective function J did not change for the last three iterations of the minimization process.[103].

Figure 6.3 shows a plot of the objective function versus number of iterations β for the two implementations of the SPSA algorithms compared with the steepest descent and the POD based calibration methods. Note that the gradient used in the steepest descent algorithm was obtained from the finite difference method using one-sided

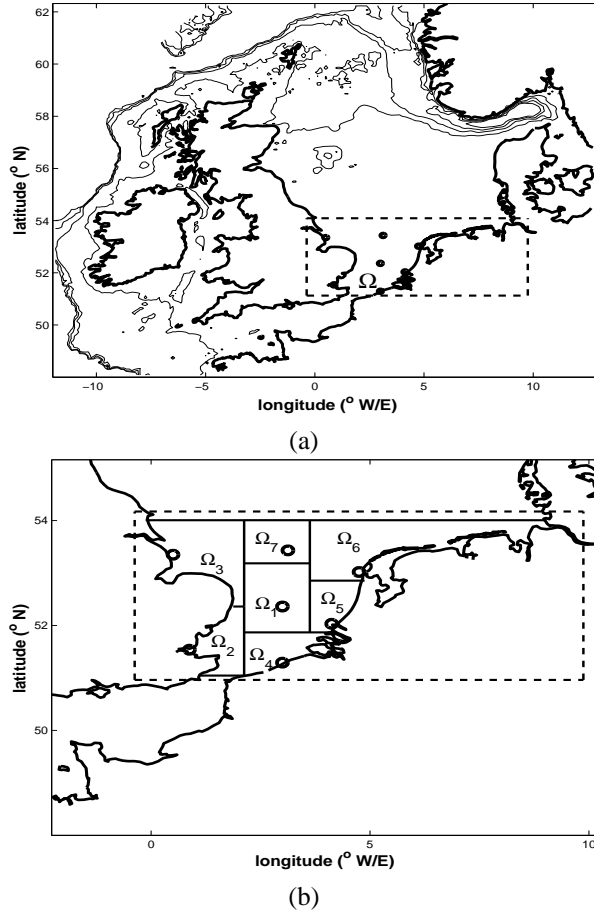


Figure 6.2: (a) Shows the domain Ω (dashed rectangle), of DCSM. (b) Shows the subdomains Ω_1 , Ω_2 , Ω_3 , Ω_4 , Ω_5 , Ω_6 and Ω_7

perturbation. The graph shows that both SPSA and Average SPSA give comparable results, although for Average SPSA the decrease in the objective function J is more at early iterations. Also the rate of convergence of Average SPSA is slightly better than the SPSA. However, in terms of rate of convergence both SPSA and Average SPSA are less efficient than steepest descent and POD based calibration methods. The steepest descent algorithm converges in 10 iterations as compared to 20 and 15 iterations in SPSA and Average SPSA respectively. However, the cost of single iteration in SPSA algorithm is far less than the steepest descent algorithm.

For all the algorithms, there was a significant improvements in parameters for regions coinciding with the UK, Dutch and Belgian coast, but there was not much improvement in deep water regions Ω_1 and Ω_7 . Since the subdomains containing deep areas are less sensitive as compared to the subdomains containing shallow areas, so it

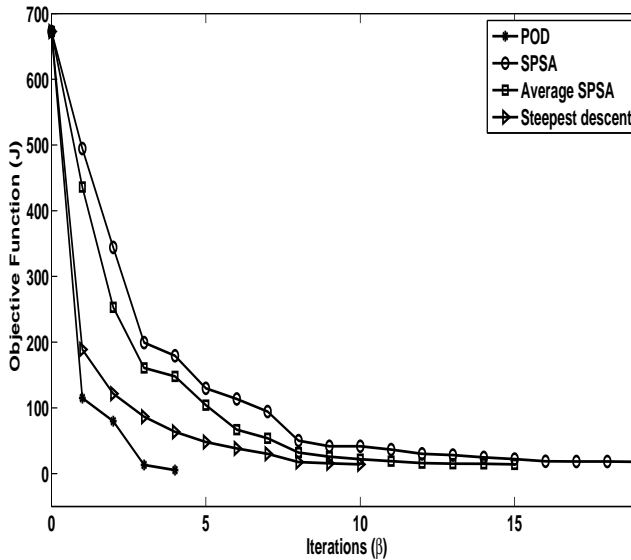


Figure 6.3: Successive iterations β of the minimization process

is much difficult to estimate γ_k in regions Ω_1 and Ω_7 .

Table 6.1 lists the measure (ζ) between the updated estimated parameters γ^{up} obtained after calibration with different optimization algorithms and the true parameter estimate γ^t . The measure is defined as the two norm of the difference between estimated parameters γ^{up} obtained after optimization and the true parameter estimate γ^t divided by the norm of the true parameter estimate γ^t ([34]).

$$\zeta = \frac{\|\gamma^{up} - \gamma^t\|_2}{\|\gamma^t\|_2} \quad (6.15)$$

By this measure, steepest descent (21%) performed the best followed by Average SPSA (29%) and SPSA (35%). Since the stochastic gradient in the SPSA algorithm is based on two perturbations of the independent random samples, it is more likely that the SPSA algorithm improves more sensitive areas. The table also lists the same measure for shallow regions. In this case, all the algorithms steepest descent (6.49%), Average SPSA (6.29%) and SPSA (9.95%) performed very well. Here Average SPSA matched the performance of steepest descent algorithm. In Average SPSA the gradient was the average of only two independent stochastic gradients. One would expect better performance by the inclusion of more stochastic gradients in Average SPSA.

Table 6.2 presents the RMSE between estimated parameters (γ^{up}) and the true parameters (γ^t) after iterations β_5 , β_{10} , β_{15} and β_{20} of SPSA algorithm for calibration stations and compares it with Average SPSA and steepest descent algorithms. The

ζ	SPSA	Average SPSA	Steepest Descent
All parameters	35.11%	29.27%	21.02%
Sensitive parameters	9.95%	6.29%	6.49%

Table 6.1: Comparison of estimated parameters to true parameters for the twin experiment

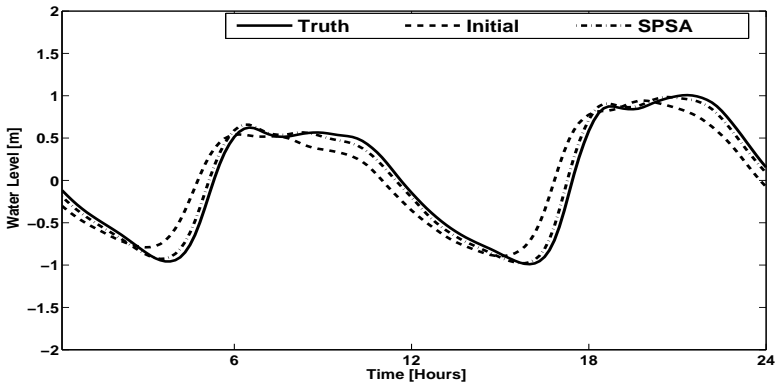
RMSE for SPSA algorithm after iteration β_5 is 9.95 compared to 8.92 and 6.05 in Average SPSA and steepest descent algorithm respectively. So SPSA and Average SPSA are comparable at this point. The RMSE for SPSA after 10 iterations is comparable to the RMSE of steepest descent method after only 5 iterations. Since the cost of one iteration of steepest descent is 8 model simulations compared to 3 model simulations in SPSA algorithm, SPSA is 1/4 times efficient than steepest descent at this point and one would expect SPSA to be more efficient if we have large number of parameters.

The RMSE with SPSA after β_{15} and average SPSA after β_{10} is similar. At this point the computational costs of both SPSA and Average SPSA are also comparable. It is also clear from the table 6.2 that the smallest RMSE value is achieved by steepest descent method in 10 iterations.

