

Master of Science Thesis

Exploration of Ensemble Kalman Filter for parameter estimation with plasma actuators

**Twin experiment and application using experimentally obtained
quasi-steady flow fields**

C.W.Schoemakers

March 18, 2014



Technische Universiteit Eindhoven

Exploration of Ensemble Kalman Filter for parameter estimation with plasma actuators

**Twin experiment and application using experimentally obtained
quasi-steady flow fields**

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

C.W.Schoemakers

March 18, 2014



Delft University of Technology

Copyright © Aerospace Engineering, Delft University of Technology
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled "**\Exploration of Ensemble Kalman Filter for parameter estimation with plasma actuators**" by **C.W.Schoemakers** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: March 18, 2014

Supervisors:

Dr. M.Kotsonis

Dr.ir. B.W. van Oudheusden

Dr. D.Ragni

Ir. I.Azijli

Summary

This thesis aims to explore and enable parameter estimation using the data assimilation method of the ensemble Kalman filter (EnKF). More specifically applied to the quasi-steady flow field, measured using particle image velocimetry (PIV), of a plasma actuator in quiescent flow where the parameters to be estimated are those describing the force field generated. The ensemble Kalman filter combines experimental observations together with a prior, created through numerical simulation, in a stochastic framework, to compute a closer estimation of the true state of the system together with an estimation of the error. This method has already proven to be able to improve the spacial and temporal accuracy experimental observations, as well as providing an estimation of the parameters describing the system. The EnKF has not yet been used in the field of plasma actuators, and can prove a valuable tool in improving upon existing experimental methods, including the calculation of the pressure field, and the time dependent force field. The calculation of the pressure field has mostly been ignored, and the determination of the time dependent force field has been done with limited success so far using only experimental methods only.

The EnKF and a finite difference laminar Navier-Stokes solver have been implemented in Matlab. This solver can compute the flow field given a prescribed force field, which will simulate the working of the plasma actuator. The observations are used in the form of time-averaged particle image velocimetry obtained by [Kotsonis \(2012\)](#) during previous studies, creating a steady state system. A slight modification of the traditional EnKF method in steady state is proposed in order to correct for sub-optimal update steps of the EnKF when combining the experimental observations with the numerical model. The proposed modified version called the semi-steady state EnKF (SSS-EnKF) integrates the updated system forward in time in between updates.

The first part of the research is focused on the understanding of the SSS-EnKF within the plasma actuator framework. A twin problem is set up with artificially generated observations to allow for a controlled environment to work in. The sensitivity of the results is measured with respect to several filter settings, such as ensemble size, prior mean and variance, creation of the prior, filter type and integration time between updates. The results show a high sensitivity to the initial prior mean and variance, which is most likely due to the necessary introduction of bias error to keep all ensemble member stable in the numerical solver. The results show relatively low sensitivity to the other filter settings.

Using the optimal filter settings determined by the twin problem, the SSS-EnKF is used using the actual PIV data for eight different plasma actuator configurations. The results of the filtering is compared to existing estimations of the force field calculated by [Kotsonis \(2012\)](#). Most results show good agreement with the existing results. Some results, however, show poor comparison due to the high sensitivity of the initial prior. Although the current method is not as robust as existing methods, I believe that major improvements can be made if further research is done, allowing not only for a robust method of estimating the quasi-steady force field, but also accurately calculating the time resolved force field, together with the pressure field.

Table of Contents

Summary	v
List of Figures	xi
List of Tables	xix
1 Introduction	1
2 Background; plasma actuators and data assimilation techniques	3
2.1 Plasma actuator	3
2.1.1 Physical background	3
2.1.2 Typical Configuration	4
2.1.3 Mechanical and Electrical Behavior	5
2.1.4 Numerical Simulation	6
2.2 Parameter estimation in data assimilation	7
2.2.1 Estimation of optimal model parameters	8
2.2.2 Variational minimizing schemes	8
3 Methodology	11
3.1 EnKF	11
3.1.1 Kalman Filter	11

3.1.2 Ensemble Kalman Filter	14
3.1.3 EnKF for steady state problem	18
3.1.4 Initial EnKF requirements	18
3.2 PIV measurements as observations	19
3.2.1 Actuator set-up	20
3.2.2 Observational error	21
3.3 Incompressible flow solver as forecast model	26
3.3.1 Conservation of Mass	27
3.3.2 Conservation of momentum	27
3.3.3 Time integration	30
3.3.4 Boundary conditions and grid	30
3.3.5 Measurement operator	32
3.4 Initial set-up	34
3.4.1 Body force parameterization	34
3.4.2 Solver stability	35
4 Twin Problem	37
4.1 Set-up	38
4.1.1 Creation of the artificial observations	38
4.1.2 Creation of the initial prior	38
4.1.3 Univariate statistics of initial priors	39
4.1.4 Bivariate statistics of initial priors	42
4.1.5 Twin problem Iter settings	44
4.2 Twin Problem Results	44
4.2.1 Steady State EnKF	46
4.2.2 Semi Steady State Filtering	48
4.2.3 Optimal settings for PIV	56

5 Semi Steady State filtering with PIV data	59
5.1 Initial Priors for PIV	59
5.2 Parameter estimation	60
5.2.1 Best force field estimates	61
5.2.2 Filter performance	61
5.2.3 Thrust comparison	62
5.2.4 Pressure correction	63
6 Conclusion	67
Bibliography	69
A PIV setup	73
B Twin problem initial prior statistics	75
C SSEnKF Results	91
C.1 Parameter development SSEnKF	92
C.2 Divergence plots SSEnKF	93
D Results Twin problem	95
D.1 Final parameter estimation	96
D.2 Parameter estimation development and divergence plots	99
E PIV filtering with DEnKF results	113
E.1 Parameter estimation results	114
E.2 Force field plots all cases	116
E.3 Parameter estimation development and divergence plots	122

List of Figures

2.1	Schematic of typical DBD configuration and operation (taken from Kotsonis (2012))	4
2.2	Some of the first ever measured induced velocity produced by a DBD actuator (taken from Roth et al. (1998))	5
2.3	Body force determination using <i>gradient method</i> (taken from Kotsonis (2012))	6
3.1	DBD geometric layout	20
3.2	Example of total velocity profiles for the observational domain: Full observational domain (<i>top</i>) and cropped observational domain (<i>bottom</i>).	22
3.3	Divergence of the flow field in full observational domain for actuator case 2 in s^{-1}	22
3.4	Helmholtz decomposition results velocity in x-direction u . From top to bottom: u from PIV, u_{sol} , u_{irrot}	24
3.5	Helmholtz decomposition results velocity in y-direction v . From top to bottom: v from PIV, v_{sol} , v_{irrot}	24
3.6	Picture of complete grid	31
3.7	Comparison numerical solver. Above: Kotsonis (2012) using OpenFoam, below: Matlab code using grid defined by table 3.4	33
4.1	$ \mathbf{E}_{To}^f - \mathbf{Hx}_{To} $ for prior 2b at $To = 0.5s$. <i>top</i> ; x-velocity. <i>bottom</i> ; y-velocity . . .	40
4.2	Univariate statistics of prior 2b x-velocity, $To = 50$; μ (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	41
4.3	Univariate statistics of prior 2b y-velocity, $To = 50$; μ (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	42
4.4	Ensemble members of initial prior 2b, $To = 1.0s$ of a far field flow variable vs a flow variable in the observational domain	43

4.5 Ensemble members of initial prior 2b, $T_o = 1.0$ s of a far field flow variable vs initial parameters	44
4.6 SSEnKF results prior 1a, $T_o = 1.0$ s; $\frac{RMSE_o}{100}[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\Sigma[m=s]$ (* blue line)	47
4.7 Static EnKF results prior 1b; parameter mean and ± 3 error bars. Lines of constant values are equal to true parameter value.	47
4.8 Plot of $RMSE_o$ vs $RMSE_f$ for the EnKF. Different colors represent different Iter version settings; red (v1), blue (v2), cyan (v3), black (v4). The different symbols represent the different initial priors; * (prior 1a/b), + (prior 2a/b), o (prior 3a/b), \square (prior 4), x (prior 5), \diamond (prior 6)	48
4.9 Plot of $RMSE_o$ vs $RMSE_f$ for the DEnKF. Different colors represent different Iter version settings; red (v1), blue (v2), cyan (v3), black (v4). The different symbols represent the different initial priors; * (prior 1a/b), + (prior 2a/b), o (prior 3a/b), \square (prior 4), x (prior 5), \diamond (prior 6)	49
4.10 SSS-EnKF results prior 5v1, $N = 60$, $T_o = 1.0$ s; $RMSE_o \cdot 100[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{10}[m=s]$ (* blue line), Sponge ratio [-] (red stars)	50
4.11 SSS-EnKF results prior 2v4, $N = 80$, $T_o = 0.5$ s; $RMSE_o \cdot 100[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{10}[m=s]$ (* blue line), Sponge ratio [-] (red stars)	51
4.12 SSS-EnKF results prior 1v1, $N = 80$, $T_o = 1.0$ s; $RMSE_o \cdot 100[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{10}[m=s]$ (* blue line), Sponge ratio [-] (red stars)	51
4.13 $RMSE_f$ for different initial integration times for initial priors 1a/b, 2a/b, and 3a/b. Red line denotes the average $RMSE_f$ for a single T_o	52
4.14 $RMSE_f$ for Priors 2a and 3a with different integration times between updates. Red line denoting the average of the runs.	53
4.15 Comparison EnKF with DEnKF through $RMSE_f$	54
4.16 Semi steady State DEnKF results prior 2v1, $N = 60$, $T_o = 0.75$ s; $\frac{RMSE_o}{100}[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\Sigma[m=s]$ (* blue line), Sponge ratio [-] (red *)	57
4.17 Semi steady State DEnKF parameter estimation development for prior 2v1, $N = 60$, $T_o = 0.75$ s; parameter mean and ± 3 error bars. Lines of constant values are equal to true parameter value.	57
5.1 Force field when $RMSE_o$ is lowest. Left from top to bottom, cases 1 through 5, with increasing Voltage. Right from top to bottom case 6,2,7,8, with increasing f_{ac}	62

5.2 Comparison between results obtained by Kotsonis (2012) and the SSS-EnKF method. NOTE: SSS-EnKF results are plotted with errorbars of $100 \cdot 10^{-2}$	63
5.3 $\mathbf{f} - \nabla p$. Left from top to bottom, cases 1 through 5, with increasing Voltage. Right from top to bottom case 6,2,7,8, with increasing f_{ac}	65
5.4 Force fields using <i>gradient method</i> by Kotsonis (2012). Left from top to bottom, cases 1 through 5, with increasing Voltage. Right from top to bottom case 6,2,7,8, with increasing f_{ac}	65
5.5 Comparison between results obtained by Kotsonis (2012) and the SSS-EnKF method. Incorporation the pressure gradient in the thrust calculation	66
B.1 Uni-variant statistics of prior 2a x-velocity, $To = 50$; σ^2 (top right), sk (bottom left), and k (bottom right)	76
B.2 Uni-variant statistics of prior 2a y-velocity, $To = 50$; σ^2 (top right), sk (bottom left), and k (bottom right)	76
B.3 Uni-variant statistics of prior 2a x-velocity, $To = 75$; σ^2 (top right), sk (bottom left), and k (bottom right)	77
B.4 Uni-variant statistics of prior 2a y-velocity, $To = 75$; σ^2 (top right), sk (bottom left), and k (bottom right)	77
B.5 Uni-variant statistics of prior 2a x-velocity, $To = 100$; σ^2 (top right), sk (bottom left), and k (bottom right)	78
B.6 Uni-variant statistics of prior 2a y-velocity, $To = 100$; σ^2 (top right), sk (bottom left), and k (bottom right)	78
B.7 Uni-variant statistics of prior 2b x-velocity, $To = 50$; σ^2 (top right), sk (bottom left), and k (bottom right)	79
B.8 Uni-variant statistics of prior 2b y-velocity, $To = 50$; σ^2 (top right), sk (bottom left), and k (bottom right)	79
B.9 Uni-variant statistics of prior 2b x-velocity, $To = 75$; σ^2 (top right), sk (bottom left), and k (bottom right)	80
B.10 Uni-variant statistics of prior 2b y-velocity, $To = 75$; σ^2 (top right), sk (bottom left), and k (bottom right)	80
B.11 Uni-variant statistics of prior 2b x-velocity, $To = 100$; σ^2 (top right), sk (bottom left), and k (bottom right)	81
B.12 Uni-variant statistics of prior 2b y-velocity, $To = 100$; σ^2 (top right), sk (bottom left), and k (bottom right)	81

B.13 Uni-varianc statistics of prior 3a x-velocity, $To = 50$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	82
B.14 Uni-varianc statistics of prior 3a y-velocity, $To = 50$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	82
B.15 Uni-varianc statistics of prior 3a x-velocity, $To = 75$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	83
B.16 Uni-varianc statistics of prior 3a y-velocity, $To = 75$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	83
B.17 Uni-varianc statistics of prior 3a x-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	84
B.18 Uni-varianc statistics of prior 3a y-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	84
B.19 Uni-varianc statistics of prior 3b x-velocity, $To = 50$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	85
B.20 Uni-varianc statistics of prior 3b y-velocity, $To = 50$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	85
B.21 Uni-varianc statistics of prior 3b x-velocity, $To = 75$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	86
B.22 Uni-varianc statistics of prior 3b y-velocity, $To = 75$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	86
B.23 Uni-varianc statistics of prior 3b x-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	87
B.24 Uni-varianc statistics of prior 3b y-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	87
B.25 Uni-varianc statistics of prior 4 x-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	88
B.26 Uni-varianc statistics of prior 4 y-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	88
B.27 Uni-varianc statistics of prior 5 x-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	89
B.28 Uni-varianc statistics of prior 5 y-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)	89

B.29 Uni-varianc statistics of prior 6 x-velocity, $T_o = 100$; σ^2 (top left), σ^2 (top right), σ_k^2 (bottom left), and k (bottom right)	90
B.30 Uni-varianc statistics of prior 6 y-velocity, $T_o = 100$; σ^2 (top left), σ^2 (top right), σ_k^2 (bottom left), and k (bottom right)	90
C.1 Steady State EnKF; prior 1a, $T_o = 0.5$ s	93
C.2 Steady State EnKF; prior 1a, $T_o = 0.5$ s	93
C.3 Steady State EnKF; prior 1a, $T_o = 1.0$ s	93
C.4 Steady State EnKF; prior 1b, $T_o = 0.5$ s	93
C.5 Steady State EnKF; prior 1b, $T_o = 0.75$ s	93
C.6 Steady State EnKF; prior 1b, $T_o = 1.0$ s	93
C.7 Steady State EnKF; prior 1a, $T_o = 0.5$ s	94
C.8 Steady State EnKF; prior 1a, $T_o = 0.5$ s	94
C.9 Steady State EnKF; prior 1a, $T_o = 1.0$ s	94
C.10 Steady State EnKF; prior 1b, $T_o = 0.5$ s	94
C.11 Steady State EnKF; prior 1b, $T_o = 0.75$ s	94
C.12 Steady State EnKF; prior 1b, $T_o = 1.0$ s	94
D.1 EnKF: prior 1v1, $N = 60$	100
D.2 EnKF: prior 1v1, $N = 80$	100
D.3 EnKF: prior 2v1, $N = 60$	100
D.4 EnKF: prior 2v1, $N = 80$	100
D.5 EnKF: prior 3v1, $N = 60$	101
D.6 EnKF: prior 3v1, $N = 80$	101
D.7 EnKF: prior 4v1, $N = 60$	101
D.8 EnKF: prior 5v1, $N = 60$	101
D.9 EnKF: prior 6v1, $N = 60$	102
D.10 EnKF: prior 2v2, $N = 60$	102

D.11 EnKF: prior 2v2, $N = 80$	102
D.12 EnKF: prior 3v2, $N = 60$	102
D.13 EnKF: prior 3v2, $N = 80$	103
D.14 EnKF: prior 2v3, $N = 60$	103
D.15 EnKF: prior 2v3, $N = 80$	103
D.16 EnKF: prior 3v3, $N = 60$	103
D.17 EnKF: prior 3v3, $N = 80$	104
D.18 EnKF: prior 2v4, $N = 60$	104
D.19 EnKF: prior 2v4, $N = 80$	104
D.20 EnKF: prior 3v4, $N = 60$	104
D.21 EnKF: prior 3v4, $N = 80$	105
D.22 DEnKF: prior 1v1, $N = 60$	106
D.23 DEnKF: prior 1v1, $N = 80$	106
D.24 DEnKF: prior 2v1, $N = 60$	106
D.25 DEnKF: prior 2v1, $N = 80$	106
D.26 DEnKF: prior 3v1, $N = 60$	107
D.27 DEnKF: prior 3v1, $N = 80$	107
D.28 DEnKF: prior 4v1, $N = 60$	107
D.29 DEnKF: prior 5v1, $N = 60$	107
D.30 DEnKF: prior 6v1, $N = 60$	108
D.31 DEnKF: prior 2v2, $N = 60$	108
D.32 DEnKF: prior 2v2, $N = 80$	108
D.33 DEnKF: prior 3v2, $N = 60$	108
D.34 DEnKF: prior 3v2, $N = 80$	109
D.35 DEnKF: prior 2v3, $N = 60$	109
D.36 DEnKF: prior 2v3, $N = 80$	109

D.37 DEnKF: prior 3v3, $N = 60$	109
D.38 DEnKF: prior 3v3, $N = 80$	110
D.39 DEnKF: prior 2v4, $N = 60$	110
D.40 DEnKF: prior 2v4, $N = 80$	110
D.41 DEnKF: prior 3v4, $N = 60$	110
D.42 DEnKF: prior 3v4, $N = 80$	111
E.1 Force eld contours for case 1	116
E.2 Force eld contours for case 2	117
E.3 Force eld contours for case 3	118
E.4 Force eld contours for case 4	119
E.5 Force eld contours for case 5	119
E.6 Force eld contours for case 6	120
E.7 Force eld contours for case 7	121
E.8 Force eld contours for case 8	121
E.9 Case 1, prior 2, DEnKF	123
E.10 Case 1, prior 3, DEnKF	123
E.11 Case 1, prior 4, DEnKF	123
E.12 Case 2, prior 1, DEnKF	123
E.13 Case 2, prior 3, DEnKF	124
E.14 Case 2, prior 5, DEnKF	124
E.15 Case 2, prior 6, DEnKF	124
E.16 Case 3, prior 5, DEnKF	124
E.17 Case 3, prior 6, DEnKF	125
E.18 Case 3, prior 8, DEnKF	125
E.19 Case 3, prior 9, DEnKF	125
E.20 Case 4, prior 7, DEnKF	125

E.21 Case 4, prior 8, DEnKF	126
E.22 Case 4, prior 9, DEnKF	126
E.23 Case 5, prior 7, DEnKF	126
E.24 Case 5, prior 8, DEnKF	126
E.25 Case 5, prior 9, DEnKF	127
E.26 Case 6, prior 1, DEnKF	127
E.27 Case 6, prior 3, DEnKF	127
E.28 Case 6, prior 6, DEnKF	127
E.29 Case 7, prior 5, DEnKF	128
E.30 Case 7, prior 6, DEnKF	128
E.31 Case 7, prior 9, DEnKF	128
E.32 Case 8, prior 5, DEnKF	128
E.33 Case 8, prior 8, DEnKF	129
E.34 Case 8, prior 9, DEnKF	129

List of Tables

3.1	DBD con guration	20
3.2	DBD cases	21
3.3	Values used to calculate relative variance of PIV data (using (3.27))	25
3.4	Grid speci cations	32
4.1	Parameter values of true solution	38
4.2	Initial prior set-up for twin problem; mean and variance of the parameters, (\cdot^2)	39
4.3	Overview of twin problem EnKF settings	44
4.4	Results SSEnKF using prior 1a/b; Final parameter estimation with corresponding $RMSE_o$ and $RMSE_f$	46
4.5	Overview of $RMSE_f$ results for rst three prior with settings v1	53
4.6	Average $RMSE_o$ and $RMSE_f$ of priors 1,2,3,4, and 6 for settings v1	54
4.7	Comparison results using unbias (v1) and bias (v4) forecast model, True thrust: $T_x = 9.6615$ mN, $T_y = -6.4410$ mN	55
4.8	Filter settings used for data assimilation with PIV	56
5.1	Initial prior set-up; mean and variance of the parameters, (\cdot^2)	60
5.2	Priors used for di erent cases	60
5.3	Best parameter estimations per case	61
A.1	PIV setup	74
D.1	Full results of the data assimilation for the twin problem	96

E.1 Results DEnKF with PIV data, case 1-4	114
E.2 Results DEnKF with PIV data, case 5-8	115

Chapter 1

Introduction

The dielectric barrier discharge (DBD) actuator or just plasma actuator is a promising method of achieving better flow control for e.g. aircraft. The plasma actuator can actively control the flow by transferring momentum into the flow above the actuator. This momentum transfer can be described as a body force imposed on the air. Although the plasma actuator has been studied extensively, a good general description of this body force is still lacking and is essential for flow engineers to design a plasma actuator for optimal flow control. High accurate numerical models, which do not require a general description of the body force field, are impractical for actuator design due to high computational costs.

Plasma actuator have also been studied extensively using experimental method, however, these experimental methods are mostly restricted to measuring the flow field and give only limited information about the force field. This thesis proposes the use of a data assimilation technique called the Ensemble Kalman Filter (EnKF) in combination with experimental data of the flow field generated by the plasma actuator, obtained using Particle Image Velocimetry (PIV), to calculate an estimation of the body force field. Data assimilation is a field of study where experimental data is combined with results from numerical methods in order to find the 'true' state of the system. It assumes that both numerical models as well as experimental data include error, and can be combined to better estimate the state of the system as well as the parameters used to calculate the state used by the numerical model. This combination of experimental data with numerical data also gives additional information in the form of estimated error. Data assimilation techniques can therefore provide a promising new way of calculating body force fields in an accurate but also relatively computationally inexpensive way, while also giving additional information about the parameters in the form of estimated parameter error.

The main goal of this thesis is to create a general method of characterizing the quasi-steady plasma actuator performance in terms of the force field generated using the data assimilation technique in the form of the EnKF.

Other goals of this thesis are:

1. **Reconstruction of close to the wall flow** Most techniques of determining the flow field experimentally have very poor resolution near the wall boundary; something the EnKF could reconstruct.
2. **Approximation of the uncertainty of the estimated parameters.** Most studies in determining the body force field are restricted to only finding a single optimal parameter fit given the experimental data, whereas the EnKF incorporates the uncertainty of experimental data to compute an estimation of the parameter error.
3. **Create a distinction between the pressure gradient and force field.** Velocity based experimental method can not distinct between the pressure gradient and the force field. By using both numerical simulation with the velocity based experiments a distinction can be made.
4. **Creation of a data assimilation tool** which can be built upon to work with many different body-force parameterizations, including time dependent parameterizations. As well as building a foundation for further research in the field of data assimilation with plasma actuators.

The thesis will start off by giving a general description of the plasma actuator and different data assimilation techniques. Chapter 3 will discuss the flow solver, experimental data and the EnKF in detail. Because the EnKF has not been used in these settings yet, a twin problem is set-up using artificially created experimental data in order to test the EnKF in a controlled environment. An in-depth description of the set-up and the results are discussed in chapter 4. The results from the twin problem are used to set up the EnKF in combination with the experimental data to compute the parameter estimation of a total of eight different DBD actuator configurations. The last chapter will discuss some of the main finding of the thesis as well as give some recommendations about further research option to improve the current set-up.

Chapter 2

Background; plasma actuators and data assimilation techniques

The following sections will give a brief overview of: first the operation and state of the art in the field of plasma actuators, and second an overview of data assimilation techniques.

2.1 Plasma actuator

Plasma actuators also called dielectric barrier discharge devices(DBD) have been intensely studied for the last decade or so. One of the main promises of the plasma actuator is the ability to actively control the flow around an airfoil of an aircraft; unobtrusively with low installation and operation costs, while possibly drastically increasing the aerodynamic performance of the aircraft. The amount of research in the field of plasma actuator is vast, only a brief review of the operation will be given in the follow section.

2.1.1 Physical background

The term plasma can be loosely defined as an averagely electrically neutral volume of air containing positive and negative charged particles. This definition can be sharpened by using the definition given by [Kotsonis \(2012\)](#) as that of a system of particles whose collective behavior is characterized by long-range Coulomb interactions. This system of particles, in the case of a plasma actuator, would consist of a range of charged particles including electrons, positive ions, and negative ions.

Plasma can be created by applying an electrical field of sufficient amplitude to a volume of air. The electrical field will cause the formation of ions and electrons. The strong electric

field is created by applying a high voltage of either DC or AC current to two separated electrodes. The amplitude of the electric field needed to surpass the breakdown potential depends on ambient pressure, temperature and molecular species available in the neutral gas. The charged particles created will interact with the electrical field and will therefore be accelerated through the air. These moving ions will transfer some of their momentum to the neutral air. Also, the created electrons will change the conductivity of the bulk volume, enabling a current to flow between the two electrodes. This current is called the barrier discharge when using a DBD devices

The barrier discharge can occur at a range of spacial and time scales, however, the main feature all DBD's have in common is the configuration of one exposed and one insulated electrode using a dielectric material, in combination with the use of an AC voltage potential. The use of an insulated electrode creates a charge up of capacitance on the dielectric material which causes the local field to reverse and locally terminate. To remove this capacitance and re-initiate the electric field, an AC electric potential is required. The use of an AC potential is therefore essential in order to sustain barrier discharges on the macroscopic time scale.

2.1.2 Typical Configuration

As described previously, the DBD actuator consists of two electrodes of which one is exposed to air and the other is insulated using dielectric material. Typical parameters describing a DBD actuator configuration with typical order of values are: width of electrodes (few mm), thickness and material properties of dielectric layer (few mm), horizontal gap width (zero or few mm), actuator frequency (f_{ac}) (from a 100 Hz to a few tens of kHz), and amplitude of the High Voltage (from a few kV to 20 kV). Typical materials used for the dielectric layer are Teflon, kapton, glass, ceramics or Plexiglass (Kotsonis (2012)). The electrodes are placed in an asymmetric configuration where the electrodes not placed directly on top of each other. Figure 2.1 shows a typical configuration of a DBD actuator. The asymmetric configuration and the use of an AC voltage source causes the dielectric discharge to be asymmetric as well as the momentum transfer. This last phenomenon results in an induced velocity in the fluid above the insulated electrode, which is the main reason DBD actuator are being studied today.

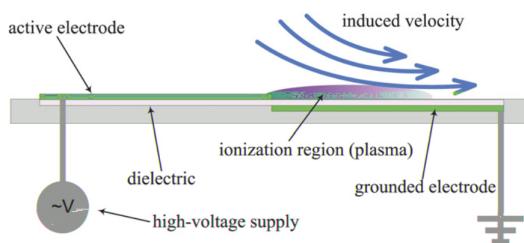


Figure 2.1: Schematic of typical DBD con guration and operation (taken from Kotsonis (2012))

2.1.3 Mechanical and Electrical Behavior

The electrical behavior of the DBD actuator is important in the process understanding the working of the DBD actuator. As discussed above it was already noticed that the electrical field needs to reach a sufficient magnitude to achieve ionization of the air. Other interesting quantities are the discharge current and the high voltage signals (HV) that goes with it. Various AC HV signals can be used for DBD actuators, however, the sine wave has been used and researched the most extensively, and the scope of this thesis will therefore be limited to only sine wave HV.

The electrical behavior can be described well as suggested by (Enloe et al. (2004a) Enloe et al. (2004b)). A plasma actuator cycle can be divided into two parts: the *forward stroke* and the *backward stroke*. The *forward stroke* is defined as the part of the cycle where the exposed electrode is negative. During this cycle the exposed electrode acts as an unlimited source of electrons. The electrons build up on the surface of the dielectric material above the insulated electrode and quench the electrical field to the point where it self terminates. During the *backward stroke* the exposed electrode is positive and this time the electrons are provided by the previously build up of electrons and negatively charged species on the exposed surface of the dielectric material.

The mechanical behavior can be described by several different quantities of interest. The induced velocity, total thrust, and the spatial and temporal body force field distribution. The induced velocity is created by the momentum transfer of the charged species of air, accelerating by the electrical field, to the neutral particles. This quantity can be measured using experimental methods such as hot wire anemometry, particle image velocity or laser Doppler velocimetry. Some of the first measured results of induced velocity were done by Roth et al. (1998) shown in Figure 2.2. The induced velocity profiles have been studied extensively experimentally in the last decade and include very high temporal and spacial resolution data.

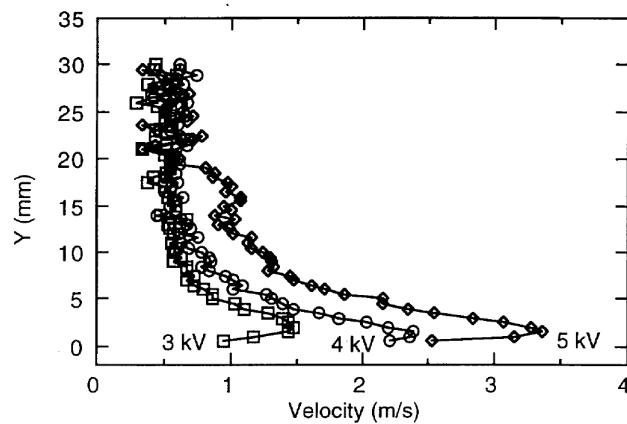


Figure 2.2: Some of the first ever measured induced velocity produced by a DBD actuator (taken from Roth et al. (1998))

With each different shape of HV signal, a different induced velocity can be observed. This induced velocity can be characterized by the shape of the wall jet and the thrust produced.

Thrust produced is defined here as the spatial-temporal averaged body force produced by the actuator. Many studies have captured the thrust production of the DBD actuator with different configurations or HV. However, few researchers have studied the time dependent and spatially distributed body force field, which represents the momentum transfer from the charged particles to the neutral air. This spatial and temporal body force field is of great interest to the flow engineer. Different methods of determining temporal averaged spatial body force have been used by [Kotsonis \(2012\)](#) using experiments. Figure 2.3 shows a body force distribution using a method used by [Kotsonis \(2012\)](#) to determine the body force distribution in continuous operation. Also, [Kriegseis et al. \(2013\)](#) used PIV data together with the Navier-Stokes equations under the assumption that all pressure gradients are zero. [Albrecht et al. \(2011\)](#) uses experimental velocity field together with the vorticity equations, again under the assumption of zero pressure gradient, but now also assuming that the cordewise gradient of the force $\frac{\delta f_y}{\delta x}$ is zero. Most of these body force determination have been done using the assumption of a quasi-steady body force, which is valid for time scale much larger than the actuator frequency $dt >> f_{ac}$. So far very little research has been done in determining the fully time dependent body force distribution. This is due to the lack of high temporal resolution experimental data and additional assumptions used in for example the *gradient method*, used by [Kotsonis \(2012\)](#), that break down when resolving the body force on a smaller time scale. However, recently [Neumann et al. \(2013\)](#) have used laser Doppler velocimetry in order to get high temporal resolution experimental data and obtained a method in determining the body force distribution in time.

It should be noted that all of these results are based on experimental data and are restricted by the accuracy of data and the modeling assumptions made. Almost all measurement techniques (etc. PIV, laser Doppler) have a very poor accuracy near the wall. Also, assumptions made by for example [Neumann et al. \(2013\)](#) result in a body force which has a possible error of atleast 10% of the maximum value determined for the body force. This is many due to the fact that most studies, including [Neumann et al. \(2013\)](#), use the assumption of a relative to the force field neglectable pressure gradient. If further improvement in accuracy of the force field is preferred, the pressure gradient should not be neglected

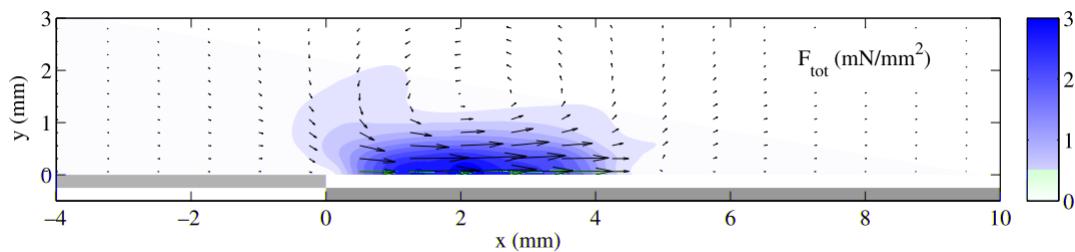


Figure 2.3: Body force determination using *gradient method* (taken from [Kotsonis \(2012\)](#))

2.1.4 Numerical Simulation

Numerical simulation of the plasma actuator can be done in many different ways depending on the goal of the simulation, computational cost restriction, and accuracy requirements.

All methods can be roughly divided in two different types: phenomenological models and first-principle models. The phenomenological models use simple models to get an expression of the body force term and implement this in the Navier-Stokes equations. These models are mostly computational inexpensive and fairly simple to implement. However, they tend to be inaccurate and need a lot of tuning of the body force, based on either experimental measurements or more accurate models, in order to give accurate results. These models also give only a little to no insight into the physical phenomena driving the flow. Shyy et al. (2002) created a model under the assumption that the electric field strength decreases linearly from edge of the exposed electrode toward the covered electrode. However, experimentally it was determined that his charge distribution was exponential Opaits et al. (2008), Cristofolini et al. (2013). Orlov and Corke (2006) made a large improvement in phenomenological models adding to the existing lumped element method first proposed by Enloe et al. (2004b). Singh and Roy (2008) used a full electrostatic model to come up with a Gaussian parameterization for the body force distribution.

On the other hand, the first-principle models use a very large set of equations, including not only the Navier-Stokes equations but also Maxwell equations and must take into account the chemistry of air. These models can result in highly accurate results and give some great insight into the physical phenomena of the plasma actuator. However, the computational cost is very high and the models are very complex. First-principle models are often used to create a parametrization of the body force field for a specific configuration of a plasma actuator. A very good overview of some phenomenological models and especially the first-principle model is given by Jayaraman and Shyy (2008).

2.2 Parameter estimation in data assimilation

Data assimilation encapsulates all techniques that use a combination of real-life measurements and numerical computer models to come up with a closer estimation of the true system state together with an estimation of uncertainty. Data assimilation techniques can be characterized by a measure of objective model performance, such as a cost function that penalizes model errors based on the root mean square error of the model output. Most models have a large number of degrees of freedom making it computationally very expensive to naively explore the model performance for every set of parameters. Even more so if parameters are time dependent. Besides data assimilation techniques, any other optimization technique can be used for the estimation of the parameters. However, these techniques do not directly include the estimation of uncertainty in the model parameters and can result in rather trivial local minima when using highly non-linear system models. Stochastic data assimilation has proven to be more robust in finding a global minimum and directly calculates an estimate of the model uncertainty. A further distinction can be made in data assimilation by so called sequential data assimilation. Sequential data assimilation uses measurements at a single time to update the system state to the, at that moment, closest to the true state of the model. This procedure is repeated every time a new measurement becomes available.

2.2.1 Estimation of optimal model parameters

Estimation of the optimal model parameters can be achieved when extending stochastic data assimilation techniques. Most sequential data assimilation techniques use an augmented state vector, which is an extension of the state space by adding the model parameters. This allows the data assimilation technique to treat the parameters as state variables. In this framework both the spatial as well as the temporal aspects of the parameters can be taken into account [Evensen \(2003\)](#). It should be noted that this method of augmenting the state space can result in additional non-linearities, which can become of concern when using certain types of data assimilation based on linear theories. These non-linearities can be caused by simple multiplication of a parameter with the state variable, or by the on/of switching behavior of the parameter.

2.2.2 Variational minimizing schemes

In order to give both the mean of the estimated parameters and the error of the estimation of the parameters it is important for the data assimilation scheme to be able to incorporate the state-dependence of the covariances. There are several techniques that include this state-dependence, such as four dimensional variational schemes (4DVAR), ensemble Kalman Filter (EnKF) and particle filters. All of these techniques have as objective to minimize the variance of the observed and the numerically computed state, and are therefore also called variational data assimilation schemes. The 4DVAR schemes can be used to estimate both the initial conditions as the model parameters. In order for this scheme to work the adjoint model including the model sensitivity of the parameter has to be calculated. The 4DVAR method is a promising approach for parameter estimation and has been used with success to reduce model error and estimate model parameters. However, the construction of the adjoint of the model including the extension of the model sensitivity of the model parameters can be complex and time consuming. Also, the 4DVAR can fail when the model or the model response to parameters is highly non-linear. Non-linear models can create many local minima and significantly increases the convergence time. Due to the complexity and the potential large computational costs using a non-linear system such as the Navier-Stokes, it is chosen to look for a method that is more robust and computationally cheaper when dealing with non-linear models.

A different set of variational minimizing schemes are those based on the Kalman filter (KF) ([Kalman, 1960](#)). These schemes provide a way to estimate the state error covariance explicitly, saving a lot of computational cost compared to the 4DVAR scheme. The Kalman filter estimates the the optimal system state by combining the prior of the state with the observations using Bayes' theorem. The prior is provided by the integration of the model state from some known prior, either the initial conditions or the previous analysis state. This prior is also called the forecast. Bayes' theorem assumes that all error in both the prior and observed state are multi variant Gaussian. This will be the case if the following two conditions are met. First, the initial prior has to be multi variant Gaussian, and secondly the model dynamics have to be linear. Several methods have been developed based on the KF in order to cope with

non-linear model dynamics. The extended Kalman filter (EKF) computes a tangent linear operator of the model in order to estimate the error covariance in the forecast. This method can be used together with a parameter augmented state space, however, the computational cost and memory requirements associated with this scheme makes it only affordable for small systems.

The Ensemble Kalman filter ([Evensen \(1994\)](#),[Burgers et al. \(1998\)](#)) has been shown to be an affordable alternative to the KF for non-linear systems. The Ensemble Kalman filter uses Monte Carlo, or ensemble integration, to estimate the forecast error covariance. For linear systems with an initial ensemble drawn from a multi variant Gaussian distribution the EnKF reduces to the KF when the number of ensemble member is sufficiently large. The initial prior is made by perturbing the initial conditions in a way to maintain a mostly Gaussian ensemble. When a forecast is made by a non-linear model, the ensemble can still capture higher modes (besides the mean and variance), depending on the ensemble size. In this case the variance calculated from the ensemble used during the analysis step does not represent the variance but more so the perturbation from the optimal values. Therefore, the EnKF becomes suboptimal filter for non-linear systems, whereas the KF is an optimal filter for linear systems. With the EnKF there is no necessity of using an adjoint or tangent operator. It also does not require the need of storing large amounts of data and parallel computing can be used to integrate the ensemble members of the ensemble members. The EnKF is a computationally cheap alternative to the EKF or 4DVAR methods.