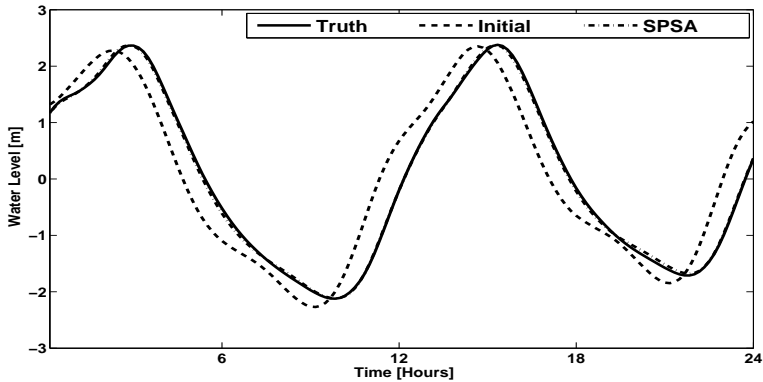
	SPSA (cm)	Average SPSA (cm)	Steepest Descent (cm)
Initial	22.80	22.80	22.80
β_5	9.95	8.92	6.05
β_{10}	5.63	4.09	2.91
β_{15}	4.10	3.27	-
β_{20}	3.55	-	-

Table 6.2: RMSE results for the minimization process after 5th, 10th, 15th and 20th iterations

Figure 6.4 presents water levels h at the two tide gauge stations Den Helder and Southend along the Dutch and English coasts respectively for the period from 18 December 1997 00:00 - 18 December 1997 24:00. These time series refer to water levels obtained from true values of the parameters, the initial values of the parameters and the estimated values of the parameters using SPSA algorithm respectively. The figure 6.4 demonstrates that the estimation methods significantly reduces the differences between time series obtained from initial parameters and the true parameters as compared to the differences between time series obtained from the estimated parameters and true parameters.



(a)



(b)

Figure 6.4: Water level timeseries for the period from 18 December 1997 00:00 - 18 December 1997 24:00 obtained from truth model, deterministic model with initial values of the estimated parameters and deterministic model after calibration respectively at the two tide gauge stations a) Den Helder and b) Southend

6.3.2 Experiment 2

The DCSM model used in this experiment is a newly designed large scale spherical grid model. This newly developed DCSM covers an area in the north-east European continental shelf, i.e. $15^{\circ}W$ to $13^{\circ}E$ and $43^{\circ}N$ to $64^{\circ}N$, as shown in (see Figure 6.5). The spherical grid has a uniform cell size of $1/40^{\circ}$ in east-west direction and $1/60^{\circ}$ in north-south direction which corresponds to a grid cell size of about $\sim 2 \times 2$ km. With this configuration there are 1120 grid cells in east-west direction and 1260 grid cells in north-south direction. The grid cells that include land are excluded from the model by the enclosures and the model contains 869544 computational grid points. The grid

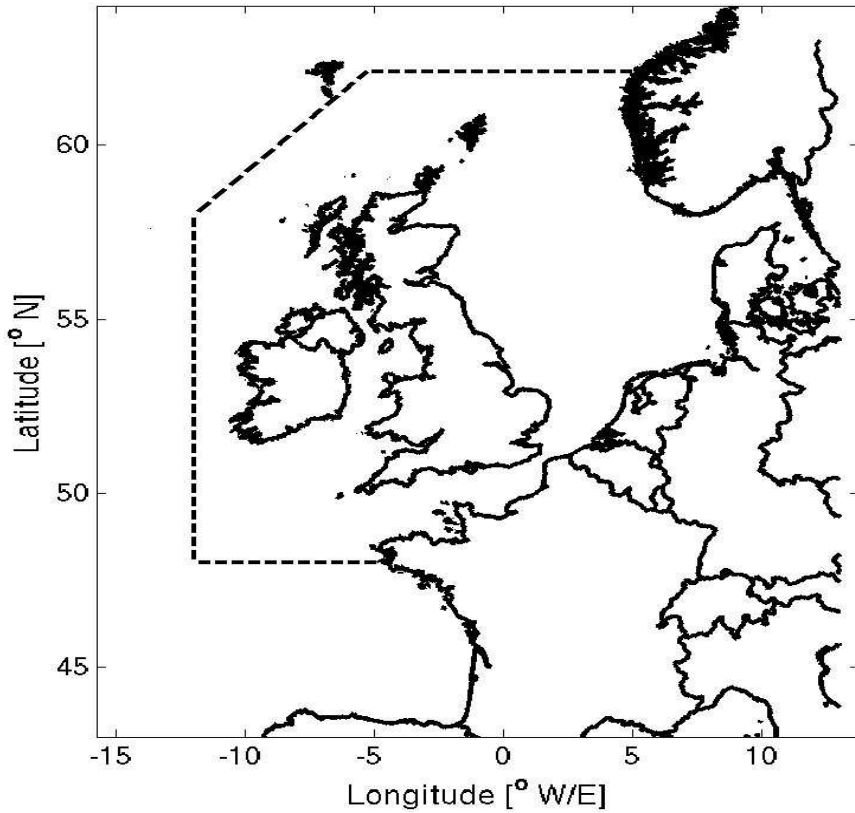


Figure 6.5: Newly developed hydrodynamic DCSM area. The dashed line represents the area of the operational DCSM extent

resolution of the spherical grid is factor five finer than the DCSM model grid used in the previous experiment. The idea was to perform numerical experiment with a very large scale model and with real data using SPSA algorithm.

The bathymetry of the model here is based on a NOOS gridded data set and for some areas in the model, ETOPO2 bathymetry data is interpolated on the computational grid. The dashed line in Figure 6.5 shows the comparison of the newly developed DCSM model area with the old DCSM. The model area of the newly developed DCSM is extended significantly in order to ensure that the open boundary conditions are located further away in deep water. A computational time step of 2 minutes has been applied.

The model performance can be assessed by comparing it to the measured (observed) dataset. The available data used here consists of two datasets of the tide gauge stations are used, namely,

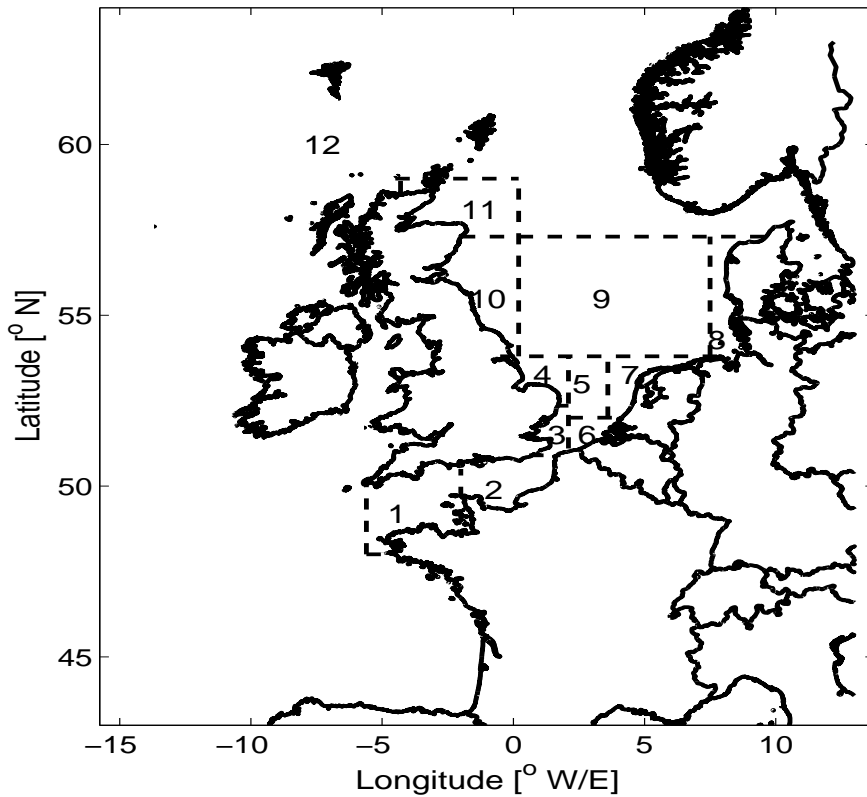


Figure 6.6: The 12 subdomains Ω_k of the DCSM used in Experiment 2

1. water level measurement data from the Dutch DONAR database,
2. BODC (British Oceanographic Data Center) offshore water level measurement data.

The target of the calibration of the parameters (i.e. depth values) is to optimize the model for its reproduction of the astronomical tide. The tide gauge data are therefore retrieved from the results of the harmonical analysis to exclude the meteorological influences.

Similar experiment was performed as in Section 5.4.6 to estimate depth values using SPSA algorithm. The numerical domain Ω was divided into the 12 subdomains $\Omega_k, k = 1, \dots, 12$ (see Figure 6.6). 50 water level locations were selected for the calibration experiment (see Figure 6.7). Observations obtained by the harmonic analysis from these 50 stations at every fifth time step (10 minutes) were used for the calibration experiments. The calibration runs were performed for the period from 28 December 2006 to 30 January 2007 (34 days). The first 4 days were used to properly

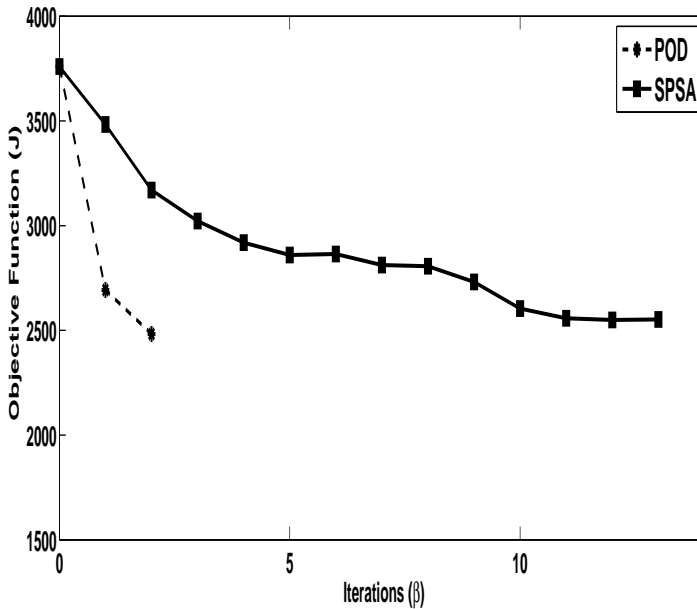


Figure 6.8: Successive iterations β of the minimization process

to 14 iterations with the SPSA respectively. However, the cost of single iteration in the POD based calibration method is much higher and is dependent on the number of parameters n^p and the POD modes r used to construct the reduced model [5]. So for this experiment the POD method required 13 initial simulations of the model to get the ensemble and then additional simulations of the model to construct the POD reduced model in each iteration β of the optimization process. The SPSA method on the other hand required only two objective function evaluations to compute the gradient in each iteration β of the optimization procedure. For this application the POD method is also fast since it is not needed to use a full simulations of the original model for the generation of the ensemble [9]. For both the experiments performed the SPSA algorithm converged in almost similar iterations although the number of parameters were different. So it is expected that the SPSA algorithm will work even with more parameters and the algorithm is independent of the number of the estimated parameters.

6.4 Conclusions

In the absence of adjoint model, the gradient is usually obtained by objective function evaluations to obtain the gradient approximations. Each individual model parameter is perturbed one at a time and the partial derivatives of the objective function with respect

to the each parameter is estimated. This method is not feasible for automated calibration when we large number of parameters are estimated. Simultaneous perturbation stochastic approximation (SPSA) method uses stochastic simultaneous perturbation of all model parameters to generate a search at each iteration. SPSA is based on a highly efficient and easily implemented simultaneous perturbation approximation to the gradient. This gradient approximation for the central difference method uses only two objective function evaluation independent of the number of parameters being optimized.

SPSA algorithm is applied to calibrate the model Dutch Continental Shelf Model (DCSM). The DCSM is an operational storm surge model, used in the Netherlands for real-time storm surge prediction in North sea. A number of calibration experiments was performed both with simulated and real data. The results from twin experiment showed that SPSA has a lower convergence rate than the steepest descent and POD based calibration methods. The steepest descent algorithm converged in 10 iterations as compared to 20 and 15 iterations in SPSA and Average SPSA respectively. However, the computational cost of single iteration in the steepest descent method is much higher and dependent on the number of parameters n^p . Although both SPSA and steepest descent methods converged to similar value of the objective function, none of the optimization algorithms achieved the expected reduction in the objective function.

The results from a very large scale tidal model and with real data showed that SPSA algorithm gives comparable results to POD based calibration method. The POD based calibration method converged in only 2 iterations as compared to 14 iterations with the SPSA respectively. The POD based calibration method though required 13 initial simulations of the original model to get the ensemble and then extra simulations to construct the POD reduced model in each iteration β of the optimization process. The SPSA method on the other hand required only two objective function evaluations to compute an approximation of the gradient in each iteration β of the optimization procedure independent of the number of estimated parameters. For this application the POD method is also fast since it is not needed to use a full simulations of the original model for the generation of the ensemble. Thus SPSA algorithm proved to be a promising optimization algorithm for model calibration for cases where adjoint code is not available for computing the gradient of the objective function.

Conclusions

Identifying uncertain parameters in large-scale numerical flow models can be done using the variational method. However, directly implementing the variational method would require the adjoint model, which requires highly complex computer code and maintenance and thus hampers its applications. This research has explored several methods for efficiently identifying uncertain parameters in a large-scale tidal model of entire European continental shelf which does not require the implementation of these complex adjoint code.

An estimation method based on model reduction is developed and investigated first for a simple 2D-advection diffusion model. We have developed estimation procedure for two projection based model reduction methods.

- Proper Orthogonal Decomposition(POD).
In POD based estimated procedure an ensemble of forward model simulations is used to determine the approximation of the covariance matrix of the model variability and the dominant eigenvectors of this matrix are used to define a model subspace. An approximate linear reduced model is obtained by projecting the original model onto this reduced subspace. The method is simple to implement.
- Balanced Proper Orthogonal Decomposition (BPOD).
In BPOD estimation procedure the model subspace is obtained while considering both inputs and outputs of the system. Thus both controllable and observable subspaces are considered while obtaining a low rank subspace for the reduced model. The presented method allows efficient computation of observable subspace when the number of outputs is large.

In both the estimation procedures the adjoint of the tangent linear model is replaced by the adjoint of this linear reduced forward model. The minimization process is carried out in reduced subspace and hence reduces the computational costs. The method and results from the experiments have been presented in Chapter 3. Numerical results from a simple pollution model of concentration $c(x,t)$ demonstrated that the POD based estimation approach successfully estimated the diffusion coefficient

for both advection dominated problems as for diffusion dominated problems. Another important message in that chapter, although lots of effort had been put in constructing a reduced order model by BPOD method, the minimization results demonstrated that both the POD and the BPOD methods performed similarly.

For a feasible approximation of the model variability, a POD based estimation approach is attractive as it avoids the implementation of adjoint code and save considerable computational effort for optimization compared to the usual gradient based iterative procedures where gradient is obtained from the original nonlinear model in each iteration. In Chapter 4 the POD based estimation method is used to calibrate numerical tidal models. Results from twin experiments showed that the POD based calibration method performed very efficiently to estimate depth values in the selected regions of the model domain. The computational costs of the method are dominated by the generation of an ensemble of forward model simulations. The simulation period of the ensemble is equivalent to the timescale of the original model. It is also found in that study that an accurate reduced model is obtained from an ensemble with a relatively short simulation period as compared to the calibration period and the POD based calibration offers an efficient method compared to the classical adjoint method.