The EnKF has been successfully been applied to problems in field of hydrogeological ([Chen and Zhang \(2006\)](#),[Franssen and Kinzelbach \(2009\)](#),[Hendricks Franssen and Kinzelbach \(2008\)](#), and in the field of oceanography ([Anderson \(2001\)](#),[Annan et al. \(2005\)](#), [Yang and Delsole \(2009\)](#), [Skachko et al. \(2009\)](#), [Orescanin et al. \(2009\)](#), [Koyama and Watanabe \(2010\)](#), [Olivares \(1998\)](#), [Hacker and Snyder \(2005\)](#)). Most of these problems are only slightly time dependent and the parameters are assumed constant during forecasting. However, [Yang and Delsole \(2009\)](#) shows that it is possible to estimate time varying parameters with the EnKF with only slight modifications. Also, if the forecast time step is sufficiently small the EnKF does a fairly good job in accurately estimating the time varying parameters. Most of these studies were used to reduce the total model error and have successfully done so. While the main goal was to improve the state estimate, the studies also showed to find a good estimation of the parameters, both in space and time. However, the parameters are assumed constant during integration and can only change during the data assimilation analysis. Therefore, temporal variability of parameters can only be estimated when the time scale of the parameter fluctuation is much larger then that of the time between assimilation steps.

The EnKF is limited to problem which include only small non-linearties between analysis steps because of it's multi-variant Gaussian assumption. A filter that can capture more or all modes of a pdf when integrating forward in time is the particle filter (PF)([van Leeuwen \(2009\)](#), [Doucet et al. \(2000\)](#)). This filter uses the same Monte Carlo method to approximate the pfd but does not use the assumption of multi-variant Gaussianity. This does allow for a more sophisticated description of the pdf, however, sample degeneracy is an issue here. This results in the necessity of using ensemble sizes orders of magnitude larger then used in the EnKF, making the whole system computationally very expensive.

Chapter 3

Methodology

The EnKF will be used to combine an incompressible Navier-Stokes solver with 2D PIV measurements of a quasi-steady flow field generated by an actuator in quiescent flow. The main goal is to determine the values of the parameters describing a quasi-steady body force created by the DBD actuator together with error estimates for these parameters. The complete project including the EnKF and incompressible Navier-Stokes solver have been implemented in Matlab.

3.1 EnKF

This thesis will be limited to the Ensemble Kalman Filter (EnKF) and variations of this technique. The EnKF is a method originally proposed by [Evensen \(1994\)](#) and later slightly modified to the latest definition [Evensen \(2003\)](#). The EnKF is based on the standard Kalman Filter (KF) which is an optimal filter for linear systems evolving in time. The EnKF is a filter known to be able to handle non-linearity in systems [Hendricks Franssen and Kinzelbach \(2008\)](#) and is also able to perform parameter estimation [Evensen \(2009\)](#). The method is relatively simple and has lower computational costs than alternative data assimilation methods such as Particle Filters (PF) or the 4DVAR. The following sections will introduce the EnKF and give an overview of equations used for setting up the problem.

3.1.1 Kalman Filter

The KF is a sequential filter, meaning it integrates a probability density function (pdf) forward in time (*forecast*) and updates this pdf of the state whenever measurements are available (*analysis*). The model is then reinitialized using this new state before forward integration is continued. The KF is a variance-minimizing analysis, it combines both the forecast, created by

the forward integration of the model, and the observations together with their respective error covariance to determine the new state of the model as well as the updated error covariance in which the variance is minimized.

Forecast step

The first step of the KF is the forecast, which integrates a known prior pdf forward in time. The KF assumes a Gaussian state represented here by Ψ .

$$\Psi \sim \mathcal{N}(\cdot; \mathbf{P}); \quad (3.1)$$

where $\cdot \in \mathbb{R}^n$ represents the mean and $\mathbf{P} \in \mathbb{R}^{n \times n}$ represents the covariance matrix. The mean of the state can be integrated forward in time from a known condition \cdot_i using a linear model operator $\mathbf{F} \in \mathbb{R}^{n \times n}$ to create the mean of the forecast as,

$$\cdot_{i+1}^f = \mathbf{F} \cdot_i; \quad (3.2)$$

where $i + 1$ denotes the time level at which measurements are available and superscript f denotes the forecast. The KF assumes the model to be in-perfect which is simulated by stochastic model errors \cdot with zero mean.

$$\cdot \sim \mathcal{N}(0; \mathbf{Q}) \quad (3.3)$$

Now the state covariance can be represented by the combination of the linear operator and the stochastic model noise.

$$\mathbf{P}_t^f = \mathbf{F} \mathbf{P}_k^f \mathbf{F}^T + \mathbf{Q}; \quad (3.4)$$

Eq. (3.2) and (3.4) allows to rewrite (3.1) as the complete state of the forecast as:

$$\Psi_{i+1}^f \sim \mathcal{N}(\mathbf{F} \cdot_i; \mathbf{F} \mathbf{P}_i \mathbf{F}^T + \mathbf{Q}); \quad (3.5)$$

Analysis

The next step of the KF is combining the forecast state with observations using Bayes' rule. This step is called the analysis or sometimes update step.

Assume observations $\mathbf{d}_{i+1} \in \mathcal{R}^m$ with error $\sim \mathcal{N}(0; \mathbf{R})$. The observations relate to the true state \mathbf{d}^{true} as:

$$\mathbf{d}_{i+1} = \mathbf{d}_{i+1}^{true} + \mathbf{e}_{i+1} \quad (3.6)$$

The observations are related to the state space through the observational operator $\mathbf{H} \in \mathbb{R}^{m \times n}$.

$$\mathbf{d}_{i+1} = \mathbf{H}_{i+1}^{true} + \mathbf{e}_{i+1} \quad (3.7)$$

The forecast state and the observations are combined using Bayes' rule which states:

$$p(\mathbf{x}_{i+1} | \mathbf{d}_{i+1}) \propto p(\mathbf{d}_{i+1} | \mathbf{x}_{i+1}) \cdot p_o(\mathbf{x}_{i+1}); \quad (3.8)$$

where $p_o(\mathbf{x}_{i+1})$ is the prior of the state given by 3.5, and $p(\mathbf{d}_{i+1} | \mathbf{x}_{i+1})$ also called the likelihood, and in the case of the KF is given by:

$$p(\mathbf{d}_{i+1} | \mathbf{x}_{i+1}) = \mathbf{d}_{i+1} - \mathbf{H}_{i+1}; \quad (3.9)$$

Combining (3.8) with the equations above one gets the equations for the mean (\mathbf{x}^a) and the covariance (\mathbf{P}^a) of the analysis state:

$$\mathbf{x}^a = \mathbf{x}^f + \mathbf{P}_{i+1}^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_{i+1}^f \mathbf{H}^T + \mathbf{R})^{-1} (\mathbf{d}_{i+1} - \mathbf{H}_{i+1}^f); \quad (3.10)$$

where the analysis error covariance is defined as

$$\mathbf{P}^a = \mathbf{P}_{i+1}^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_{i+1}^f \mathbf{H}^T + \mathbf{R})^{-1} \mathbf{H} \mathbf{P}_{i+1}^f; \quad (3.11)$$

Equations (3.10) and (3.11) are often represented using the Kalman gain matrix:

$$\mathbf{K} = \mathbf{P}^f \mathbf{H}^T (\mathbf{H} \mathbf{P}^f \mathbf{H} + \mathbf{R})^{-1} \quad (3.12)$$

This Kalman gain shows how the Kalman Filter, and Kalman Filter based methods, uses a linear weighted combination of the \mathbf{P}^f and covariance \mathbf{P}^f to construct the re-initialization \mathbf{x}^a and \mathbf{P}^a , based on the measurements and error covariance of both the forecast as the measurements. This combination is chosen to minimize the total variance in the updated state \mathbf{x}^a . The linear update is ideal for updating velocity fields which are divergence free (which will be the case in this study), because of the fact that any linear combination of divergence free fields will again result in a divergence free field. This allows for re-initializing the model without being concerned about creating a non-divergence free initial condition. The EnKF will retain this property as will be discussed later.

The Kalman Filter only applies under the assumption of multivariate Gaussian pdf. This assumption can only hold true if: the prior is multivariate Gaussian, and the prior remains multivariate Gaussian during integration. This requires the condition that the dynamical model is linear as is shown by (3.2) and (3.4).

3.1.2 Ensemble Kalman Filter

When dealing with non-linear systems, multivariate Gaussianity of the pdf is not maintained when integrating forward in time. In order to still give a reasonable estimation of the first two moments (mean and variance) of a pdf, integrated by a non-linear model, the EnKF was developed. The EnKF is based on the idea that you can approximate the pdf by an ensemble of model states. These states are then integrated forward in time individually to create a new ensemble with which the first two moment can be calculated again. This method can still handle small non-linearity as shown by e.g. (Anderson (2001), Annan et al. (2005), Yang and Delsole (2009)).

Again following Evensen (2003) we will discuss how the EnKF is built up. The first extension by the EnKF over the KF is the representation of a pdf using an ensemble of model states. When an infinite amount of different samples are drawn from a pdf any statistical moment of this pdf can be calculated using this ensemble. The EnKF uses only a finite ensemble size and will therefore always only give an approximation of these moments. However, the error of this Monte Carlo sampling will decrease proportional to $1=\sqrt{N}$. This can be done for a full model consisting of dimensions n with an ensemble size of N . This cloud of model states can be used as an approximation to the actual pdf, not necessarily a Gaussian pdf, of the model without the need of storing all moments separately.

The next part of the EnKF is the propagation of the pdf forward in time. The stochastic

differential equation for a non-perfect nonlinear model can be written as:

$$d = \mathbf{f}(\cdot)dt + \mathbf{g}(\cdot)d\mathbf{q}; \quad (3.13)$$

This equation represents the change in the model state d as a combination of the model propagation by $\mathbf{f}(\cdot)$ and the random contribution from a stochastic forcing term $\mathbf{g}(\cdot)d\mathbf{q}$ representing the model errors. $d\mathbf{q}$ is a vector Brownian motion process with covariance $\mathbf{Q}dt$. When additive Gaussian model errors forming a Markov process are used, Kolmogorov's equation (also called Fokker-Planck equation) can be derived describing the time evolution of the pdf (\cdot) of the model state using [Jazwinski \(1970\)](#):

$$\frac{\partial}{\partial t} + \sum_i \frac{\partial f_i(\cdot)}{\partial i} = \frac{1}{2} \sum_{i,j} \frac{\partial^2 (\mathbf{g}\mathbf{Q}\mathbf{g}^T)_{ij}}{\partial i \partial j}, \quad (3.14)$$

where $\mathbf{g}\mathbf{Q}\mathbf{g}^T$ represents the covariance matrix for the model errors and f_i is the i^{th} component number of the model operator \mathbf{f} . (3.14) contains no approximations and the solutions of this equation can be assumed exact. These solutions can then be used to calculate the moments of the pdf such as mean and variance and can be applied to the analysis step of the KF. However, in many problems (3.14) is difficult to solve exactly in the case of non-linear systems, therefore, the EnKF uses a method called the Markov Chain Monte Carlo (MCMC) method to approximately solve (3.14). As described above, a pdf can be described by an ensemble of model states. This ensemble is then integrated forward in time using (3.13). The new ensemble is an equivalent to solving (3.14) under the assumptions that the ensemble is large enough to describe all moments of the pdf. If the dynamical model is linear and the initial ensemble is drawn from a normal distribution the pdf can be described by its mean and variance for all time, reducing the EnKF to the KF. Other methods already exist for solving the Fokker-Planck equation, which are used for the standard KF, but are not applicable for non-linear systems. If the model dynamics are non-linear, the pdf cannot be described by just its mean and covariance. However, the mean and covariance of the pdf still determine the mean path and dispersion about that path (anomalies). The MCMC used by the EnKF can still give an approximation to the solution of (3.14) and is therefore an extension of the KF using non-linear models. If the non-linearities in the model dynamics become weaker, the EnKF error estimations become more accurate and will look more like a regular KF.

The final analysis scheme of the EnKF can be formulated either as a single update of the complete ensemble or as a combination of an update of the mean and an update of the anomalies around that mean. In this thesis we will describe the analysis scheme of the EnKF as the later and will follow [Sakov et al. \(2010\)](#).

Let N be the ensemble size, n the number of model states and m be the number of observations. Then $\mathbf{E} \in \mathbb{R}^{n \times N}$ is a matrix holding an ensemble of size N of n model states. The

ensemble average \mathbf{x} and the anomalies \mathbf{A} can be given by:

$$\mathbf{x} = \frac{1}{N} \mathbf{E} \mathbf{1}; \quad (3.15)$$

$$\mathbf{A} = \mathbf{E} \left(\mathbf{I} - \frac{1}{N} \mathbf{1} \mathbf{1}^T \right); \quad (3.16)$$

where $\mathbf{1} \in \mathcal{R}^N$ is a vector of ones, and $\mathbf{I} \in \mathcal{R}^{N \times N}$ is the identity matrix. Letting superscript f and a refer to the forecast and the analysis variables. The linear ensemble update can then be written as a change in the mean

$$\mathbf{x}^a - \mathbf{x}^f \equiv \mathbf{x} = \mathbf{A}^f \mathbf{G} \mathbf{s}; \quad (3.17)$$

and a change in the anomalies

$$\mathbf{A}^a - \mathbf{A}^f \equiv \mathbf{A} = \mathbf{A}^f \mathbf{T}; \quad (3.18)$$

where $\mathbf{s} \in \mathcal{R}^m$ is the scaled innovation vector

$$\mathbf{s} = \mathbf{R}^{-1/2} (\mathbf{d} - \mathbf{H} \mathbf{A}^f) = \sqrt{N-1}; \quad (3.19)$$

where $\mathbf{d} \in \mathcal{R}^m$ is the vector of observations. $\mathbf{R} \in \mathcal{R}^{m \times m}$ contains the observation error covariance, and $\mathbf{H} \in \mathcal{R}^{m \times n}$ is a matrix of interpolation coefficients mapping the variables from the state to the observations. Matrices \mathbf{G} and \mathbf{T} can be represented in terms of the scaled ensemble observations anomalies $\mathbf{S} \in \mathcal{R}^{N \times m}$:

$$\mathbf{S} \equiv \mathbf{R}^{-1/2} \mathbf{H} \mathbf{A}^f \sqrt{N-1}; \quad (3.20)$$

Matrix $\mathbf{G} \in \mathcal{R}^{N \times m}$ can be written in two ways:

$$\mathbf{G} = (\mathbf{I} + \mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T \quad (3.21a)$$

$$= \mathbf{S}^T (\mathbf{I} + \mathbf{S} \mathbf{S}^T)^{-1}; \quad (3.21b)$$

Eq. 3.21a requires the inverse of a $N \times N$ matrix while (3.21b) requires the inverse of a $m \times m$ matrix and either one may be used to minimize computational cost.

There are numerous types of EnKF's e.g. Deterministic EnKF (DEnKF), Ensemble Transform KF (ETKF), Localized Ensemble Transform KF (LETKF), Localized EnKF (LEnKF). All of these types update the mean in an equivalent way but only differ in the definition of \mathbf{T} for (3.18). For this thesis only two types of the EnKF were studied: the traditional EnKF with perturbed observation (Evensen (1994), Burgers et al. (1998)) and the 'deterministic' EnKF (DEnKF by Sakov and Oke (2008)). Both will be used in this thesis in order to investigate optimal estimations for the given problem.

The traditional EnKF uses the method of perturbed observations. It uses a scaled matrix $\mathbf{D} \in \mathbb{R}^{m \times N}$ of random perturbations sampled from a Gaussian distribution.

$$\mathbf{D} = \frac{\mathbf{R}^{-1/2} \mathbf{D}}{\sqrt{N-1}}; \quad (3.22)$$

where \mathbf{D} has the properties so that $\mathbf{D}\mathbf{1} = 0$ and $\frac{\mathbf{D}\mathbf{D}^T}{N-1} \rightarrow \mathbf{R}$ when $N \rightarrow \infty$. This matrix is used to create \mathbf{T} as

$$\mathbf{T} = \mathbf{G}(\mathbf{D}-\mathbf{S}): \quad (3.23)$$

The DEnKF updates the covariance estimate in a 'deterministic' way and represents a sub-optimal scheme equivalent to the ensemble square root filter (ESRF) in the case of small corrections. Due to the fact that the problem is essentially steady state, corrections are expected to be small, affirming the decision to also applying the DEnKF for this problem. The transform matrix \mathbf{T} for the DEnKF is given as

$$\mathbf{T} = -\frac{1}{2} \mathbf{G} \mathbf{S}: \quad (3.24)$$

The parameter estimation can be included in the EnKF by simply augmenting the state space \mathbf{E} with the parameters. The EnKF will handle the parameters similar to flow variable in the unobserved domain. Also, it should be noted from (3.17)-(3.16) that the EnKF still updates the state of the system by a linear combination of the different ensemble members. Therefore, if the initial ensemble is divergence free, the updated ensemble will also be divergence free.

Eqs. (3.17)-(3.24) form the main set of equations used for the EnKF and have been implemented in Matlab.

3.1.3 EnKF for steady state problem

The EnKF is originally designed to create updates for time-dependent systems. The problem at hand is quasi-steady state problem and the EnKF will therefore reduce to a type of steady state EnKF (SSEnKF). If the problem were to be truly steady and more importantly the prior was multivariate Gaussian, the EnKF would not need to integrate the ensemble update forward in time, i.e. $\mathbf{E}^f = \mathbf{E}^a$, and the filter would converge to a steady state estimation of the field and parameters within a couple updates. As will be discussed more extensively later on in Chapter 4, this method will fail for the case of quasi-steady DBD actuator due to the non-Gaussianity in the prior. The non-Gaussianity of the prior causes a loss of connection between the model state and the parameters after an assimilation step. Therefore, in order to return the correct connection between the flow variables and parameters, a new variation of the SSEnKF is suggested similar to Sørensen (2004). This new variation suggested in this thesis, called the Semi Steady State EnKF (SSS-EnKF), is to integrate the ensemble members forward in time by some random time step dT creating $\mathbf{E}_{t=t_u+dT}^f$ where t_u represents the last time the ensemble was updated. Now $\mathbf{E}_{t=t_u+dT}^f$ is used for the data assimilation. This procedure is repeated until the parameters converge to a specific value. The use of forward time integration between update steps will allow information about the parameters to 'flow' back into the state variables, correcting for previous sub-optimal update steps. The EnKF is expected to converge, although slower than the true SSEnKF. When the covariance of the parameters reduces, the system will start to look more and more linear in this small range of model parameters. The locally semi-linear model will create pdf's which look more Gaussian during integration and allows to assume that the final converged error estimation are a good approximation to the true error. Low values of estimated covariance dependent on the resolution of the observations as well as the correctness of the parameterization of the body force, both of which will be discussed later on.

Imperfections in the model, such as again the non-linearity of the model, sub-optimal update steps, the imperfect parameterization of the body force and other physics not incorporated in the model can cause the EnKF to converge to the wrong values. These imperfections can cause the the EnKF to converge too fast, finding only a local minimum. There are numerous ways to account for these model imperfections, but for this thesis, only multiplicative inflation is used (Anderson, 2001). Multiplicative inflation grows the anomalies after each update by a certain percentage of the anomalies itself, mostly between 1 and 5 %. This is a simple but rather ad-hoc way of introducing additional error in the ensemble to incorporate for the model imperfection. However, the analysis of the impact of different values of inflation or different inflation techniques is beyond the scope of this study and therefore the value of multiplicative inflation will be set to a value of 3%.