As a next step, the POD based model reduced approach is used for the estimation of the water depth and space varying bottom friction coefficient values in a very large-scale DCSM model. The model used here is the recently designed large-scale spherical grid based water level model for the northwest European continental shelf (around 10^6 computational grid points). It covers a much larger deep water area than the operational DCSM and has a spatial resolution that is a factor 5 finer in both latitudinal and longitudinal directions. This is the first application of the POD based calibration method to a very large-scale model and with real data. To achieve convergence, the method needs to be updated by constructing a new POD model by generating an ensemble of forward model simulations with suboptimal estimated parameters. We have also found in the present study that a new ensemble is not always required with the updated parameters. Instead of defining a new model subspace of the leading eigenvectors by generating a new ensemble with suboptimal parameters, we can obtain a reduced model by projecting original model with the updated parameters onto the same subspace.

Results from numerical experiments showed that the calibration method again performed very efficiently. A POD reduced model of much smaller size is constructed instead of original model with state space $\sim \mathfrak{R}^{3 \times 10^6}$. The RMS errors for the tide gauge stations used for both calibration and validation periods have improved significantly with an overall improvement of more than 50% is observed after the calibration in comparison with the initial model. The results also demonstrate that in total only 11 full model simulations are required to calibrate the DCSM with 20 degrees of freedom. Thus the POD calibration method offers a very efficient minimization technique compared to the classical adjoint method without the burden of implementation of the adjoint.

In the absence of adjoint model, the gradient is usually obtained by objective function evaluations to obtain the gradient approximations. Each individual model parameter is perturbed one at a time and the partial derivatives of the objective function with respect to the each parameter is estimated. This method is not feasible for au-

tomated calibration when large number of parameters are estimated. Simultaneous perturbation stochastic approximation (SPSA) method uses stochastic simultaneous perturbation of all model parameters to generate a search at each iteration. SPSA is based on a highly efficient and easily implemented simultaneous perturbation approximation to the gradient. This gradient approximation for the central difference method uses only two objective function evaluation independent of the number of parameters being optimized. SPSA algorithm has applied to calibrate the model DCSM and results are presented in Chapter 6. The results from experiments showed that SPSA has a lower convergence rate than steepest descent and POD based calibration methods, however the computational cost in each iteration of the SPSA method is far less than these methods. The results demonstrated that the SPSA algorithm proved to be a promising optimization algorithm for model calibration for cases where adjoint code is not available for computing the gradient of the objective function.

Appendix A

The Reduced Adjoint State

Equation (5.18) introduces the adjoint model $\hat{v}(t_{i+1})$, that is elaborated here. The approximate objective function \hat{J} can be written as:

$$\hat{J}(\Delta\gamma) = \sum_{i=1}^m [\{\mathbf{Y}(t_i) - H(\mathbf{X}^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)]^T R_i^{-1} [\{\mathbf{Y}(t_i) - H(\mathbf{X}^b(t_i))\} - \hat{H}\xi(t_i, \Delta\gamma)] \quad (\text{A.1})$$

The reduced adjoint state variables \hat{v} are introduced in equation (A.1) as:

$$\hat{J}(\Delta\gamma) = \bar{J}(\Delta\gamma) + \sum_{i=1}^m \hat{v}(t_{i+1})^T [\xi(t_{i+1}, \Delta\gamma) - A_i \xi(t_i, \Delta\gamma)] \quad (\text{A.2})$$

where $\hat{J} \equiv \bar{J}$. The matrix A_i is defined here as:

$$A_i = \begin{pmatrix} \tilde{M}_i & \tilde{M}_i^\gamma \\ 0 & I \end{pmatrix} \quad (\text{A.3})$$

The incremental changes in \hat{J} , $\xi(t_i, \Delta\gamma)$ and \hat{v} due to incremental change in one of the components of $\Delta\gamma$ gives:

$$\begin{aligned} \Delta\hat{J} = & \sum_{i=0}^{m-1} \hat{v}(t_{i+1})^T [\Delta\xi(t_{i+1}, \Delta\gamma) - A_i \Delta\xi(t_i, \Delta\gamma)] + \\ & \sum_{i=0}^{m-1} \Delta\hat{v}(t_{i+1})^T [\xi(t_{i+1}, \Delta\gamma) - A_i \xi(t_i, \Delta\gamma)] - \\ & \sum_{i=0}^{m-1} \hat{v}(t_{i+1})^T \frac{\partial \xi(t_{i+1}, \Delta\gamma)}{\partial \Delta\gamma} \Delta\gamma + \left[\frac{\partial \hat{J}}{\partial \xi(t_i, \Delta\gamma)} \right]^T \Delta\xi(t_i, \Delta\gamma) \end{aligned} \quad (\text{A.4})$$

The above expression after simple calculations yields:

$$\begin{aligned} \Delta \hat{J} = & \sum_{i=1}^{m-1} \Delta \xi(t_i, \Delta \gamma) [\hat{v}(t_i)^T - A_i \hat{v}(t_{i+1})^T] + \left[\frac{\partial J}{\partial \xi(t_i, \Delta \gamma)} \right]^T \Delta \xi(t_i, \Delta \gamma) + \\ & \hat{v}(t_m)^T \Delta \xi(t_m, \Delta \gamma) - \sum_{i=0}^{m-1} \hat{v}(t_{i+1})^T \frac{\partial \xi(t_{i+1})}{\partial \Delta \gamma} \Delta \gamma \end{aligned} \quad (\text{A.5})$$

An expression for the reduced adjoint model $\hat{v}(t_{i+1})$; $i \in \{m-1, \dots, 1\}$ solved backward in time is followed from above expression:

$$\hat{v}(t_i) = A_i^T \hat{v}(t_{i+1}) + B(t_i) \quad (\text{A.6})$$

with $\hat{v}(t_m)$ equals zero. $B(t_i)$ is given by:

$$B(t_i) = \begin{pmatrix} 2\hat{H}^T R_i^{-1} [\{\mathbf{Y}(t_i) - H(\mathbf{X}^b(t_i))\} - \hat{H}\xi(t_i, \Delta \gamma)] \\ 0 \end{pmatrix}$$

Once the reduced adjoint states $\hat{v}(t_i)$ are known, the gradient $\frac{\partial \hat{J}}{\partial \Delta \gamma}$ is found as:

$$\frac{\partial \hat{J}}{\partial (\Delta \gamma)} = \sum_i -[\hat{v}(t_{i+1})]^T \frac{\partial \xi(t_{i+1})}{\partial (\Delta \gamma)} \quad (\text{A.7})$$

Bibliography

- [1] *WAQUA User's Guide, The Automatic calibration of the WAQUA model*, Rijkswaterstaat, National Institute for Coastal and Marine Management/RIKZ The Netherlands, 1994.
- [2] *WAQAD User's Guide*, Ministry of Transport, Public Works and Water Management, The Netherlands, 2003.
- [3] G. ALFONSI, C. RESTANOB, AND L. PRIMAVERAL, *Coherent structures of the flow around a surface-mounted cubic obstacle in turbulent channel flow*, J.of Wind Engineering and Industrial Aerodynamics, 91 (2003), pp. 495–511.
- [4] M. U. ALTAF AND A. W. HEEMINK, *Using POD within variational data assimilation*, in G. Paltineanu and E. Popescu and L. Toma (eds.) Trends and Challenges in Applied Mathematics, Bucharest, Romania, 2007.
- [5] M. U. ALTAF, A. W. HEEMINK, AND M. VERLAAN, *Inverse shallow-water flow modelling using model reduction*, International Journal for Multiscale Computational Engineering, 7 (2009), pp. 577–596.
- [6] ———, *Efficient adjoint free estimation of uncertain parameters in a large scale tidal model of the european continental shelf by POD*, in Tenth Symposium on Stochastic Hydraulics and fifth International Conference on Water Resources and Environment Research, Quebec city, Canada, 2010.
- [7] ———, *Identification of uncertain parameters in a tidal model of north sea by spsa*, in J. C. Pereira and A. Sequeire (eds.) international Conference on Numerical Analysis and Applied Mathematics (ICNAAM 2010), Rhodes, Greece, 2010.
- [8] ———, *Model-reduced variational data assimilation for shallow water flow modeling*, in J. C. Pereira and A. Sequeire (eds.) 5th European Conference on Computational Fluid Dynamics, Lisbon, Portugal, 2010.
- [9] M. U. ALTAF, M. VERLAAN, AND A. W. HEEMINK, *Efficient identification of uncertain parameters in a large scale tidal model of european continental shelf*