3.1.4 Initial EnKF requirements

For the EnKF to give a good estimation of the mean of the both the field and the parameters several requirements need to be met. The term good estimation is defined as an estimation close to the true values, as if the forecast model was very close to linear. The following list

of requirements are expected to be met:

1. Small variance in the ensemble resulting in a close to multivariate Gaussian ensemble because the non-linear system is expected to behave close to linear when covariance in the ensemble is small. This can be achieved by using very accurate measurements and will be discussed in Section 3.2.
2. Unbiased observations. The error in the observations should be limited to unbiased error so after multiple analysis steps the average is close to the true solution. (Subsection 3.2.2)
3. Unbiased forecast. This is a very strong requirement due to the complexity of the underlying physics of the DBD actuator determining the body force. (Section 3.4)
4. The prior needs to be able to capture flow field sufficiently well in order to recreate the observations by linear combinations of the prior ensemble, while maintaining sufficient connectivity between state variables and parameters to avoid divergence. (Section 3.4)
5. Enough inflation needs to be added during assimilation to avoid divergence of the es-)

(PIV) of a series of plasma actuation configurations in quiescent flow. All measurements were done before the start of this study by Marios Kotsonis (Delft University of Technology). The following section gives an overview of the different actuator cases that have been studied, and

Table 3.2: DBD cases

Case #	actuator frequency (f_{ac})	applied voltage (V)
1	2 kHz	8 kV
1	2 kHz	10 kV
2	2 kHz	12 kV
3	2 kHz	14 kV
4	2 kHz	16 kV
5	1 kHz	10 kV
6	3 kHz	10 kV
7	4 kHz	10 kV

3.2.2 Observational error

As discussed briefly in Section 3.1.4, it is important for the observations to have the following properties:

1. Negligible bias error.
2. Error quantifiable in terms of variance.
3. Observational variance to be very small.

Bias error in observations

If the observations are biased, the EnKF would not be able to find the true solution because it will not be included in the prior set of ensemble members. This bias can be introduced into the observations through e.g. 3D flow effects, wall reflection of the laser, compressibility, PIV analysis method. However, free stream velocities do not exceed 6 m=s and can therefore be assumed fully incompressible.

The wall effect is the naming of the reflection of laser light by the wall. This causes the measurements to become very inaccurate. This reflection can be seen during the PIV data analysis in the form of very high signal to noise ratios and non zero values for tangential and normal velocities at the wall. To reduce the bias error this wall effect causes, the observational domain is cropped. The four lowest points of the PIV data are removed. High signal to noise ratios can also be found near the east, west, and north edges of the observational domain. This is due to particles leaving or entering the domain, resulting in suboptimal use of the cross-correlation techniques. This source of error is also corrected by cropping of some points on those sides of the domain. Other sources of error due to the PIV analysis method are assumed to be negligible.

It is assumed that the new observational domain includes enough information to correctly identify the parameters. This also allows for the reconstruction of induced jet flow close to the wall during the data assimilation. This could be of interest for many experimentalists using PIV, and there have been previous attempts in reconstructing boundary layers in the

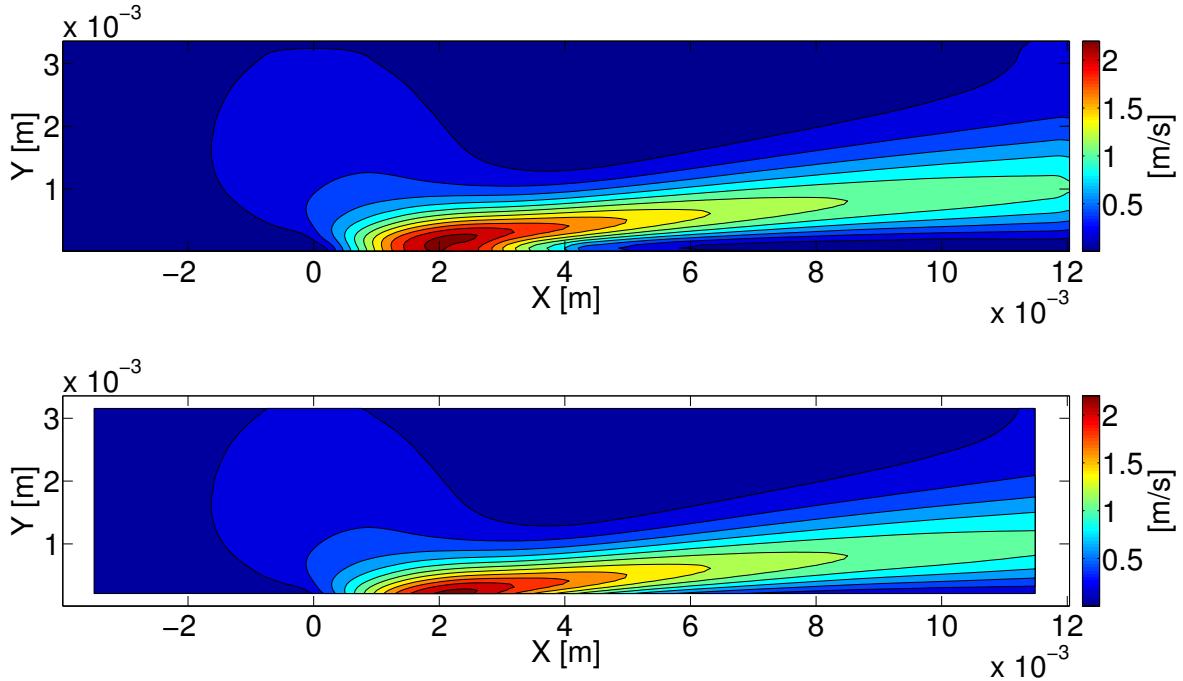


Figure 3.2: Example of total velocity profiles for the observational domain: Full observational domain (*top*) and cropped observational domain (*bottom*).

recent past (Sciacchitano et al., 2011). Figure 3.2 shows the observational domain before and after cropping. Figure 3.2b is used for the data assimilation with the EnKF.

The 3D flow effects can be observed as the divergence of the field under the valid assumption of fully incompressible flow ($U_{max} \simeq 5 \frac{m}{s}$). Figure 3.3 shows the divergence of the flow field for case 2. It can be seen that divergence is very large in the region close to the wall, which is most likely caused by the wall reflection rather than 3D flow effects. This divergence of the field cannot quantify this error in terms of the velocity i.e. $\nabla \mathbf{u}$ has units of s^{-1} . Therefore, use is made of the Helmholtz decomposition in order to come up with a quantitative estimation of the divergence error. The Helmholtz theorem states that any vector field can be decomposed

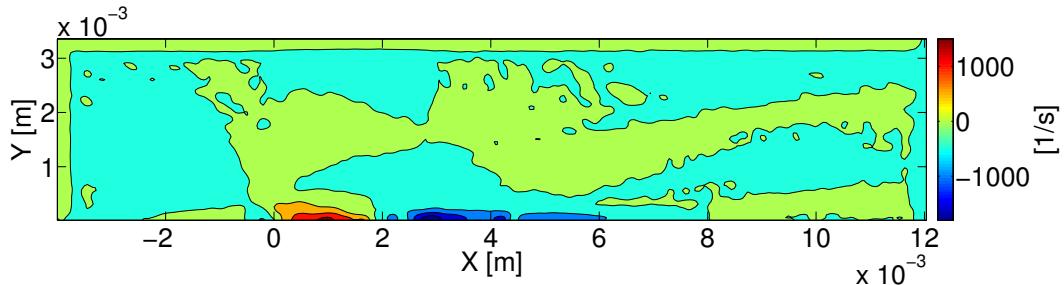


Figure 3.3: Divergence of the flow field in full observational domain for actuator case 2 in s^{-1}

into a solenoidal part and an irrotational part.

$$\mathbf{U} = \mathbf{U}_{irrot} + \mathbf{U}_{sol} = \nabla \cdot \mathbf{A} + \nabla \times \mathbf{A} \quad (3.25)$$

The magnitude of the solenoidal part of the velocity components in the x and y direction could give some of a quantitative estimation of the bias error in the total flow field for each case. However, the method used to decompose the field is essential to the magnitude of the solenoidal part, and any conclusion to these results should be done with care. This is because (3.25) has an unique solution for a given set of boundary condition, where as in most cases this boundary condition is not exactly known. One way of finding the irrotational part of the flow field is by taking the divergence of (3.25):

$$\nabla \cdot \mathbf{U} = \nabla^2 \cdot \mathbf{A} : \quad (3.26)$$

Eq. (3.26) is a Poisson problem and can be solved for \mathbf{A} , which can be used to determine the \mathbf{U}_{irrot} and \mathbf{U}_{sol} . The boundary condition for the Poisson problem were to be taken of the Dirichlet type ($\mathbf{A}' = \nabla U$) with values equal to the values of the velocity found from the PIV. However, this method is most likely to fail because of the incorrect boundary conditions due to the high error in the lower region of the PIV domain.

A different method of decomposing the field is to use Sequential Matching Pursuit filtering (SMP) proposed by Schiavazzi et al. (2012), proposes to decompose the velocity field as a linear combination of local solenoidal waveforms. This method has additional advantages over most other methods in that large local errors will not be propagated to neighboring velocities.

Figures 3.4 and 3.5 show the results of the SMP method.¹ For this method the cropped version of the PIV window is used, in an attempt to only quantify the bias error that will be included during the data assimilation. Both figures show larger values for the irrotational part of the flow field near the regions of high divergence, as expected. Especially the velocity in the y-direction shows high relative error near the region of plasma actuation. The bias error around the region of the plasma actuator could be contributed to the transient start up of the actuator still present or the averaging method used to construct the quasi-steady flow field. Even more points could be cropped off the bottom of the PIV window to reduce this error further, with the risk of losing potential essential information about the flow field. The data assimilation has been done using the unfiltered data. This is done because it is not directly evident that the divergence free field shown in figures 3.4 and 3.5 is the 'true' field. i.e. any different method of decomposition could result in a different divergence free field.

Other systematic error can also be present in the measurements due to for example misalignment of the field of view etc. These types of bias error are not quantified in this thesis and are assumed to be negligible.

¹A code set-up by Ir. I. Iliass Azijli was used to compute the decomposition.

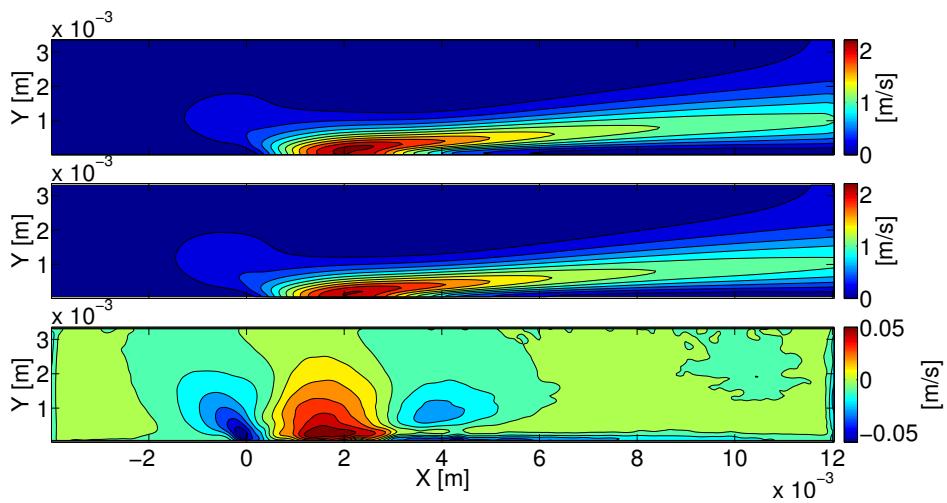


Figure 3.4: Helmholtz decomposition results velocity in x-direction u . From top to bottom: u from PIV, u_{sol} , u_{irrot} .

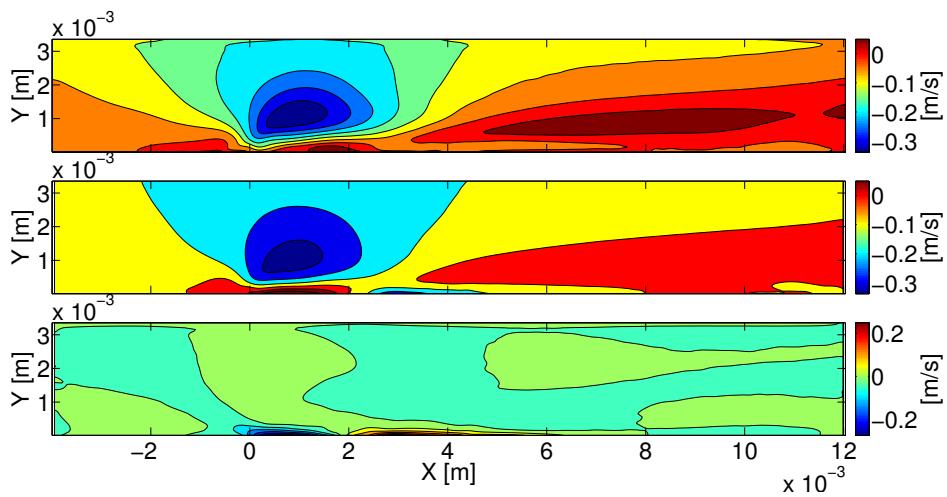


Figure 3.5: Helmholtz decomposition results velocity in y-direction v . From top to bottom: v from PIV, v_{sol} , v_{irrot} .

Quantification of variance in observations

For the EnKF to give a good estimation of both the mean and the associated error of the estimated parameters it is essential that the system becomes as close to linear as possible. This is achieved by using observations with very small, and unbiased error. When running the modified version of the EnKF, the ensemble covariance will reduce to values smaller than the covariance in the observations. The smaller the ensemble covariance, the more the model behaves as a linear system in the range of the ensemble.

In order to start the EnKF, this unbiased error needs to be quantified in terms of variance. A simple method of approximating this error is to follow [Raf \(2007\)](#) which suggest that PIV error is typically around 0.1*pixel size. Using [Scarano \(2007\)](#) we can come up with an analytical estimation of the relative variance of the PIV.

$$\frac{U}{U} = \sqrt{\left(\frac{\Delta t}{\Delta t}\right)^2 + \left(\frac{\Delta x}{\Delta x}\right)^2}; \quad (3.27)$$

where, Δt is the time between the two short time measured PIV window, Δx is the 0.1 * pixel size, Δt is the laser pulse frequency, Δx is a quarter of the PIV interrogation window.

Table 3.3 shows the values used to calculate the observation variance. The value of 0.0125*U will be used for all cases. Additional observation error of 0.01 $\frac{m}{s}$ is added to incorporate for extra unknown error.

Table 3.3: Values used to calculate relative variance of PIV data (using (3.27))

	value
Δt	$1e^{-9}$ s
Δx	$1.56e^{-4}$ m
Δt	$0.1e^{-3}$ s
Δx	$1.25e^{-5}$ m
$\frac{\sigma_U}{U}$	0.0125

3.3 Incompressible flow solver as forecast model

As has been discussed in 2.1 a plasma actuator can be simulated using various different models. For this thesis a phenomenological model is used in the form of a laminar incompressible Navier-Stokes solver. The assumption of incompressibility and laminar flow has been verified by several studies (e.g. [Kotsonis \(2012\)](#)). The momentum transfer between the ionized particles and the neutral particles in the air will be simulated using a body force field. This body force can be parameterized reducing the different number of parameters and therefore simplifying the problem. First an overview will be given of the set-up of the flow solver. In the data assimilation jargon this flow solver will be used as forecast model. The forecast model should meet the following criteria:

1. Discretization error, both in space and time, should be negligible compared to measurement error. This allows the variance in the forecast ensemble only to be caused by the perturbed set of parameter and no bias is introduced. (Note, the forecast model was initially set-up to also be able to use time dependent body force implementations)
2. Computationally as cheap as possible without violating criteria 1.
3. Able to handle a wide range of different shapes and magnitudes of body forces.

The following section will cover the description of the flow solver constructed and will discuss how the flow solver meets the criteria above. A discretization scheme that proved to meet these criteria was a central in space and forward Euler in time discretization.

The governing equations of the dimensional incompressible Navier-Stokes which are solved for read as follows:

Conservation of mass,

$$\nabla \cdot \mathbf{u} = 0; \quad (3.28)$$

Conservation of momentum,

$$\frac{D\mathbf{u}}{Dt} - \Delta \mathbf{u} = -\frac{\nabla p}{\rho} + \mathbf{f}; \quad (3.29)$$

where, \mathbf{u} is the vector field containing the velocities in the x and y direction, Δ is the kinematic viscosity, ρ is the air density, p is the pressure field and \mathbf{f} is the vector field of the body force distribution.

A staggered grid approach is used. In such a grid, the pressures are defined at the midpoints of the cell and the velocities are defined at the midpoints of the walls of the cell.

3.3.1 Conservation of Mass

A finite volume approach is used to discretize the equations. A control volume is taken around the point where the pressure is defined. The continuity equation (3.28) is then integrated over this volume.

$$\int_{\Omega_k} \nabla \cdot \mathbf{u} d\Omega_k = 0 \quad (3.30)$$

To convert this volume integral to an integral over the boundaries of the cell, Gauss' Theorem is applied. This leads to (3.31).

$$\int_{\partial\Omega_k} \mathbf{u} \cdot \mathbf{n} dS \approx 0 \quad (3.31)$$

In order to discretize (3.31), the boundary integral is split along the sides of the control volume and each integral is then evaluated using the midpoint rule. The finite volume discretization of the equation leads to,

$$u_{i+\frac{1}{2},j} \cdot h_j + v_{i,j+\frac{1}{2}} \cdot h_i - u_{i-\frac{1}{2},j} \cdot h_j - v_{i,j-\frac{1}{2}} \cdot h_i = 0; \quad (3.32)$$

The terms h_i and h_j represent the width and the height of the cell respectively.

3.3.2 Conservation of momentum

The conservation of momentum equations is defined by equation 3.29. Rewriting the momentum equation in the x-direction,

$$\frac{\partial \mathbf{u}}{\partial t} + u \frac{\partial \mathbf{u}}{\partial x} + v \frac{\partial \mathbf{u}}{\partial y} + \frac{\partial p}{\partial x} = \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \frac{\mathbf{f}_x}{\rho}; \quad (3.33)$$

which can be rewritten in the gradient form as,

$$\frac{\partial u}{\partial t} + \vec{\nabla} \cdot \begin{pmatrix} u^2 + p - \frac{\partial u}{\partial x} \\ uv - \frac{\partial u}{\partial y} \end{pmatrix} = \frac{\mathbf{f}_x}{\rho}; \quad (3.34)$$

Integrating (3.34) over the control volume Ω_k yields,

$$0 = \int_{\Omega_k} \left[-\frac{\mathbf{f}_x}{\rho} + \frac{\partial u}{\partial t} + \vec{\nabla} \cdot \begin{pmatrix} u^2 + p + \frac{\partial u}{\partial x} \\ uv - \frac{\partial u}{\partial y} \end{pmatrix} \right] d\Omega \quad (3.35)$$

$$= \int_{\Omega_k} -\frac{\mathbf{f}_x}{\rho} + \frac{\partial u}{\partial t} d\Omega + \int_{\partial\Omega_k} \begin{pmatrix} u^2 + p - n u \frac{\partial u}{\partial x} \\ uv - \frac{1}{Re} \frac{\partial u}{\partial y} \end{pmatrix} \cdot \begin{pmatrix} n_x \\ n_y \end{pmatrix} \cdot dS. \quad (3.36)$$

Where, $\vec{n} = (n_x; n_y)^T$ is the unit normal pointing outwards from the boundary of the control volume Ω_k .