- by proper orthogonal decomposition*, accepted for publication in *Int. J. Numer. Meth. Fluids*, (2010).
- [10] ———, *Simultaneous perturbation stochastic approximation for tidal models*, *Ocean Dynamics*, submitted, (2010).
- [11] A. C. ANTOULAS, *Approximation of large-scale Dynamical Systems*, USA: SIAM, 2005.
- [12] A. F. BENNET AND P. C. MCINTOSH, *Open ocean modeling as an inverse problem: tidal theory*, *J. Phys. Oceanogr.*, 12 (1982), pp. 1004–1018.
- [13] G. BERKOOZ, P. HOLMES, AND J. LUMLEY, *The proper orthogonal decomposition in the analysis of turbulent flows*, *Ann. Rev. Fluid Mech*, 25 (1993), pp. 777–786.
- [14] Y. CAO, J. ZHU, Z. LUO, AND I. M. NAVON, *Reduced order modeling of the upper tropical Pacific ocean model using proper orthogonal decomposition*, *Comp. and Math. with applications*, 52 (2006), pp. 1373–1386.
- [15] Y. CAO, J. ZHU, I. M. NAVON, AND Z. LUO, *A reduced-order approach to four-dimensional variational data assimilation using proper orthogonal decomposition*, *Int. J. Numer. Meth. Fluids*, 53 (2007), pp. 1571–1583.
- [16] J. CARRERA AND S. P. NEUMAN, *Estimation of aquifer parameters under transient and steady state conditions, part 1: Maximum likelihood method incorporating prior information*, *Wat. Resour. Res.*, 22 (1986), pp. 199–210.
- [17] D. C. CHIN, *Comparative study of stochastic algorithms for system optimization based on gradient approximation*, *IEEE Transactions on Systems, Man, and Cybernetics*, 27 (1997), pp. 244–249.
- [18] R. COURANT AND D. HILBERT, *Methods of Mathematical Physics*, New York: Wiley Interscience, 1953.
- [19] P. COURTIER, *Le modele adjoint, outil pour des experiences de sensibility*, Note de travail 166. EERM, Paris France, 36 (1986).
- [20] P. COURTIER AND O. TALAGRAND, *Variational assimilation of meteorological observations with direct and adjoint shallow water equations*, *Tellus*, 42 (1990).
- [21] P. COURTIER, J. N. THEPAUT, AND A. HOLLINGSWORTH, *A strategy for operational implementation of 4d-var, using an incremental approach*, *Quarterly Journal of the Royal Meteorological Society*, 120 (1994), pp. 1367–1387.
- [22] D. N. DAESCU AND I. M. NAVON, *A dual weighted approach to order reduction in 4dvar data assimilation*, *Mon. Wea. Rev.*, 136 (2008), pp. 1026–1041.
- [23] S. K. DAS AND R. W. LARDNER, *On estimation of parameters of hydraulic models by assimilation of periodical tidal data*, *J. Geophys Res.*, 96 (1991), pp. 15,187–15196.

- [24] ———, *Variational parameter estimation for a two dimensional numerical tidal model*, International Journal for Numerical Methods in Fluid, 15 (1992), pp. 313–327.
- [25] DE VRIES, *The implementation of the WAQUA/CSM16 model for real time storm surge forecasting.*, Technical reports TR-131, KNMI, de Bilt, The Netherlands, 1991.
- [26] F. DELAY, A. BUORO, AND G. DE MARSILY, *Empirical orthogonal functions analysis applied to the inverse problem in hydrogeology: Evaluation of uncertainty and simulation of new solutions*, Math. Geology, 33 (2001), pp. 927–949.
- [27] R. DELAY, *Atmospheric data analysis*, Cambridge University Press, 1991.
- [28] DELTACOMMISSIE, *Eindverslag en interimadviezen rapport Deltacommissie*, Staatsdrukkerij: Den Haag, 1961.
- [29] H. ELBERN, H. SCHMIDT, AND A. EBEL, *Variational data assimilation for tropospheric chemistry modeling*, J. Geophys. Res., 102 (1997), pp. 15967–15985.
- [30] G. EVENSEN, *Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics*, J. Geophys. Res., 99 (1994), pp. 10,143–10,162.
- [31] F. FANG, C. C. PAIN, I. M. NAVON, D. PIGGOTT, G. J. GORMAN, P. A. ALLISON, AND A. J. H. GODDARD, *A pod reduced-order unstructured mesh ocean modelling method for moderate reynolds number flows*, Ocean Modeling, 28 (2009), pp. 127–136.
- [32] F. FANG, C. C. PAIN, I. M. NAVON, D. PIGGOTT, G. J. GORMAN, P. E. FARRELL, P. A. ALLISON, AND A. J. H. GODDARD, *A pod reduced-order 4d-var adaptive mesh ocean modelling approach*, Int. J. Numer. Meth. Fluids, (2008).
- [33] R. FLETCHER, *Prctical methods for optimization*, Padstow: A Wiley-Interscience Pulication, 1991.
- [34] G. GAO AND A. C. REYNOLDS, *A stochastic algorithm for automatic history matching*, SPEJ, 12 (2007), pp. 196–208.
- [35] L. GERENCSEK, S. D. HILL, AND Z. VAGOO, *Discrete optimization via spsa*, in Proc. of American Control Conference, USA, 2001.
- [36] H. GERRITSEN, *What happened in 1953? the big flood in the netherlands in retrospect*, Phil. Trans. R. Soc. A, 363 (2005), pp. 1271–1291.
- [37] H. GERRITSEN, J. DE FRIES, AND M. PHILIPPART, *The dutch continental shelf model, in: Quantitative skill assement for coastal ocean models*, Coastal and stuarine studies (D. Lynch and A. Davies eds.) American Geophysical Union, 47 (1995), pp. 425–467.
- [38] R. GIERING AND T. KAMINSKI, *Recipes for adjoint code construction*, ACM Transactions on Mathematical Software, 24 (1998), pp. 437–474.

- [39] K. GLOVER, *Int. J. of Control*, 39 (1984), p. 1115.
- [40] M. D. GUNZBURGER, *Reduced-order modeling, data compression and the design of experiments*, in Second DOE workshop on multiscale Mathematics, Broomfield, Colorado, July 20–22 2004.
- [41] S. P. HAN, *A globally convergent method for nonlinear programming*, *Journal of Optimization, Theory and Application*, 22 (1977).
- [42] A. W. HEEMINK AND H. KLOOSTERHUIS, *Data assimilation for non-linear tidal models*, *International Journal for Numerical Methods in Fluids*, 11 (1990), pp. 1097–1112.
- [43] A. W. HEEMINK, E. E. A. MOUTHAAAN, AND M. R. T. ROEST, *Inverse 3D shallow water flow modeling of the continental shelf*, *Continental Shelf Research*, 22 (2002), pp. 465–484.
- [44] I. HOTEIT AND D. T. PHAM, *Evolution of the reduced state space and data assimilation schemes based on the kalman filter*, *J. Meteor. Soc. Japan*, 81 (2006), pp. 21–39.
- [45] D. W. HUTCHISON AND S. D. HILL, *Simulation optimization of airline delay with constraints*, in *Proc. 36th IEEE Conference on Decision and Control*, 1997.
- [46] M. ILAK AND C. W. ROWLEY, *Reduced-order modeling of channel flow using traveling pod and balanced pod*, in *3rd AIAA Flow Control Conference*, San Francisco, USA, June 5–8 2006.
- [47] T. KAMINSKI, R. GIERING, AND M. SCHOLZE, *An example of an automatic differentiation-based modeling system*, *Lecture Notes Comput. Sci.*, 2668 (2003), pp. 5–104.
- [48] K. KARHUNEN, *Zur spektral theorie stochastischer prozsee*, *Ann. Acad. Sci. Fennicae*, 34 (1946).
- [49] G. M. KEPLER, H. T. TRAN, AND H. T. BANKS, *Reduced-order Compensator control of species transport in cvd reactor*, *Optimal Control Appl. and meth.*, 21 (2000), pp. 143–160.
- [50] J. KIEFER AND J. WOLFOWITZ, *Stochastic estimation of a regression function*, *Ann. Math. Statist.*, 23 (1952), pp. 462–466.
- [51] K. KUNISH AND S. VOLKWEIN, *Galerkin proper orthogonal decomposition methods for general equations in fluid dynamics*, *Siam J. Num. Analysis*, 40 (2002), pp. 492–515.
- [52] S. LALL, J. E. MARSDEN, AND S. GLAVA, *A subspace approach to balanced truncation for model reduction of nonlinear control systems*, *Int. J. Robust Nonlin. Contr.*, (2002), pp. 519–535.

- [53] R. W. LARDNER, A. H. AL-RABEH, AND N. GUNAY, *Optimal estimation of parameters for a two dimensional hydrodynamical model of the arabian gulf*, J. Geophys. Res. Oceans, 98 (1993), pp. 229–242.
- [54] A. S. LAWLESS, N. C. NICHOLS, C. BOESS, AND A. BUNSE-GERSTNER, *Using model reduction methods within incremental 4dvar*, Mon. Wea. Rev., 136 (2008), pp. 1511–1522.
- [55] J. LEENDERTSE, *Aspects of a computational model for long-period water wave propagation*, PhD thesis, Rand Corporation, Memorandum RM-5294-PR, Santa Monica, 1967.
- [56] M. LOEVE, *Functions aleatoire de second ordre*, Revue Science, 84 (1946), pp. 195–206.
- [57] C. LOPEZ AND E. GARCIA-HERANDEZ, *Low-dimensional dynamical system model for observed coherent structures in ocean setellite data*, Physica A, 328 (2003), pp. 233–250.
- [58] X. Q. LU AND J. C. ZHANG, *Numerical study on spetially varying bottom friction coefficient of a 2-d tidal model with adjoint method*, Continental Shelf Research, 26 (2006), pp. 1905–1923.
- [59] J. L. LUMLEY, *The structure of inhomogeneous turbulence*, in A. M. Yaglom and V. I. Tatarski (eds), Nauka, Moscow, 1967, pp. 166–178.
- [60] H. V. LY AND H. T. TRAN, *Proper orthogonal decomposition for flow calculations and optimal control in a horizontal cvd reactor*, Quarterly J. of Applied Mathematics, 60 (2002), pp. 631–656.
- [61] A. W. G. E. A. M. E. PHILIPPART, *DATUM2-data assimilation with altimetry techniques used in a tidal model, 2nd program*, Technical Report NRSP-2 98-19, BCRS, Delft, October 1998, 1998.
- [62] B. C. MOORE, *Principal component analysis in linear systems: controllability, observability and model reduction*, IEEE Trans. Automat. Contr., 26 (1981), pp. 17–32.
- [63] E. MOUTHAN, A. W. HEEMINK, AND K. ROBACZEWSKA, *Assimilation of ers-1 altimeter data in a tidal model of the continental shelf*, Deutsche Hydrographische Zeitschrift, 36 (1994), pp. 285–319.
- [64] I. M. NAVON, *Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography*, Dyn. Atmos. and Oceans (Special issue in honor of Richard Pfeffer), 27 (1997), pp. 55–79.
- [65] J. NOCEDAL, *Updating quasi-newton matrices with limited storage*, Mathematics of Computation, 35 (1990), pp. 773–782.
- [66] D. S. OLIVER, A. C. REYNOLDS, AND N. LIU, *Inverse theory for petroleum reservoir characterization and history matching*, UK: Cambridge, 2008.

- [67] H. M. PARK AND C. H. CHO, *low dimensional modeling of flow reactors*, Int. J. Heat and Mass Transfer, 39 (1996), pp. 3311–3323.
- [68] K. PEARSON, *On lines and planes of closest fit to points in space*, Phil. Mag., 2 (1901), pp. 559–572.
- [69] R. D. PRABHU, C. S. SCOTT, AND Y. CHANGLY, *The influence of control on proper orthogonal decomposition of wall-bounded turbulent flows*, Phys. Fluids, 13 (2001), pp. 229–242.
- [70] C. L. PROVOST, M.-L. GENCO, AND F. LYARD, *Modelling and predicting tides over the world ocean*, in: *Quantitative skill assement for coastal ocean models*, Coastal and stuarine studies (D. Lynch and A. Davies eds.) American Geophysical Union, 47 (1995).
- [71] M. RATHINAM AND L. PETZOLD, *A new look at proper orthogonal decomposition*, SIAM Journal on Numerical Analysis, 41 (2003), pp. 1893–1925.
- [72] C. ROBERT, S. DURBIANO, E. BLAYO, J. VERRON, J. BLUM, AND F. X. LEDIMET, *A reduced order strategy for 4d-var data assimilation*, J. Mar. Syst., 57 (2005), pp. 70–82.
- [73] C. ROWLEY, *Model reduction for fluids, using balanced proper orthogonal decomposition*, International Journal of Bifurcation and Chaos, 15 (2005), pp. 997–1013.
- [74] C. ROWLEY, T. COLONIUS, AND R. M. MURRAY, *Model reduction for compressible flow using pod and galerkin projection*, Physica D, 189 (2004), pp. 115–129.
- [75] W. H. A. SCHILDERS, H. A. VAN DER VORST, AND J. ROMMES, *Model order reduction: theory, research aspects and application*, Springer: Berlin, 2008.
- [76] A. SEGERS, A. W. HEEMINK, M. VERLAAN, AND M. VAN LOON, *A modified rrsqrt-filter for assimilating data in atmospheric chemistry models*, Environmental Modeling and Software, 15 (2000), pp. 663–671.
- [77] J. H. SEINFELD AND C. KRAVARIS, *Distributed parameter identification in geophysics-petroleum reservoirs and aquifers*, in S. G. Tzafestas (ed.) Distributed parameter control systems, Pergamon press, Oxford, 1982, pp. 367–390.
- [78] L. SIROVICH, *chaotic dynamics of coherent structures*, Physica D, 37 (1987), pp. 126–145.
- [79] T. R. SMITH, *Low-dimensional models of plane Couette flow using the proper orthogonal decomposition*, PhD thesis, Princeton University, USA, 2003.
- [80] J. C. SPALL, *Implementation of the simultaneous perturbation algorithm for stochastic optimization*, IEEE Transactions on Aerospace and Electronic Systems, 34 (1998), pp. 817–823.