Equation 3.36, is discretized over a control volume surrounding the unknown velocity.

When solving this equation for the velocity at the next time level one gets:

$$u_{i+\frac{1}{2},j}^{n+1} = -\Delta t \cdot \frac{\rho_{i+1,j} - \rho_{i,j}}{\frac{1}{2}(h_i + h_{i+1})} + R u_{i+\frac{1}{2},j}; \quad (3.37)$$

where Ru is given by:

$$\begin{aligned} Ru_{i+\frac{1}{2},j} &= \frac{f_{x(i+\frac{1}{2},j)}}{\Delta t} + u_{i+\frac{1}{2},j}^n \frac{1}{2}(h_i + h_{i+1}) \\ &\quad - \left[\frac{u_{i+1,j}^2 - u_{i,j}^2}{\frac{h_i + h_{i+1}}{2}} \right] + \left[\frac{u v_{i+\frac{1}{2},j+\frac{1}{2}} - u v_{i+\frac{1}{2},j-\frac{1}{2}}}{h_j} \right] \\ &\quad + \Delta t \left\{ \frac{\left(\frac{\partial u}{\partial x} \right)_{i+1,j} - \left(\frac{\partial u}{\partial x} \right)_{i,j}}{\frac{h_i + h_{i+1}}{2}} + \frac{\left(\frac{\partial u}{\partial y} \right)_{i+\frac{1}{2},j+\frac{1}{2}} - \left(\frac{\partial u}{\partial y} \right)_{i+\frac{1}{2},j-\frac{1}{2}}}{h_j} \right\}. \end{aligned} \quad (3.38)$$

A similar discretization can be done for the y-momentum resulting in:

$$v_{i,j+\frac{1}{2}}^{n+1} = -\Delta t \cdot \frac{\rho_{i,j+1} - \rho_{i,j}}{\frac{1}{2}(h_j + h_{j+1})} + R v_{i,j+\frac{1}{2}}; \quad (3.39)$$

with Rv being given by:

$$\begin{aligned}
Rv_{i,j+\frac{1}{2}} = & \frac{f_{y(i,j+\frac{1}{2})}}{\Delta t} + v_{i,j+\frac{1}{2}}^n \frac{1}{2}(h_j + h_{j+1}) \\
& - \left[\frac{v_{i,j+1}^2 - v_{i,j}^2}{\frac{h_j + h_{j+1}}{2}} \right] + \left[\frac{uv_{i+\frac{1}{2},j+\frac{1}{2}} - uv_{i-\frac{1}{2},j+\frac{1}{2}}}{h_i} \right] \\
& + \Delta t \left\{ \frac{\left(\frac{\partial v}{\partial y} \right)_{i,j+1} - \left(\frac{\partial v}{\partial y} \right)_{i,j}}{\frac{h_j + h_{j+1}}{2}} + \frac{\left(\frac{\partial v}{\partial x} \right)_{i+\frac{1}{2},j+\frac{1}{2}} - \left(\frac{\partial v}{\partial x} \right)_{i-\frac{1}{2},j+\frac{1}{2}}}{h_i} \right\} ;
\end{aligned} \tag{3.40}$$

In order to satisfy the conservation of mass (3.28), (3.37) and (3.39) are substituted into (3.32) for u and v .

$$\begin{aligned}
Ru_{i+\frac{1}{2},j} \cdot h_j - \Delta t \frac{p_{i+1,j} - p_{i,j}}{\frac{1}{2}(h_i + h_{i+1})} \cdot h_j + Rv_{i,j+\frac{1}{2}} \cdot h_i - \Delta t \frac{p_{i,j+1} - p_{i,j}}{\frac{1}{2}(h_j + h_{j+1})} \cdot h_i \\
- Ru_{i-\frac{1}{2},j} \cdot h_j - \Delta t \frac{p_{i,j} - p_{i-1,j}}{\frac{1}{2}(h_{i-1} + h_i)} \cdot h_j - Rv_{i,j-\frac{1}{2}} \cdot h_i - \Delta t \frac{p_{i,j} - p_{i,j-1}}{\frac{1}{2}(h_{j-1} + h_j)} \cdot h_i = 0;
\end{aligned} \tag{3.41}$$

This completely eliminates the flow variables at the next time level and results in a linear system of equations which can be solved for the pressure. This pressure field satisfies the continuity and can be used to calculate \mathbf{u} at the next time level using eq. 3.37, and eq. 3.39. The partial derivatives of u and v as well as the square terms and the cross terms are approximated using a finite difference as follows:

$$\left(\frac{\partial u}{\partial x} \right)_{i,j} = \frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{h_i}; \tag{3.42}$$

$$\left(\frac{\partial u}{\partial y} \right)_{i,j} = \frac{u_{i,j+\frac{1}{2}} - u_{i,j-\frac{1}{2}}}{\frac{h_j + h_{j+1}}{2}}; \tag{3.43}$$

$$u_{i,j}^2 \approx \frac{u_{i+\frac{1}{2},j}^2 + u_{i-\frac{1}{2},j}^2}{2}; \tag{3.44}$$

$$uv_{i+\frac{1}{2},j+\frac{1}{2}} \approx 2 \left(\frac{h_j \cdot u_{i+\frac{1}{2},j+1} + h_{j+1} \cdot u_{i+\frac{1}{2},j}}{h_j + h_{j+1}} \cdot \frac{h_i \cdot v_{i+1,j+\frac{1}{2}} + h_{i+1} \cdot v_{i,j+\frac{1}{2}}}{h_i + h_{i+1}} \right) \tag{3.45}$$

3.3.3 Time integration

The scheme described in the previous section is conditionally stable. During simulation, the time stepping is not only restricted by this conditional stability but also by the necessity of very low discretization error in time. As stated earlier, the discretization error should be sufficiently small so it can be ignored during the data assimilation compared to the variance introduced by the variance in parameters. Therefore, a list of restriction can be set-up for the largest time step:

$$dt \leq \frac{h_{min}^2}{4}; \quad (3.46)$$

$$dt \leq \frac{2}{(|u| + |v|)}; \quad (3.47)$$

$$dt \leq \Delta t_{disc}; \quad (3.48)$$

where Δt_{disc} is a user specified time step to ensure sufficiently low discretization error. The time step used by the solver is based on the smallest of the values specified by eq. 3.46, 3.47, 3.48.

Other time schemes have been investigated for the time integration. The forward Euler scheme described above is only first order accurate in time. If higher temporal resolution is needed, perhaps in future studies which investigate the fully time resolved bodyforce field instead of the quasi-steady state force field, higher order schemes are needed. To achieve this higher order explicit schemes can be used, such as central difference in time or even a runge-kutta implementation. However, when dealing with incompressible flow, it is not evident that a higher order time discretization for the velocity will directly result in an equal increase in accuracy. This is due to the fact that the pressure field will still be solved with only first order accuracy and will therefore reduce the accuracy of the calculated velocity field. For a good overview of higher order explicit runge-kutta schemes for incompressible flow the reader is referred to [Sanderse and Koren \(2012\)](#).

3.3.4 Boundary conditions and grid

The initial condition is condition is specified in every case as zero velocity in the whole domain at $t = 0$. The quasi-steady plasma actuator can further be simulated using a constant in-time body force field. Different set of boundary conditions can be specified in order to correctly simulate the flow field generated by the plasma actuator. When starting the simulation a start-up vortex can be observed which needs to either move out of the domain or far away from the area of interest to ensure a steady state solution. The integration forward in time to steady state should be as efficient as possible due to the fact that this procedure has to be applied to each individual ensemble member.

Let Ω denote the domain with $\partial\Omega$ its boundaries. Let $\Omega = \Omega_w \cup \Omega_o$. Ω_w is the boundary

of the wall, and Ω_o denotes the open boundaries of the domain. The bottom of the domain is simulated as a flat plate with zero slip velocity and zero normal velocity.

$$\mathbf{u} = 0 \quad \text{on } \Omega_w; \quad (3.49)$$

For the open domain Ω_o , two main approaches have been studied in order to meet the criteria given in section 3.3 with slight variations in boundary conditions.

1. Small computational domain where the start-up vortex moves out of the domain completely. Has as main advantage very low computational cost due to small amount of grid points.
2. Large computational domain with coarsening of the grid towards the sides of the domain where the start-up vortex is always simulated while it dissipates. Has higher computational costs than option 1 due to larger number of grid points and longer time until convergence to steady state. This option could prove more robust because there will be no boundary reflections when changing the bodyforce during data assimilation and has already been used by [Kotsonis \(2012\)](#) in simulating the flow field.

Several different sets of boundary conditions for the small domain have been studied in order to let the start-up vortex move out of the domain without reflections. Ranging from constant pressure boundaries (3.50), to a combination of zeros shear stress and total velocity,pressure combination.

$$\rho\mathbf{n} = 0 \quad \text{on } \Omega_o; \quad (3.50)$$

$$\rho\mathbf{n} - \mathbf{n} \cdot \nabla \mathbf{u} = 0 \quad \text{on } \Omega_o; \quad (3.51)$$

$$\rho\mathbf{n} + \frac{1}{2}|\mathbf{u}|^2 \cdot \mathbf{n} = 0 \quad \text{on } \Omega_o; \quad (3.52)$$

$$\rho\mathbf{n} - \mathbf{n} \cdot \nabla \mathbf{u} + \frac{1}{2}|\mathbf{u}|^2 \cdot \mathbf{n} = 0 \quad \text{on } \Omega_o; \quad (3.53)$$

$$\rho\mathbf{n} - \mathbf{n} \cdot \nabla \mathbf{u} + \left[\frac{1}{2}|\mathbf{u}|^2 S_o(\mathbf{n} \cdot \mathbf{u}) \right] \cdot \mathbf{n} = 0 \quad \text{on } \Omega_o; \quad (3.54)$$

where \mathbf{n} denotes the unit normal vector of the boundary, and S_o is a smoothing function suggested by [Dong et al. \(2014\)](#) in order to reduce outflow boundary reflections. These boundary conditions have been implemented in the flow solver described above, but have proven unable to handle the start up vortex. Even though these type of

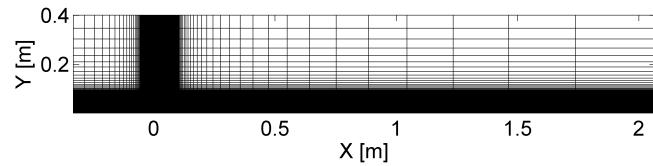


Figure 3.6: Picture of complete grid

boundary conditions have proven to do reasonably well for most simulations (reference something), the reflections could be caused by the absence of a free flow velocity moving any reflection out of the domain.

Therefore, the large domain approach was used. For the large domain the same initial condition with zero velocity were used. Boundaries conditions have been set to constant pressure (3.50) and zero velocity at the wall (3.49). The grid needs to be able to capture the full jet inside the PIV window while being able to dissipate the start-up vortex and any other start up anomalies.

For simplicity and accuracy the computational grid inside the PIV window is taken equal to that of the PIV measurements. The grid is then coarsened with a linearly towards the sides, and finally an extra layer of 10-20 lines is added with exponentially growing spacing. These exponential layers have been added because the flow solver needs to run for longer periods of time and be more robust to different force fields. Any vorticity to move into this region is expected to dissipate before it reaches the boundary. Table 3.4 shows the characteristic values for the grid and figure 3.6 shows a picture of the numerical grid used. The flow solver was tested by running a body force field determined in [Kotsonis \(2012\)](#) for case 5, which displays the highest induced velocities. Figure 3.7 shows this comparison. Difference between the solutions can be contributed by different time stepping schemes and the difference in grid resolution.

Table 3.4: Grid specifications

	value
Size PIV window	$16.03 \times 3.35 \text{ (mm)}$
Cells PIV window	322×74
Size numerical domain	$2.39 \times 0.40 \text{ (m)}$
Cells numerical domain	534×189
$dx _{y=0}$	0.05 (mm)
$dy _{y=0}$	0.05 (mm)

3.3.5 Measurement operator

The mapping from the forecast domain to the observational domain is done by linear interpolation of the staggered grid to the non-staggered PIV domain. The observational operator \mathbf{H} is therefore completely linear.

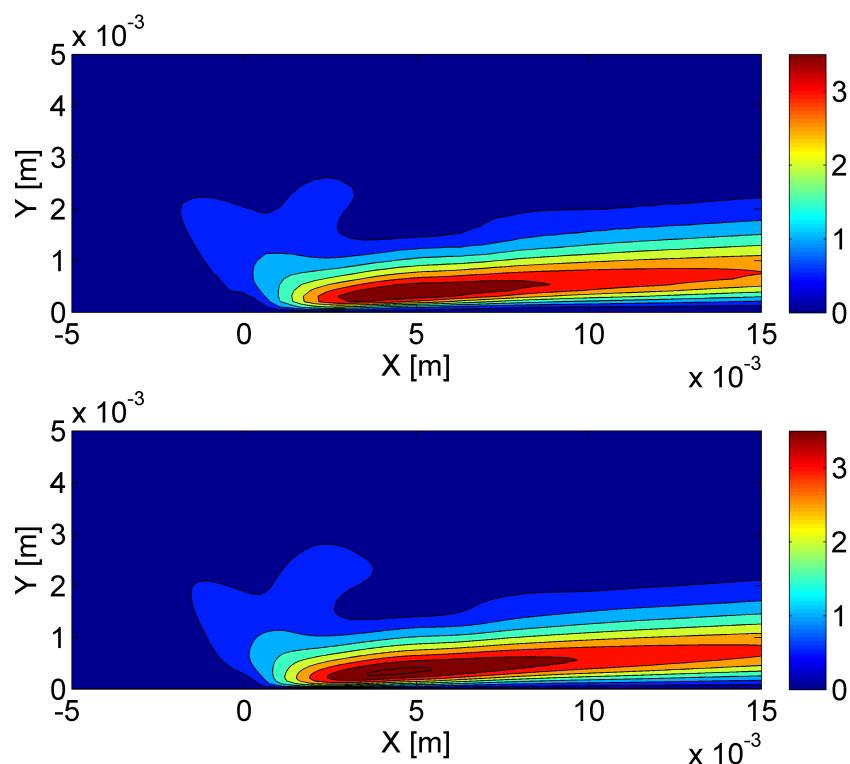


Figure 3.7: Comparison numerical solver. Above: Kotsonis (2012) using OpenFoam, below: Matlab code using grid defined by table 3.4

3.4 Initial set-up

3.4.1 Body force parameterization

For the EnKF to be able to estimate the parameters correctly, the parameterization has to be able to capture the solution sufficiently well to be able describe the flow field with reasonable accuracy. If not, the EnKF might not be able to find a good solution, and might even diverge. However, the number of parameters should also be kept relatively low, and should be mostly restricted to physical shapes and magnitudes. This is to ensure minimum computational cost by lower number of ensemble members necessary.

Parameters added to the state space do not necessarily need to be identifiable i.e. the forecast state in the observational domain depends on the parameter. Due to the stochastic nature of the EnKF, these parameters will automatically grow their covariance towards a uniform distribution. However, one interesting aspect concerning the DBD plasma actuator arises when using only velocity based measurements in determining the body force field. Starting with the incompressible steady Navier-Stokes equation:

$$CD(\mathbf{u}) = -\nabla p + \mathbf{f}; \quad (3.55)$$

where $CD(\mathbf{u})$ are the convective and diffusive terms. $CD(\mathbf{u})$ can be calculated using the PIV data. This leaves the right hand side of (3.55) in the form of the pressure and the body force as unknowns. The pressure will adapt in order to satisfy the conservation of mass, however, it is not evident that this makes the problem unique. Using Helmholtz decomposition to decompose the body force in an irrotational part \mathbf{f}_f and a solenoidal part \mathbf{A}_f we get:

$$CD(\mathbf{u}) = -\nabla p + \nabla \cdot \mathbf{f}_f + \nabla \times \mathbf{A}_f \quad (3.56)$$

$$= -\nabla(p + \phi_f) + \nabla \times \mathbf{A}_f; \quad (3.57)$$

(3.57) shows that when using only the velocity based data, no distinction can be made between the pressure and the irrotational part of the body force. Only the solenoidal part of the body force field \mathbf{A}_f will be unique. One way to solve this problem is to state that the pressure gradients are relative small and can be ignored. However, in an attempt to improve on the body force field estimation a different approach will be used in this thesis. This is done by using a fixed parameterization of the body force field which will have a unique value for ϕ_f and \mathbf{A}_f for all parameters. If this parameterization is assumed to be physical, any resulting pressure gradient can be assumed physical as well.

As described in section 2.1.4, numerous ways of body force parameterizations exist. For the application to this EnKF problem it was chosen to work with an adaption of the body force

parameterization used by [Singh and Roy \(2008\)](#). This force parameterization was done using a full electrostatic model to which a Gaussian fit was applied. The adaption made to the existing model parameterization were to ensure physical solutions when the Kalman Filter is running. The existing parameterization by [Singh and Roy \(2008\)](#) is:

$$\mathbf{f} = F_{x0} \exp\left\{-\left[\frac{x - x_0 - y - y_0}{y^2}\right]^2 - x(y-y_0)^2\right\} \hat{i} + F_{y0} \exp\left\{-\left[\frac{x - x_0}{y}\right]^2 - y(y-y_0)^2\right\} \hat{j}; \quad (3.58)$$

where F_{x0} and F_{y0} are the maximum force magnitude in the x- and y-direction dependent on the actuator HV. y_0 and x_0 determine the position of the force, and again y_0 together with x and y are forming parameters for the shape of the field. In order to restrict the body force to a more physical set of solutions the following adaptations were made: First, the fourth order increase in maximum force dependent on the actuator HV was removed. Secondly, the shape of the force field in both x- and y-direction are coupled. Thirdly, the values for x and y_0 are taken to be absolute values. Negative values for these parameters result in very 'strange' force fields which could prove the flow solver to fail. Taking the absolute value for these parameters does create some high non-linearities when the ensemble includes both positive and negative values for this parameter. Lastly, an extra parameter is added to be able to move the complete force field up and down. The resulting parameterization is given as follows:

$$\mathbf{f} = (A_x \hat{i} + A_y \hat{j}) \exp\left\{-\left[\frac{x - x_0 - y - y_{off} - |y_0|}{y - y_{off}}\right]^2 - |(y - y_{off} - |y_0|)|^2\right\}; \quad (3.59)$$

It is assumed that (3.59) can sufficiently well capture the body force field generated by the DBD actuators studied in this thesis. This parameterization is especially attractive for the EnKF because of its low number of parameters and low cost of implementation (i.e. no additional models need to be solved for). The main downside of the parameterization is that it still cannot perfectly capture the physical body force, most likely increasing the final root mean square error of the system. Also, the use of taking the absolute values for x and y_0 is not ideal for the EnKF. It is assumed that if the ensemble only occasionally includes negative values for these parameters, and will therefore not influence the mean and error estimation significantly.