- [81] ———, *Adaptive stochastic approximation by the simultaneous perturbation method*, IEEE Transactions on Automatic Control, 45 (2000), pp. 1839–1853.
- [82] G. S. STELLING, *On the construction of computational methods for shallow water flow problem*, PhD thesis, Rijkswaterstaat Communications 35, Rijkswaterstaat, 1984.
- [83] O. TALAGRAND, *Assimilation of observations, an introduction*, Journal of the Meteorological Society of Japan, 75 (1997), pp. 191–209.
- [84] C. B. M. TE STROET, *Calibration of stochastic groundwater flow models*, PhD thesis, Technical University Delft, The Netherlands, 1995.
- [85] P. G. J. TEN-BRUMMELHUIS, *Parameter estimation in tidal flow models with uncertain boundary conditions*, PhD thesis, Twente University, the Netherlands, 1992.
- [86] P. G. J. TEN-BRUMMELHUIS AND A. W. HEEMINK, *Parameter identification in tidal models with uncertain boundary conditions*, Stochastic Hydrology and Hydraulics, 4 (1990), pp. 193–208.
- [87] P. G. J. TEN-BRUMMELHUIS, A. W. HEEMINK, AND H. F. P. VAN DEN BOOGARD, *Identification of shallow sea models*, Int. J. for Num. Met. Fluids, 17 (1993), pp. 637–665.
- [88] F. X. LE DIMET AND O. TALAGRAND, *Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects*, Tellus, 38 (1986), pp. 97–110.
- [89] W. C. THACKER AND R. B. LONG, *Fitting models to inadequate data by enforcing spatial and temporal smoothness*, J. Geophys. Res., 93 (1988), pp. 10655–10664.
- [90] X. TIAN, Z. XIE, AND A. DAI, *An ensemble-based explicit four-dimensional variational assimilation method*, J. Geophys. Res., 113 (2008).
- [91] E. TZIPERMAN, W. C. THACKER, AND R. B. LONG, *Oceanic data analysis using a general circulation model, part2: A north atlantic model*, J. Phys. Oceanography, 22 (1992), pp. 1458–1485.
- [92] D. S. ULMAN AND R. E. WILSON, *Model parameter estimation for data assimilation modeling: Temporal and spatial variability of the bottom drag coefficient*, J. Geophys. Res. Oceans, 103 (1998), pp. 5531–5549.
- [93] G. K. VERBOOM, J. G. DE RONDE, AND R. P. VAN DIJK, *Een model van het Europese Continentale Plat voor windopzet en waterkwaliteitsberekeningen*, Rijkswaterstaat, Tidal Waters Division WAO-87.021/Delft Hydraulics Z96.00 (in Dutch), 1987.
- [94] ———, *A fine grid tidal flow and storm surge model of the north sea*, Continental Shelf Res, 12 (1992), pp. 213–233.

- [95] M. VERLAAN, *Efficient Kalman filtering algorithms for hydrodynamic models*, PhD thesis, Technical University Delft, The Netherlands, 1998.
- [96] M. VERLAAN AND A. W. HEEMINK, *Data assimilation schemes for non-linear shallow water flow models*, *Adv. Fluid Mechanics*, 96 (1996), pp. 277–286.
- [97] M. VERLAAN, E. MOUTHAAAN, E. KUIJPER, AND M. PHILIPPART, *Parameter estimation tools for shallow water flow models*, *Hydroinformatis*, 96 (1996), pp. 341–348.
- [98] M. VERLAAN, A. ZIJDERVELD, H. VRIES, AND J. KROOS, *Operational storm surge forecasting in the Netherlands: developments in last decade*, *Phil. Trans. R. Soc. A*, 363 (2005), pp. 1441–1453.
- [99] P. T. M. VERMEULEN AND A. W. HEEMINK, *Model-reduced variational data assimilation*, *Mon. Wea. Rev.*, 134 (2006), pp. 2888–2899.
- [100] P. T. M. VERMEULEN, A. W. HEEMINK, AND C. B. M. TE STROET, *low-dimensional modeling of numerical groundwater flow*, *Hydrological Processes*, 18 (2004), pp. 1487–1504.
- [101] ———, *Model inversion of transient nonlinear groundwater flow models using model reduction*, *Water Resources Research*, 42 (2006).
- [102] P. T. M. VERMEULEN, A. W. HEEMINK, AND J. R. VALSTAR, *Inverse modeling of groundwater flow using model reduction*, *Wat. Resour. Res.*, 41 (2005).
- [103] C. WANG, L. GAOMING, AND A. C. REYNOLDS, *Production optimization in closed-loop reservoir management*, *SPE Journal*, 14 (2009), pp. 506–523.
- [104] K. WILCOX AND J. PERAIRE, *Balanced model reduction via the proper orthogonal decomposition*, *AIAA Journal*, 40 (2002).

Summary

Identifying uncertain parameters in large-scale numerical flow models can be done using the variational method. However, for implementing the variational method the adjoint model have to be available, which requires highly complex computer code and maintenance and thus hampers its applications. To ease this problem, this thesis has explored several methods for efficiently identifying uncertain parameters in a large-scale tidal model of the entire European continental shelf which does not require the implementation of these complex adjoint code.

In this study, as a first step an estimation method based on model reduction is developed and investigated for the estimation of diffusion coefficient in a simple 2D-advection diffusion model. Two projection based model reduction methods were considered, namely proper orthogonal decomposition (POD) and Balanced proper orthogonal decomposition (BPOD). In the POD based estimation method an ensemble of forward model simulations is used to determine an approximation of the covariance matrix of the model variability and a small number of the leading eigenvectors of this matrix is used to define a model subspace. By projecting the original model onto this subspace an approximate linear reduced model is obtained. Once the reduced model is available its adjoint can be implemented easily and the minimization problem is solved completely in reduced space with very low computational cost. BPOD is also a model reduction method which considers both inputs and outputs of the system while determining the reduce subspace. The estimation method has been extended by including BPOD procedure into the estimation procedure. Numerical results from a simple pollution model demonstrate that the POD based estimation approach successfully estimate the diffusion coefficient for both advection dominated problems as for diffusion dominated problems. Another important message in this study, although lots of effort had been made in constructing a reduced order model by the BPOD method, the minimization results demonstrated that both the POD and the BPOD methods performed similarly.

Preliminary results showed the validity of the POD based model reduction methods for parameter estimation. As a next step, the POD based estimation method is used to calibrate numerical tidal models. Results from (twin) numerical experiments showed that the POD based calibration method performed very efficiently to estimate depth values in the selected regions of the model domain. The computational costs of

the POD based calibration method are dominated by the generation of an ensemble of forward model simulations where the simulation period of the ensemble is equivalent to the timescale of the original model. It has also been found in the study that it is not needed to use a full simulations of the original model for the generation of the ensemble.

The POD based calibration method has also been implemented for the estimation of the water depth and space varying bottom friction coefficient values in a very large-scale DCSM model. The recently designed large-scale spherical grid based water level model for the northwest European continental shelf (around 1000000 computational grid points) has been used for this purpose. This has been the first application of the POD based calibration method to a very large-scale model and with real data. Results from numerical experiments showed that the calibration method performs very efficiently. An overall improvement of more than 50% was observed after the calibration in comparison with the initial model. The results also demonstrated that the POD based calibration method offered a very efficient minimization technique compared to the classical adjoint method without the burden of implementation of the adjoint.

As a concluding step, to estimate depth values in the model DCSM, a Simultaneous perturbation stochastic approximation (SPSA) method has been used. The method uses stochastic simultaneous perturbation of all model parameters to generate a search at each iteration. SPSA is based on a highly efficient and easily implemented simultaneous perturbation approximation to the gradient. This gradient approximation for the central difference method uses only two objective function evaluations independent of the number of parameters being optimized. The results from experiments showed that SPSA has a lower convergence rate than POD based calibration method, however the computational cost in each iteration of the SPSA method is usually far less than the POD based calibration method. The results also demonstrated that the SPSA algorithm proved to be a promising optimization algorithm for model calibration for cases where adjoint code is not available for computing the gradient of the objective function.

Samenvatting

Het schatten van onbekende parameters in grootschalige numerieke stromingsmodellen kan gedaan worden met behulp van de variationele methode. Om deze methode te kunnen implementeren moet men de beschikking hebben over het geadjungeerde model. Het daarvoor noodzakelijke programmeerwerk is zeer complex, hetgeen het toepassen van de variationele methode onaantrekkelijk maakt. In dit proefschrift worden enkele methoden onderzocht voor het efficiënt schatten van onbekende parameters zonder gebruik te maken van het geadjungeerde model. Deze methoden worden toegepast op grootschalige getijdige modellen van het Europese continentale plat.