3.4.2 Solver stability

Due to the fact that a conditionally unstable time integration scheme is used, some extra aspects of concern arise. The forecast model has been set up to check the restrictions on the time step given by (3.46)-(3.48) every forward integration step it computes. This ensures the

stability of the forecast model during integration. However, the most stringent time constraint using the set-up described in section 3.3.4 is (3.47) dependent on the flow velocity. As will be shown in later on in Chapter 4, the linear update method used during assimilation applied to the non-linear system creates very high velocities in the far field. These nonphysical high velocities cause the forecast model to become computationally unfeasible. In an attempt to make the integration computationally cheaper a '*sponge-layer*' was introduced. This *sponge-layer* is a function $f_{\text{sponge}}(x; y)$ ranging between one, inside the PIV window, and a ratio between the maximum velocity existing inside the PIV window and maximum velocity existing in the far field $r_{\text{sponge}} = \frac{U_{\text{maxPIV}}}{U_{\text{maxfarfield}}}$, at the far field. The value U_{maxPIV} is selected from the flow fields of all ensemble members, whereas $U_{\text{maxfarfield}}$ is determined only by a single ensemble member. The function decreases linearly from one to r_{sponge} in the 'buffer' region where the grid spacing increases linearly. This function is multiplied with the flow field of a single ensemble member if $U_{\text{maxfarfield}} > (U_{\text{maxPIV}} + 1)$. The resulting ensemble will have reduced the far field velocities of equal order of magnitude as flow velocities inside the PIV window.

This method of reducing far field velocities can introduce additional bias into the system. If high bias i.e. high values of r_{sponge} , are used, the results of the EnKF can become useless. Therefore, the EnKF is set up to stop filtering as soon as r_{sponge} drops below 0.05. These very high sponge ratios can emerge due to high anomalies occurring close to the PIV window, inside the buffer region, and are not suppressed sufficiently by the proposed *sponge-layer* method.

Chapter 4

Twin Problem

In order to study the capability of the EnKF, in combination with the forecast model described above, to capture the true solution of a plasma actuator body force, a twin problem is set up. A twin problem in data assimilation is the use of artificially created measurements as observations in order to remove the aspect of bias observations, as well as knowing the true solution. This gives a controlled environment in which the sensitivity of the problem to different settings of the EnKF can be analyzed.

The main goal of the twin problem is to prove the feasibility of finding a true set of parameters describing the quasi steady body force field in the high non-linear system of Navier-Stokes. Sensitivity of the solution will be measured with respect to:

1. Ensemble size (N).
2. Initial prior integration time (T_o).
3. Initial prior random seed initialization.
4. Initial prior mean and covariance.
5. Integration time between assimilation steps (dT).
6. Use of a bias body force.

Initial prior denotes the prior which is used for the first update of the EnKF, $\mathbf{E}_{T_o}^f$, where T_o is the integration time from initial quiescent flow. A total of 24 different initial priors have been set up.

The chapter will first cover the set-up of the twin problem and the results of the twin problem will be discussed in the second part. The last part of the chapter includes recommended settings for the data assimilation with PIV data as observations.

4.1 Set-up

This section will cover the set-up of the twin problem, including the creation of the artificial observations and the creation of the initial prior used to start the filtering. The statistical properties of the initial priors are described by some of their univariate and bivariate statistical properties. The last part of this section the parameter sensitivity of the forecast model is discussed and an overview is given of the different filter settings used for the twin problem.

4.1.1 Creation of the artificial observations

A true solution of the flow field will be used to create the artificial observations for the twin problem. This true solution is created using the forecast model. The forecast model is initialized with quiescent flow with a fixed set of body force parameters. The grid has been refined by doubling the number of points in order to account make the true solution more accurate then the forecast model can predict. The model is then integrated forward in time to $t = 1.0$ s. From this point the integration continues to $t = 2.0$ s, while saving all the flow variables every 0.1 ms. These flow variables are then used to create the observations using:

Parameter	value
$F_x [N=m^2]$	3000
$F_y [N=m^2]$	-2000
$x_o [m]$	0.001
$y_o [m]$	0.001
	$1e^6$
$y_{offset} [m]$	0

$$\mathbf{d} = \mathbf{H}\mathbf{x}^{true} + : \quad (4.1)$$

The parameters settings used for the creation of the true solution are given in table 4.1. The averaging method described in Section 3.2 does not result in observations without any unbiased noise and the observational error used to create the observations from the true solution in (4.1) is therefore taken to be zero ($\sim \mathcal{N}(0;0)$).

4.1.2 Creation of the initial prior

The initial priors can be defined by; (1) parameters mean and covariance, (2) ensemble size, (3) integration time, and(4) random seed initialization. The priors are created by constructing an ensemble $\mathbf{E}_{t=0} \in R^{n \times N}$ with the flow variables set to zero and the parameters are sampled from a Gaussian distribution based on the initial parameter mean and variance. $\mathbf{E}_{t=0}$ is integrated to three different times: $T = 0.5$ s, $t = 0.75$ s, and $t = 1.0$ s. The integration forward in time from quiescent flow is done in an attempt to maintain full connectivity between the parameter values and its flow variables. A total of eight different prior settings are integrated forward in time, resulting in a total of 24 different initial priors. In order to

Table 4.2: Initial prior set-up for twin problem; mean and variance of the parameters, (e^{-2})

#	$F_x [N=m^2]$	$F_y [N=m^2]$	$x_o [m]$	$y_o [m]$		$y_{offset} [m]$	N	seed
1a	3000 (500)	-2000 (500)	0.001 ($1e^{-8}$)	0.001 ($1e^{-8}$)	$1e^6 (1e^6)$	$0 (1e^{-6})$	60	1
1b	3000 (500)	-2000 (500)	0.001 ($1e^{-8}$)	0.001 ($1e^{-8}$)	$1e^6 (1e^6)$	$0 (1e^{-6})$	80	1
2a	2800 (500)	-1800 (500)	0 ($1e^{-8}$)	0.001 ($2e^{-8}$)	$2e^6 (1e^6)$	$1e^{-4} (1e^{-6})$	60	1
2b	2800 (500)	-1800 (500)	0 ($1e^{-8}$)	0.001 ($2e^{-8}$)	$2e^6 (1e^6)$	$1e^{-4} (1e^{-6})$	80	1
3a	3500 (500)	-1500 (500)	-0.002 ($1e^{-8}$)	0.001 ($2e^{-8}$)	$2e^6 (1e^6)$	$1e^{-4} (1e^{-6})$	60	1
3b	3500 (500)	-1500 (500)	-0.002 ($1e^{-8}$)	0.001 ($2e^{-8}$)	$2e^6 (1e^6)$	$1e^{-4} (1e^{-6})$	80	1
4	3500 (500)	-1500 (500)	-0.002 ($1e^{-8}$)	0.001 ($2e^{-8}$)	$2e^6 (1e^6)$	$1e^{-4} (1e^{-6})$	60	2
5	3500 (50)	-1500 (50)	-0.002 ($1e^{-12}$)	0.001 ($2e^{-12}$)	$2e^6 (1e^3)$	$1e^{-12} (1e^{-6})$	60	1
6	2800 (00)	-1800 (500)	0 ($1e^{-8}$)	0.001 ($2e^{-8}$)	$2e^6 (1e^6)$	$1e^{-4} (1e^{-6})$	60	2

study the sensitivity of the EnKF solution, all priors are different from each other in certain aspects. Priors 1 through 3 have been set up for both an ensemble size of $N = 60$ and $N = 80$, in order to determine the influence of ensemble size of the solution. Prior 1a/b are set up with parameter means equal to those used to create the true solution. The results of the EnKF using initial prior 1a/b can show how much the non-multivariate Gaussianity included in the ensemble introduced by the non-linear dynamics will influence the parameter estimation of the EnKF. Prior 2a/b have been set up with the parameter means set to different values from the true solution. The initial parameter variance is selected in an attempt so that the true solution can be constructed using interpolation of the ensemble members. Prior 3a/b have been set up with again a different mean for the parameters, and the same variance. Prior 4, uses the same mean as for prior 2 but now with a different random seed initialization to create the initial ensemble and prior 5 uses a very low variance for the parameters, not allowing for the interpolation between ensemble member to create the updated ensemble. Table 4.2 gives a full overview of the settings used to create the initial priors. The values in between brackets denote the variance used for the given parameter.

4.1.3 Univariate statistics of initial priors

One of the things to ensure the proper working of the EnKF, especially in steady state, is the necessity of close to multivariate Gaussian initial priors. Due to the fact that it is difficult to study whether this Gaussianity is met for the full state space of n dimensions, this section will be restricted to univariate statistics of the initial priors. Many statistical moments can be calculated from the initial prior ensemble $\mathbf{E}_{T_0}^f$. However, this study will restrict itself to the calculation of the mean, variance, skewness and kurtosis. The skewness of a single ensemble member Ej of $\mathbf{E}_{T_0}^f$ can be calculated as:

$$sk = \frac{\sqrt{N(N-1)}}{N-1} \frac{\frac{1}{N} \sum_{i=1}^N (Ej_i - \bar{j})^3}{\left(\sqrt{\frac{1}{N} \sum_{i=1}^N (Ej_i - \bar{j})^2} \right)^3}; \quad (4.2)$$

where \bar{j} is the mean of the ensemble member Ej . The kurtosis of single ensemble member can be calculated as:

$$k = \frac{N-1}{(N-2)(N-3)} \left((N+1) \frac{\frac{1}{N} \sum_{i=1}^N (Ej_i - \bar{j})^4}{\left(\frac{1}{N} \sum_{i=1}^N (Ej_i - \bar{j})^2 \right)^2} - 3(N-1) \right) + 3: \quad (4.3)$$

Using (4.2) and (4.3) an expected value for a perfect Gaussian pdf is zero for the skewness and three for the kurtosis. The finite number of ensemble members already allow the calculated values to perturb from these expected values a little. High values for the skewness and kurtosis can be contributed to the non-linear dynamics of the forecast model. Figures 4.2 and 4.3 show an example of the calculated field statistics. Both figures show the univariate statistics of flow variables inside the PIV window, however the contour range is set to the maximum and minimum values found in the whole flow domain. Figure 4.2 show the statistics of flow variables in the x-direction, and figure 4.3 shows the statistics of flow variables in the y-direction. This can be done for all different initial priors and the full overview of these statistics is given in B.

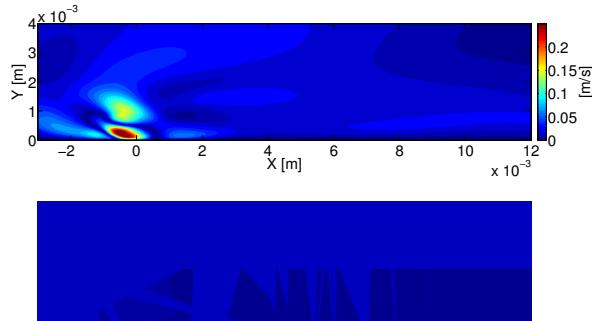


Figure 4.1: $|\mathbf{E}_{To}^f - \mathbf{Hx}_{To}|$ for prior 2b at $To = 0.5$ s.
top; x-velocity. bottom; y-velocity

ference for the prior case 2b at $To = 0.5$ s. This difference is of similar order of magnitude as the perturbations contained in the ensemble, as is shown in the top right graph of figures 4.2 and 4.3. This could mean that after an update of the EnKF the parameters mean contained in \mathbf{E}^a would be estimated incorrectly by an order equal to the original variance used to create

The top left graph shows the mean of the flow variables in \mathbf{E}_{To}^f . Information about the loss of connectivity between the mean of the flow variables in the observational domain (\mathbf{Hx}_{To}) and the mean of the parameters can be obtained from this statistics by comparing \mathbf{Hx}_{To} to the flow field of a single forecast model integration with parameters equal to the input mean of parameters in \mathbf{E}_{To}^f . If the forecast model were to be linear, the difference between \mathbf{Hx}_{To} and \mathbf{E}_{To}^f would be zero. Figure 4.1 shows this dif-

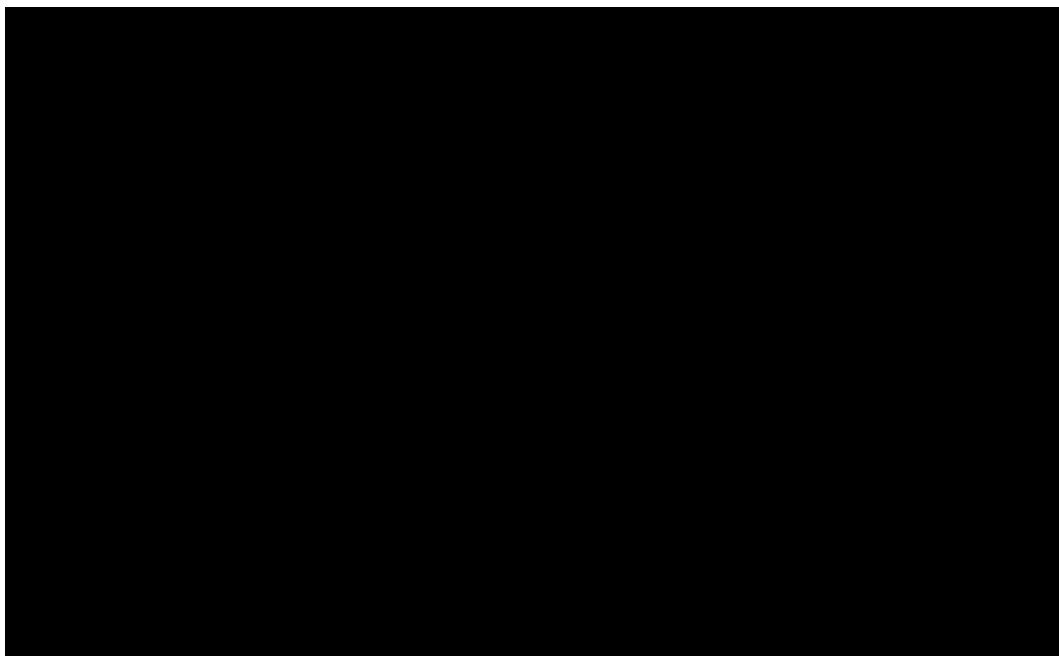


Figure 4.2: Univariate statistics of prior 2b x-velocity, $To = 50$; σ^2 (top right), sk (bottom left), and k (bottom right)

would be lost. In such a case it would be difficult, or nearly impossible to identify the true parameters. This reasoning only holds if the body force parameterization is assumed to be a good representation of the true force field and the initial parameter spread used to create the prior encapsulates the true solution, which is the case for the twin problem but not necessarily for the problem using PIV data.

Furthermore, the bottom two graphs of figures 4.2 and 4.3 show the skewness (left) and the kurtosis (right) of the flow variables on E_{To}^f . These two quantities can give a measure of non-Gaussianity in E_{To}^f . However, it should be noted that even if the skewness is equal to zero or the kurtosis equal to three, this does not guarantee a Gaussian distribution. E.g. if the skewness is equal to zero this only shows the mean is equal to the median, while both tails of the pdf can still have significant perturbations around the tails of a Gaussian distribution. The kurtosis can be interpreted as the heaviness of the tails of a pdf.

Comparing the statistics for other initial priors with each other, it can be noted that the integration of the forecast model to values of $t = 0.75$ s and $t = 1.0$ s show a significant reduction of skewness and kurtosis inside the PIV window. This could be interpreted as that not all ensemble members included in $E_{To=0.5}^f$ have reached a steady state solution in the observational domain. However, the maximum and minimum for skewness and kurtosis do not reduce with longer integration times, signifying the non-linear dynamics of the forecast model mainly present in the far field.

Inspection of the far field statistics (not shown in any figure) show high values of variance. This is most likely due to difference in the start-up vortex with different sets of parameters.



Figure 4.3: Univariate statistics of prior 2b y-velocity, $To = 50$; μ (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

Unless these perturbations are multi-variate Gaussian distributed with respect to values inside the observational domain, these flow variables can take on highly non-physical values, after the first update of the EnKF.

Finally, these statistics can also give an indication how well the ensemble can calculate the variance as perturbation around the mean. This can be done by comparing the distribution and magnitude of the variance for initial priors with the same parameter mean and variance but different ensemble sizes, or seed initialization. The comparison of variance distribution between prior with equal parameter mean and variance, show a large dependence on the random seed initial. Therefore, it can be concluded that the ensemble size is too small to accurately capture the perturbation around the mean for the initial prior.

4.1.4 Bivariate statistics of initial priors

Due to the high number of variables contained in the problem, only a limited study of the bivariate statistics will be given in this section. The visualization of the bivariate statistics is given by plotting the ensemble members of two variables vs one another. If the model were to behave linear, the points would form a straight line, with the highest concentration of points somewhere in the center of the line. The offset of the ensemble member from this line signify the non-linearity in the model and therefore also the non-Gaussianity.

One interesting result of the bivariate statistics is the probability distribution between variables in the far field with high variance and variable inside to observable domain. Non-

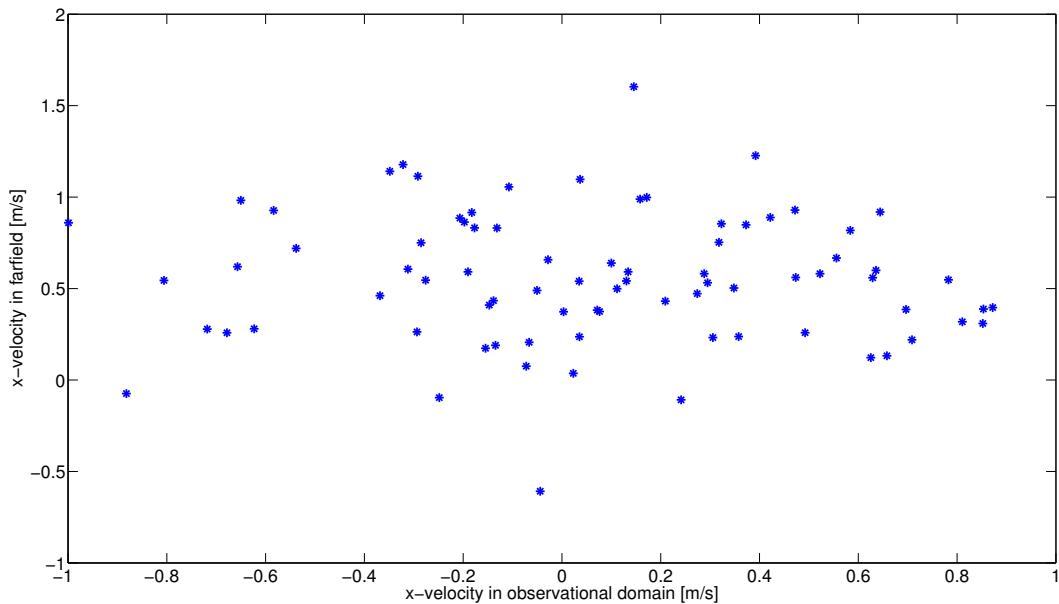


Figure 4.4: Ensemble members of initial prior 2b, $T_o = 1.0\text{s}$ of a far field flow variable vs a flow variable in the observational domain

Gaussian bivariate distributions between these variable can predict high nonphysical flow velocities in the far field due to very poor initial updates of the ensemble. Figure 4.4 shows the bivariate distribution between flow variable in the observational domain and one in the far field. The flow variables have been selected based on highest calculated variance. Figure 4.4 shows that the bivariate distribution shows very little signs of a Gaussian distribution at all. Therefore, it is expected that after the initial update, which is based on the flow variables in the observational domain, the far field flow parameter can take on high nonphysical values.

A second interesting bivariate distribution would be that between flow variables in the observable domain and parameters. The non-Gaussianity can measure an initial loss of connectivity between parameter and flow variable after the first update step. Figure 4.5 shows the bivariate distributions of a flow variable with high variance in the observable domain with the initial parameter values. Most distribution plots show very little signs of Gaussianity. Only the flow variable y_{offset} shows an inverse Gaussian distribution. This is only the bivariate distribution of one single flow variable. The analysis uses all the flow variables inside the observational domain, and the system could therefore still look worse or better overall in terms of Gaussianity.

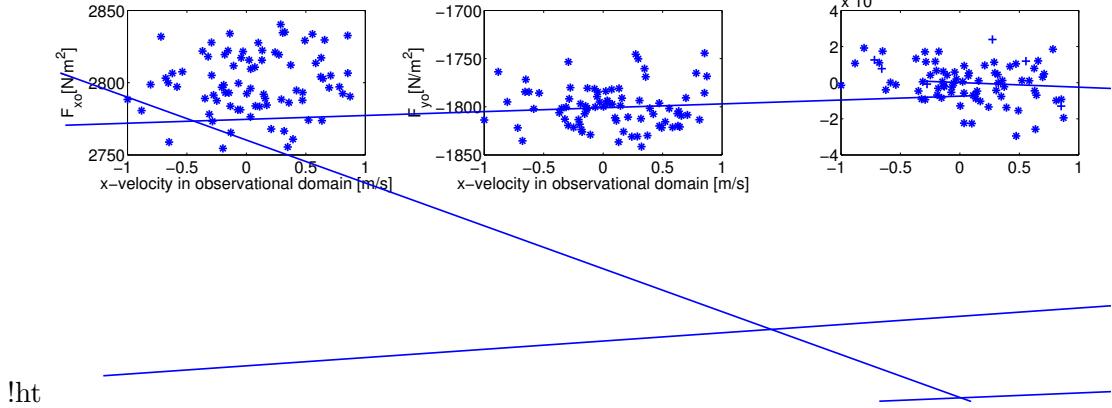


Figure 4.5: Ensemble members of initial prior 2b, $T_o = 1.0\text{s}$ of a far field flow variable vs initial parameters

4.1.5 Twin problem filter settings

Numerous runs of the EnKF have been done using slight variation in the initial settings in an attempt to further understand the problem. A total of four different filter setting are used.

Table 4.3: Overview of twin problem EnKF settings

	Version setting #			
	v1	v2	v3	v4
Prior	1, 2(a/b), 3(a/b), 4, 5, 6	2(a/b), 3(a/b)	2(a/b), 3(a/b)	2(a/b), 3(a/b)
T_o [s]	0.5, 0.75, 1.0	1.0	1.0	0.5, 0.75, 1.0
dT [ms]	0.1	0.05	0.2	0.1
filter	EnKF/DEnKF	EnKF/DEnKF	EnKF/DEnKF	EnKF/DEnKF
inflation	multiplicative 3 %	multiplicative 3 %	multiplicative 3 %	multiplicative 3 %
bias force	-	-	-	y_o

All settings are set up in a way to be able to see the effects of type of filter and ensemble size on the final results. Versions 2 and 3 are setup to study the sensitivity to different integration times between updates (dT), while v4 is set up to study the ability of the (D)EnKF to capture the body force even if the force parameterization is biased. This bias force is created by setting the parameter y_o to a constant value of 2 mm. For all settings, inflation of 3 percent is used in the form of multiplicative inflation.

4.2 Twin Problem Results

A standard approach to evaluate the convergence of the parameters in a steady state EnKF problem is to run the filter until the parameters and variance stop changing. However, due to the addition of bias through the *sponge-layer* in the forecast model to suppress high

nonphysical flow velocities, it is not always possible to run the data assimilation for large periods of time. This bias can cause the parameters to diverge away from the true value, sometimes causing the forecast model to force quits due to extremely high sponge ratios. Therefore, the converged parameters are taken those parameters included in the ensemble creating the lowest root mean square error in the observable domain ($RMSE_o$) defined as:

$$RMSE_o = \sqrt{\frac{\sum_{i=1}^m (\mathbf{d}_i - (\mathbf{Hx})_i)^2}{m}}. \quad (4.4)$$

The usage of lowest $RMSE_o$ to determine the converged parameters is only a fix to determine the 'so far' best estimation before very large bias error is introduced into the system.

An additional quantity for the evaluation of the parameter estimation is proposed in the form of the force field root mean square error ($RMSE_f$). This quantity can only be calculated when the true parameter values are known.

$$RMSE_f = \sqrt{\frac{\sum_{i=1}^m (f_x^{true} - f_x^{estimate})^2}{m}} + \sqrt{\frac{\sum_{i=1}^m (f_y^{true} - f_y^{estimate})^2}{m}}. \quad (4.5)$$

f^{true} and $f^{estimate}$ are the force field created by the set of true parameters and estimated parameters, respectively, in the observational domain.

The (D)EnKF has the problem of filter divergence. Filter divergence occurs when the analysis preformed moves the updated ensemble away from the true solution while still decreasing its covariance, therefore also called covariance divergence. This reduction in covariance gives a false impression that the filter is working correctly. The divergence can be due to many factors such as: Limited ensemble size, method to create prior, non-Gaussianity, high levels of bias error etc. It should be noted that the filter can still converge to the true solution even if it diverged in previous update steps. Another type of divergence can also occur but is less likely called catastrophic divergence (Harlim and Majda, 2010) and is defined as the ensemble reaching values of machine infinity in finite time.

For the specific problem of the parameter estimation, two types of covariance divergence are of interest. First is the divergence of the observable flow field away from the true flow field while reducing the average variance in the observable domain, i.e. $RMSE_o$ increases while average variance decreases. From now on this type will be referred to as *standard filter divergence*. Average variance in the observable domain (\mathcal{S}) is defined by (4.6).

$$\mathcal{S} = \frac{\sum_{i=1}^m \sigma_i^2}{m} \quad (4.6)$$

The second type of divergence is parameter divergence, where the parameters move away from the true parameters ($RMSE_f$ increases, while the $RMSE_o$ decreases). This type will be referred to as *parameter divergence*. Both type of filter divergence can cause the filter to move away from the global minimum, finding a local minimum instead. It is expected that the current method of the SSS-EnKF proposed, will always converge to atleast a local minimum, even if previous update steps diverged. In the current case this might not be always true because the addition of the *sponge layer* causes the response surface to potentially change drastically, and does not allow the time for a full convergence to the minimum in some cases.

4.2.1 Steady State EnKF

Table 4.4: Results SSEnKF using prior 1a/b; Final parameter estimation with corresponding $RMSE_o$ and $RMSE_f$

	F_{xo} [$\frac{N}{m^2}$]	F_{yo} [$\frac{N}{m^2}$]	x_o [mm]	y_o [mm]	β [-]	y_{offset} [mm]	$RMSE_o$ [m/s]	$RMSE_f$ [N/m]
Truth	3000.00	-2000.00	1.00	1.00	1.00E+06	0	-	-
$T_o = 1.0$ s	2807.57	-1776.37	1.04	1.13	9.97E+05	-3.15E-02	0.012619	35.007
$N = 60$	2.01	0.85	2.12E-10	2.24E-10	3.28E+04	6.22E-11	0.012619	35.007
$T_o = 1.0$ s	3200.43	-2023.41	1.02	0.99	9.91E+05	-1.13E-02	0.012812	22.464
$N = 80$	5.06	1.36	2.08E-10	6.28E-10	5.49E+04	1.23E-10	0.012812	22.464
$T_o = 0.5$ s	2850.52	-1996.74	1.04	1.10	1.00E+06	-3.86E-02	0.019617	20.977
$N = 60$	1.27	0.30	2.21E-10	2.10E-10	3.81E+03	9.62E-11	0.019617	20.977
$T_o = 0.5$ s	3264.42	-2164.12	0.97	0.93	9.86E+05	1.98E-02	0.019573	22.234
$N = 80$	2.82	0.84	2.33E-10	4.45E-10	1.45E+04	1.95E-10	0.019573	22.234
$T_o = 0.75$ s	2902.76	-1840.41	1.09	1.10	9.85E+05	-3.76E-02	0.015830	22.076
$N = 60$	2.47	0.94	2.16E-10	1.99E-10	2.87E+04	5.84E-11	0.015830	22.076
$T_o = 0.75$ s	2885.21	-1986.21	1.01	1.03	9.93E+05	-1.40E-02	0.013751	8.411
$N = 80$	2.47	0.84	1.98E-10	3.23E-10	4.69E+04	9.94E-11	0.013751	8.411

As has been discussed in section 3.1, the non-linearities introduced in the initial priors are most likely to let the steady state EnKF diverge during the update step and converge to parameter values away from the true parameter values. This is demonstrated in this section by using the EnKF in steady state with initial prior 1a/b, a total of six different runs. This initial prior has parameter means equal to that of the true state, and if the system was linear it is expected to converge towards the true parameter mean. Table 4.4 shows the final parameter estimation for the SSEnKF runs. The final parameter means are offset by values larger than the corresponding variance. As can be seen in figure 4.6, the incorrect parameter estimation is due to high levels of parameter divergence, which shows the values for $RMSE_o$, $RMSE_f$ and S for a particular run. Looking at the low values for $RMSE_o$, the filter seems to be able to reconstruct the flow field well given the initial ensemble, however, while $RMSE_o$ and S decrease with every iteration, $RMSE_f$ increases, indicating parameter divergence. This parameter divergence was expected due to the high non-linearities included in the initial prior, proving the necessity for an alternative method such as the SSS-EnKF method. Figure 4.7 shows the parameter estimation development for the same run. Showing the fast initial decrease in variance while moving away from the true parameter values. A full overview of the results is given in Appendix C.

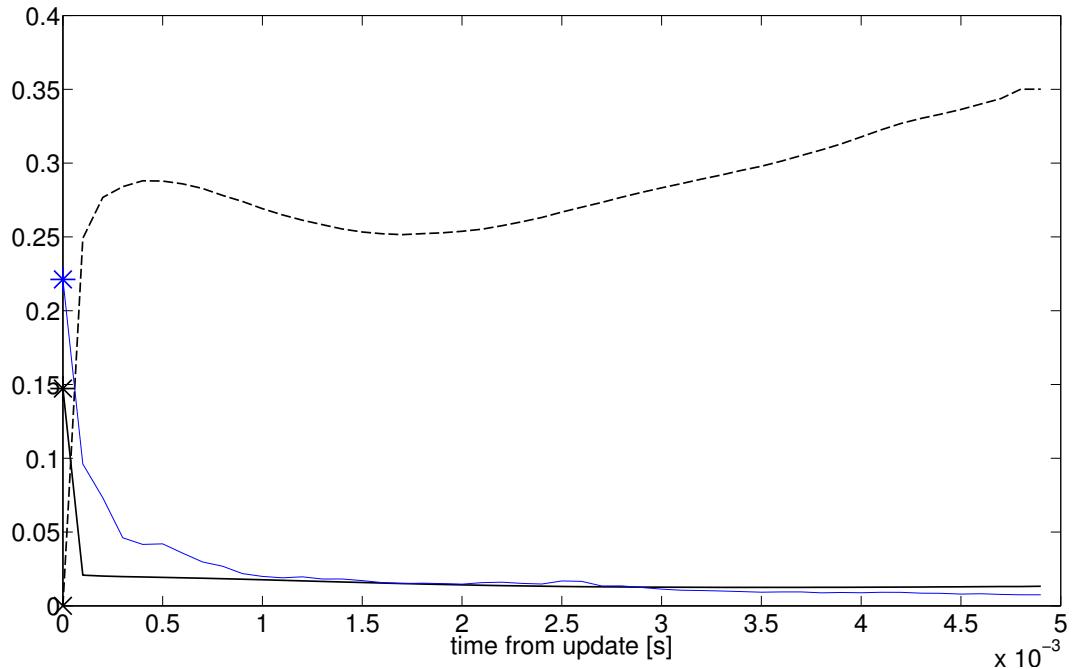


Figure 4.6: SSEnKF results prior 1a, $T_o = 1.0$ s; $\frac{RMSE_o}{100} [m=s]$ (* black solid line), $\frac{RMSE_f}{100} [N=m^2]$ (x black dashed line), $S [m=s]$ (* blue line)

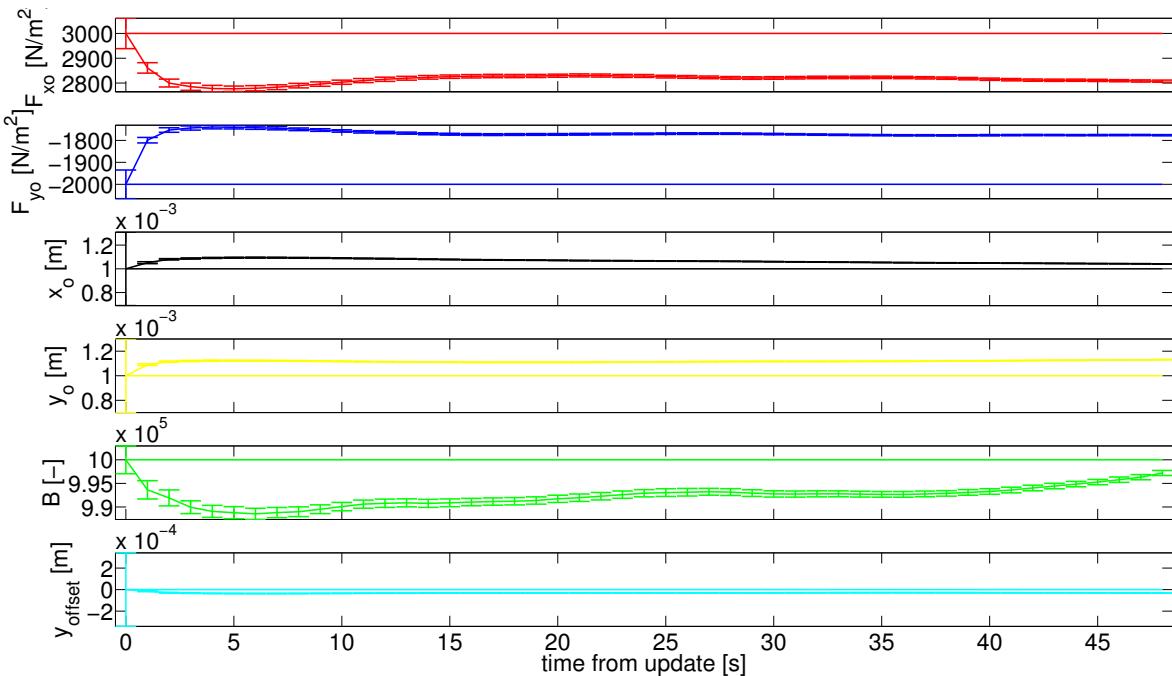


Figure 4.7: Static EnKF results prior 1b; parameter mean and $\pm 3\sigma$ error bars. Lines of constant values are equal to true parameter value.

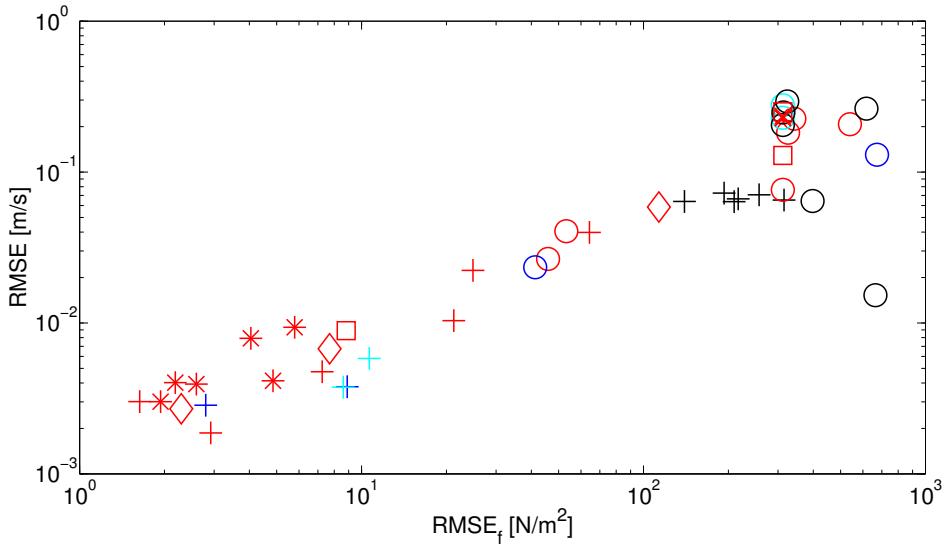


Figure 4.8: Plot of $RMSE_o$ vs $RMSE_f$ for the EnKF. Different colors represent different Iteration settings; red (v1), blue (v2), cyan (v3), black (v4). The different symbols represent the different initial priors; * (prior 1a/b), + (prior 2a/b), o (prior 3a/b), □ (prior 4), x (prior 5), ◊ (prior 6)

4.2.2 Semi Steady State Filtering

This section will show the ability of the SSS(D)EnKF to estimate the mean of the true parameters with reasonable accuracy. A discussion about the consistency of the results is given at the end of the section. The results from the twin problem will determine the best settings for the data assimilation with PIV data, within the range of parameter settings the study is conducted in. A complete overview of the results is given in Appendix D.

RMSE_o vs RMSE_f

As discussed above the minimum value for $RMSE_o$ will determine the optimal estimation of the filter. $RMSE_o$ together with $RMSE_f$, and \mathcal{S} will be used as indication of the correct working of the filter. To capture the best parameter estimation during a single run, it is essential that the $RMSE_o$ is a good indication of the correct working of the filter. The calculated values for minimum $RMSE_o$ and its corresponding $RMSE_f$ can be plotted against each other to give an indication of whether this is true.

Figures 4.8 and 4.9 indicate that $RMSE_o$ can be used as a good estimation of the correct working of the filter when $RMSE_f$ is unknown, especially in the region where $RMSE_o$ is small. The logarithmic axis are used to be able to show the wide range of results. The relationship does not seem to be exact, therefore, results should be examined closely to ensure no outlier is found.

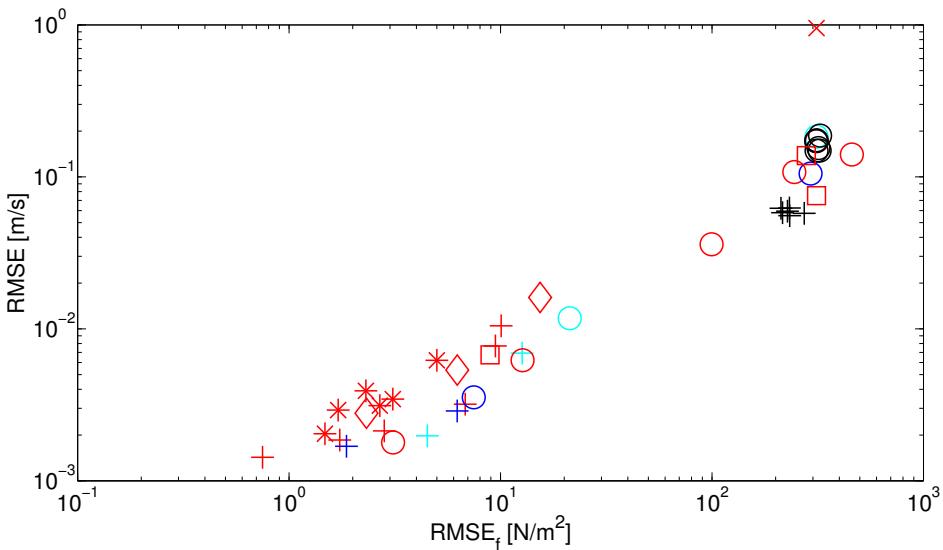


Figure 4.9: Plot of $RMSE_o$ vs $RMSE_f$ for the DEnKF. Different colors represent different DEnKF version settings; red (v1), blue (v2), cyan (v3), black (v4). The different symbols represent the different initial priors; * (prior 1a/b), + (prior 2a/b), o (prior 3a/b), □ (prior 4), x (prior 5), ◊ (prior 6)

Filter Divergence

Almost all of the data assimilation runs conducted show parameter divergence in at least the first couple of update steps, i.e. the $RMSE_o$ is reduced while the $RMSE_f$ increases. This happens even with prior 1, with the initial prior mean equal to the true mean, shown in figure 4.12. This parameter divergence using prior 1 signifies the non-linearity between the observable flow variable and the force parameters. Only for some priors no parameter divergence is observed, most likely due to high values for initial $RMSE_f$.