De eerste stap in dit onderzoek bestaat uit het bestuderen en ontwikkelen van een schattingsmethode voor de diffusiecoëfficiënt in een eenvoudig 2D advection-diffusie model, gebaseerd op model reductie. Hierbij zijn twee projectie-gebaseerde methoden beschouwd, te weten 'proper orthogonal decomposition', POD en 'balanced proper orthogonal decomposition', BPOD. Het idee achter deze schattingsmethode is als volgt: Eerst wordt een ensemble voorwaartse simulaties van het model gebruikt om een benadering te krijgen van de covariantie matrix van de variabiliteit van het model. Vervolgens worden enkele dominanteeigenvectoren van deze matrix gebruikt om een deelruimte van het model te bepalen. Door het originele model op deze deelruimte te projecteren, verkrijgt men een lineaire benadering van het model (het 'gereduceerde model'). De geadjungeerde van het gereduceerde model kan eenvoudig worden bepaald, en ook de complexiteit van het minimaliseringsprobleem in de gereduceerde ruimte is laag. BPOD is ook een model reductie methode die zowel de input als de output van het system betreft bij het bepalen van het gereduceerde model. De hierboven genoemde schattingsmethode is uitgebreid door de BPOD procedure toe te voegen in de schattingsprocedure. Numerieke resultaten van een eenvoudig concentratie model tonen aan dat POD erin slaagt de diffusiecoëfficiënt goed te schatten, zowel voor advection-gedomineerde problemen als voor diffusie-gedomineerde problemen. De numerieke resultaten tonen ook aan dat de twee methoden POD en BPOD vergelijkbaar presteren, terwijl de implementatie van BPOD complexer is.

De volgende stap is om de POD model reductie methode toe te passen op de kalibratie van numerieke getijdmodellen. Een 'tweeling' experimenten toont dat de op POD gebaseerde kalibratie methode zeer efficiënte schattingen geeft van de waterdiepte in vooraf bepaalde delen van het model domein. De rekenkundige complexiteit

van de op POD gebaseerde kalibratie methode wordt gedomineerd door de generatie van het ensemble voorwaartse modellen, waarbij de simulatie periode voor het ensemble equivalent is aan de tijdschaal van het oorspronkelijke model. Uit ons onderzoek blijkt dat het genereren van het ensemble mogelijk is zonder het originele model ver de hele calibratie periode te simuleren.

Het onderzoek heeft zich vervolgens gericht op het toepassen van de op POD gebaseerde kalibratie methode voor het schatten van de waterdiepte en de plaatsafhankelijke bodemwrijvingscoëfficiënt in een zeer grootschalig DCSM model. Hiervoor is het recent ontwikkeld model van het noordwesten van het Europese continentale plat op basis van een grootschalige bolvormig grid gebruikt (met ongeveer 1000000 rekenkundige gridpunten). Dit is de eerste toepassing van de op POD gebaseerde kalibratie methode op een grootschalig model met ‘echte’ data. Numerieke experimenten tonen dat de kalibratie methode zeer efficiënt functioneert. Na kalibratie werd een algehele verbetering van meer dan 50% waargenomen in vergelijking met het oorspronkelijke model. Deze resultaten tonen ook dat de op POD gebaseerde kalibratie methode een zeer efficiënt alternatief vormt voor de klassieke methode gebaseerd op het gebruik van geadjungeerde model.

Als laatste stap in dit onderzoek is een stochastische approximatie op basis van simultane perturbatie (‘simultaneous perturbation stochastic approximation’, SPSA) uitgevoerd. Deze methode gebruikt iteratief stochastische simultane perturbaties van alle model parameters om een zoekrichting te bepalen. SPSA is gebaseerd op hoogstefficiënte en eenvoudig te implementeren simultane stochastische perturbatie benaderingen van de gradiënt. Deze approximatie van de gradiënt in de centrale differentie methode maakt gebruik van slechts twee doelfunctie evaluaties, ongeacht het aantal parameters dat geschat dient te worden. De experimentele resultaten tonen dat SPSA een lagere convergentiesnelheid heeft dan de op POD gebaseerde kalibratie methode, echter, de berekeningen per iteratie zijn in het algemeen veel lager dan voor de op POD gebaseerde kalibratie methode. De resultaten tonen ook aan dat het SPSA algoritme veelbelovend is voor model kalibratie waarbij het geadjungeerde model niet beschikbaar is om de gradiënt van de doelfunctie te bepalen.

Acknowledgements

This thesis is a result of contributions from many people. I would like to extend my thanks to all of them and some of them in particular.

First of all, I express my sincere gratitude to my promoter and supervisor Prof. Arnold Heemink for his support, help and guidance. It was because of him, I was able to start PhD at Delft university. He is always open to ideas and encouraged me throughout my research. I really admire his patience and supervision. He is a true researcher and a man with extraordinary qualities.

I am very grateful to Martin Verlaan for his advices and very stimulating discussions and new ideas. His valuable suggestions and constant enthusiasm kept me highly motivated. It was a great pleasure to work with him. I am also grateful for his help in providing me necessary research and administrative resources at Deltares.

I would like to thank Dr. W. Hussain and A. Iqbal for their efforts and support. I thank Firmijn Zijl for providing me the Matlab analysis routines which were very useful during research. I am thankful to Herman Gerritsen for our discussions. He always acknowledged my work which kept me motivated and focussed. I thank Ghada al Serafy for healthy discussions which made my stay at Deltares enjoyable.

I would like to thank Maria for being a critical reviewer of not only the initial draft of this thesis but also for most of my papers that helped me improve. I am grateful to Sonja Cox for providing Dutch translation of the thesis summary.

There are several people, who made my study in TU Delft possible. I am very grateful to W. van Horssen for organizing and arranging many administrative matters. I would like to thank Evelyne Sharabi and Dorothee Engering for organizing and helping with various matters. I thank CICAT project assistant Franca Posts for her kindness.

I spent four very interesting years at the Mathematical Physics group and I wish to thank people who work there. I am very grateful to Kees Lemmens for his help and support in programming and software and for his patience in answering all my questions and concerns. I am grateful to Pieter Wilders who helped me with transport modeling. I would like to thank my mates Alina Barbu, Alex and Julius Sumihar for the cooperation, nice talks and dinners. I am grateful to Alex in helping me arranging the PhD defense ceremony also. I thank Julius Sumihar for being my main guide in understanding the details of the WAQUA software. I thank Nils for helping me under-

stand COSTA toolbox. This list of co-workers would't complete without mentioning the names of Remus, Wiktoria, Gosia, Sajjad, Ivan, Murtaza, Haider, Nils, Eka and Atif. I have enjoyed a wonderful time with all of them during coffee breaks, lunches.

I would like to thank Amir, Shahzad, Jawad, Aqeel and their families for their wonderful company. I will always remember the fun during the cricket games in Delft on weekends for which I am thankful to the fellow players both from Pakistan and India.

There are no words to express my gratitude and love for my parents. I am grateful to my parents for their continuous support. I thank my brother and sisters, Naveed, Sadaf's family and my friends (in particular Imran, Hadi, Ifti, Kamal, Shahzad and Amir) for all their encouragement. Finally, my special thanks goes to my wife Sadaf. You always have had a lot of love and patience. You gave me so much strength for finalising this thesis. Thank you my daughter, Maham, you gave me the great joy of becoming father. You made me understand that love and patience are two important elements in finding the way out in the daily life.

Curriculum Vitae

M. Umer Altaf was born on 31 July, 1974 in Rawalpindi, Pakistan. He finished his undergraduate degree in Mathematics in 1994 at Punjab University, Pakistan. After this, M. Umer Altaf started his studies in Applied Mathematics at Quaid-e-Azam University Pakistan. He received the title of Master of Science in Mathematics from this university in 1997. After graduation he worked as a staff member at different institutions (IMIT, Pakistan, IICS, Pakistan and CIMS, Pakistan) from 1998 to 2006. He was involved in teaching and research during this period. He received a grant from Higher education commission of Pakistan to pursue degree in Computer Science which completed in 2003.

Since 2006, M. Umer Altaf has been working on his PhD project at the Delft Institute of Applied Mathematics at Delft University of Technology, The Netherlands. He worked on the project connected with inverse modeling of the tidal flows using model reduction and has been supervised by Prof. dr. ir. A. W. Heemink. The results he obtained during next four years form the content of the current thesis entitled "Model reduced variational data assimilation for shallow water flow models". He worked as research fellow at Deltares, Netherlands during 2009 - 2010.