Standard filter divergence is observed when low values of sponge ratios occur, which therefore seems to be the driving factor of standard divergence in this problem. This also shows the high dependence on the initial prior mean, i.e. if the initial prior contains more ensemble members far away from the true, this is more likely to trigger the sponge layer with very low ratios. This can clearly be seen in prior 5, shown in figure 4.10, where initial low values of sponge ratio cause the filter to diverge dramatically.

All prior show a fast reduction $RMSE_o$ and \mathcal{S} in the first couple of update steps. For the DEnKF, \mathcal{S} is increased again after this initial period of reduction, while the EnKF continues to keep \mathcal{S} at low levels. The increase in total variance could help reduce the loss in connectivity between the observable flow variable and the estimated parameters.

The initial rapid decrease and then increase of $RMSE_o$, can be explained by the fact that the initial prior includes ensemble member well suited to reproduce the observations, however, due to the sub optimal update (due to non-Gaussianity in initial prior see Section 4.1.4), incorrect parameter values are created. After some integration forward in time these incorrect

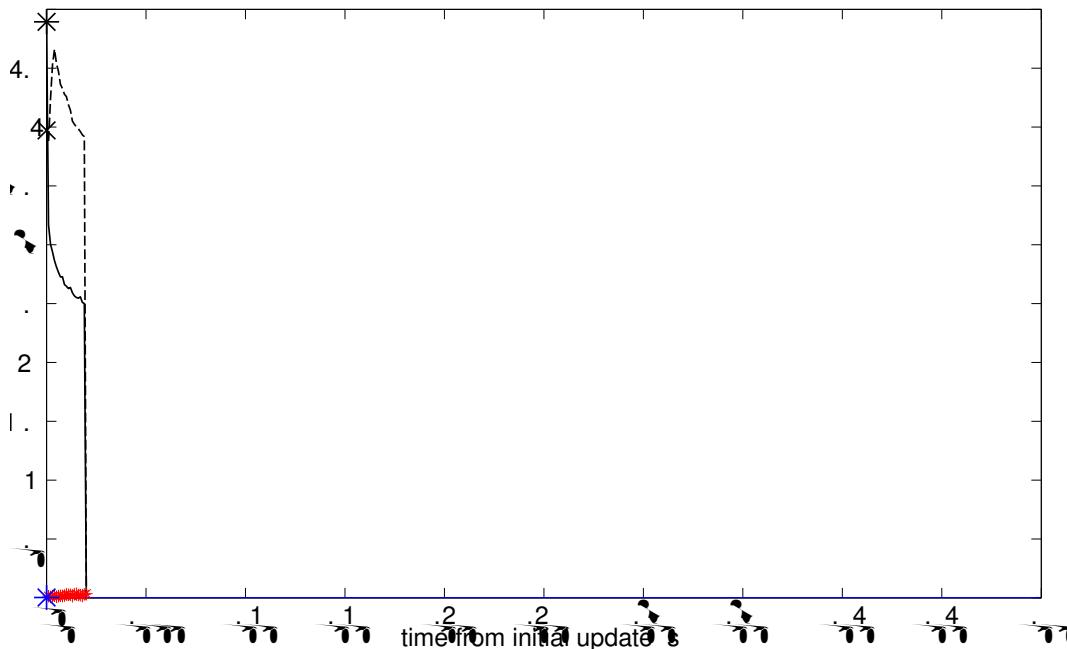


Figure 4.10: SSS-EnKF results prior 5v1, $N = 60$, $T_o = 1.0$ s; $RMSE_o \cdot 100[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\Sigma_{10}[m=s]$ (* blue line), Sponge ratio [-] (red stars)

parameters cause all ensemble members to move away from the observations, increasing the $RMSE_o$ again. This rapid decrease of $RMSE_o$ can be called an initialization period and values of $RMSE_o$ during this period will be ignored when determining the best parameter estimation during the filtering process. The initialization period can clearly be seen in figure 4.11. If the parameters created during this initialization period do not differ much from the true parameter, it is expected that the $RMSE_o$ will continue to decrease, as could be the case for runs using prior 1, with parameter mean equal to the true parameter values. This further reduction in $RMSE_o$ for prior 1 can be seen in figure 4.12.

Looking at the results for filter settings v4, for which the bias forecast is used, it becomes clear that consistently finding an minimum is difficult for the (D)EnKF. This can especially become clear when looking at figure 4.11. Especially high values for sponge ratio cause most of the runs to diverge and finally force quit. This could be explain because the new minimum, with one force parameter fixed, may lay farther away from the initial prior, causing more significant changes in the flow field after the first couple of updates.

A large overview of results including $RMSE$ values is given in Appendix D.2.

Initial prior with equal initial parameter covariance

The four characteristics of the initial priors excluding initial parameter covariance are: (1)initial parameter mean, (2) random seed initialization, (3) ensemble size (N), and (4) initial integration time (T_o).

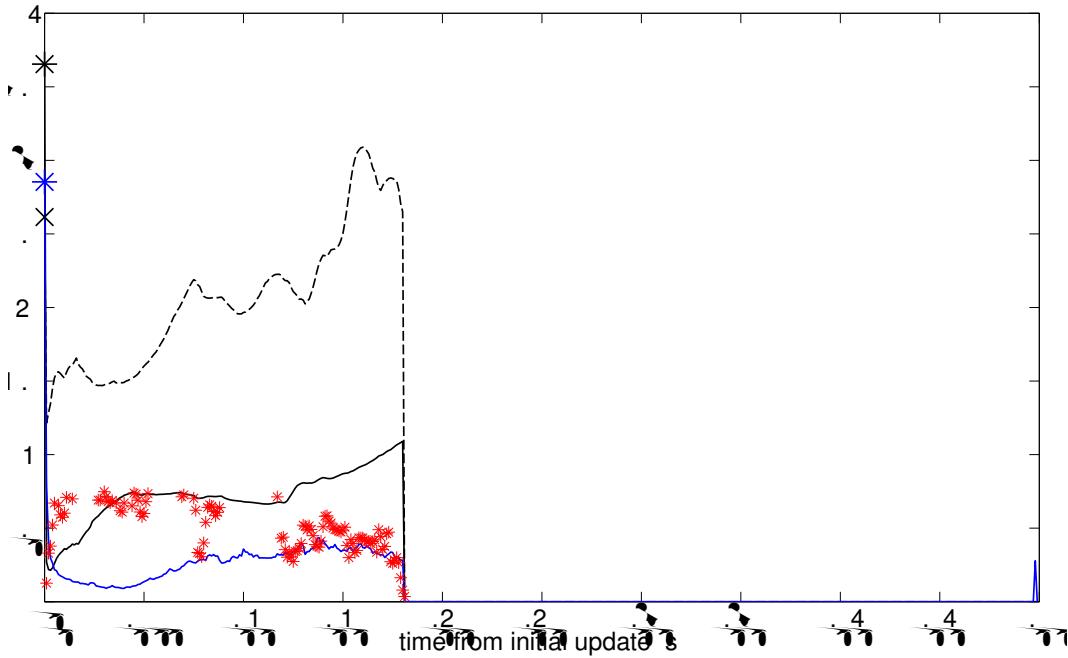


Figure 4.11: SSS-EnKF results prior 2v4, $N = 80$, $T_o = 0.5$ s; $RMSE_o \cdot 100[m=s]$ (* black solid line), $\frac{RMSE_t}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{10}[m=s]$ (* blue line), Sponge ratio [-] (red stars)

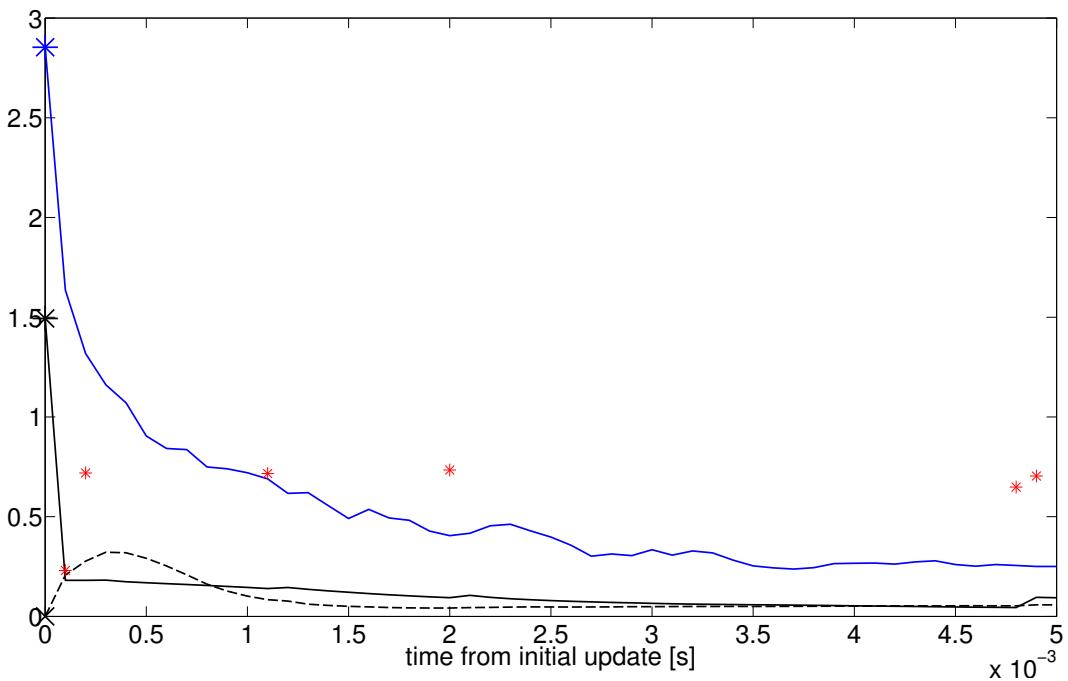


Figure 4.12: SSS-EnKF results prior 1v1, $N = 80$, $T_o = 1.0$ s; $RMSE_o \cdot 100[m=s]$ (* black solid line), $\frac{RMSE_t}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{10}[m=s]$ (* blue line), Sponge ratio [-] (red stars)

The influence of the ensemble size changes due to insufficient *Ensemble convergence*. *Ensemble convergence* occurs when the ensemble size is sufficiently large to capture the dispersion around the mean path. As has been discussed in section 4.1, the ensemble size is not sufficiently large for the initial prior to capture the initial anomalies. Therefore, the initial priors used with different ensemble sizes can be categorized as a different initial priors similar to priors with a different seed initialization.

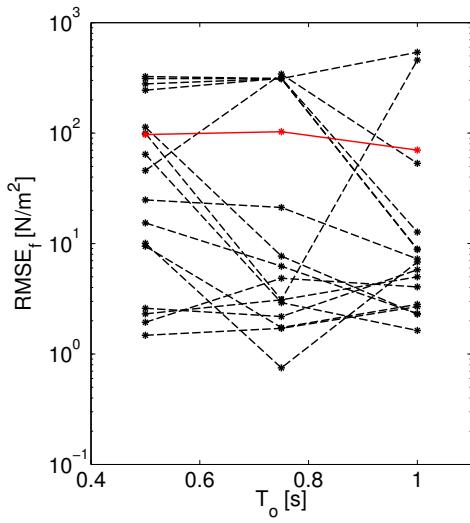


Figure 4.13: RMSE_f for different initial integration times for initial priors 1a/b, 2a/b, and 3a/b. Red line denotes the average RMSE_f for a single T_o

RMSE_f lets us select $T_o = 1.0\text{s}$ for the data assimilation with PIV data. The fact that longer initial integration times yield better results could be partially due to the fact that these more of the ensemble members in the initial prior are close to a fully steady state solution and might therefore look more linear inside and close to the observational domain.

The high dependence on initial prior mean and random seed initialization shown by table 4.5 show that the initial prior should be chosen to the best knowledge to be as close to the true solution as possible.

Table 4.5 shows the results in terms of RMSE_f for three different priors with equal initial parameter covariance. This table includes both ensemble sizes used as well as the different random seed initialization used with an ensemble size of 60. This table shows that the results of the filter, regardless of whether the EnKF or DEnKF is used, is highly dependent on the initial prior. The overall best results are generated using Prior 1, which is expected as it has the initial parameter mean equal to the true mean.

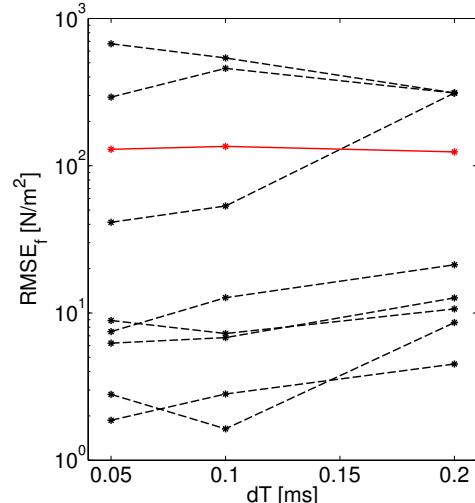
Figure 4.13 shows the sensitivity of the solution with regards to T_o . The figure shows the change in RMSE_f for the initial priors with different T_o . The red line indicates the average RMSE_f for different T_o . The slight average decrease in

Table 4.5: Overview of $RMSE_f$ results for first three prior with settings v1

	EnKF			DEnKF		
	N60	N80	N60 (different seed)	N60	N80	N60 (different seed)
Run1 v1 $T_o = 1.0\text{s}$	4.04867	5.789693	-	4.998477	2.686038	-
Run1 v1 $T_o = 0.5\text{s}$	1.937466	2.594744	-	2.308943	1.477406	-
Run1 v1 $T_o = 0.75\text{s}$	4.852966	2.184703	-	3.093261	1.707207	-
Run2 v1 $T_o = 1.0\text{s}$	7.125184	1.580311	8.705444	6.693855	2.792824	8.634252
Run2 v1 $T_o = 0.5\text{s}$	24.568347	64.098989	179.5478	10.130504	9.188797	278.316118
Run2 v1 $T_o = 0.75\text{s}$	20.947549	2.889824	300.939745	0.746923	1.672794	84.61234
Run3 v1 $T_o = 1.0\text{s}$	52.632515	533.971447	2.294111	12.623451	457.74524	2.322017
Run3 v1 $T_o = 0.5\text{s}$	45.183572	324.724818	111.683055	240.06895	97.81616	15.52464
Run3 v1 $T_o = 0.75\text{s}$	343.88384	83.671094	7.708306	313.58203	3.031682	5.99999

Integration time between updates

For the determination of the sensitivity of the solution with respect to the integration time between updates (dT), we can compare the results obtained from priors 2 and 3, in combination with the filter settings v1, v2, and v3. The $RMSE_f$ for these solutions is plotted versus the integration time and is shown in figure 4.14. When compared to figure 4.13 and table 4.5 it can be seen that the solution is far less sensitive to dT than to the initial prior.



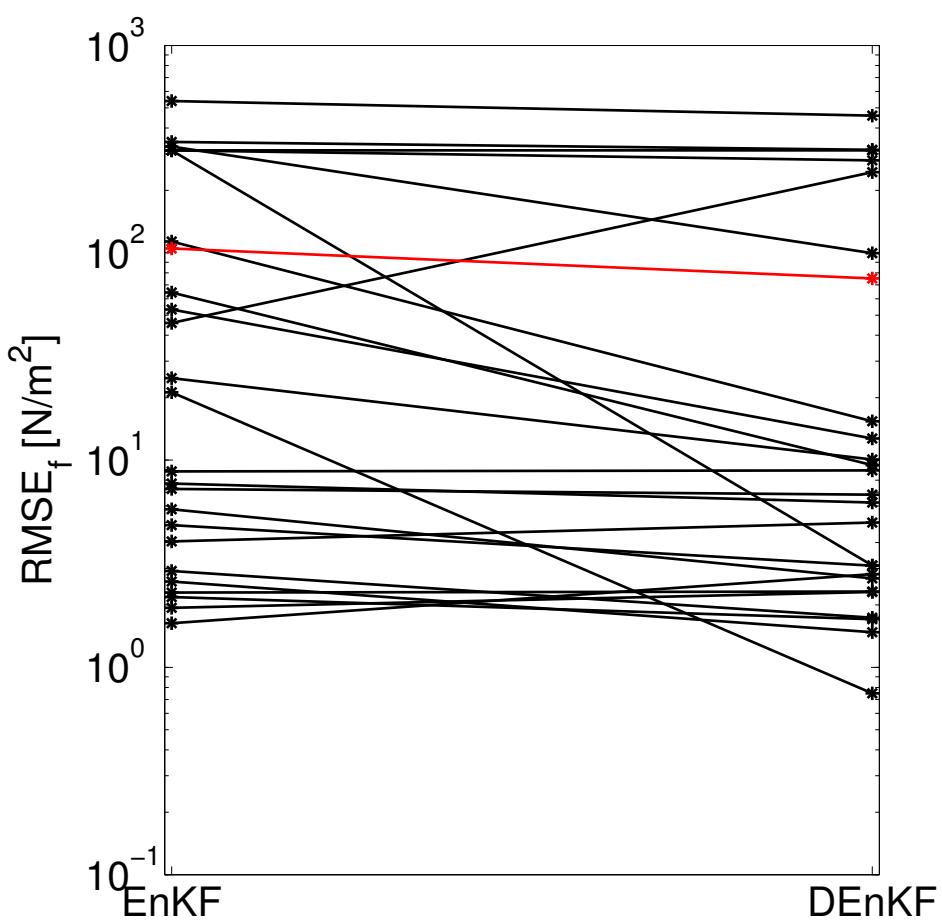
EnKF vs DEnKF

A comparison between the EnKF and the DEnKF is made by comparing the average of $RMSE_o$ and $RMSE_f$ for priors 1,2,3,4 and 6 for the filter settings v1. These runs are selected because they show some convergence and it eliminates most of the runs having very high values for $RMSE_o$ and $RMSE_f$. Table 4.6 shows the results of this averaging. It can be seen that the DEnKF outperforms the EnKF on average. However, figure 4.15 shows however that for some runs the EnKF still outperforms the DEnKF.

Figure 4.14: $RMSE_f$ for Priors 2a and 3a with different integration times between updates. Red line denoting the average of the runs.

Table 4.6: Average $RMSE_o$ and $RMSE_f$ of priors 1,2,3,4, and 6 for settings v1

	$RMSE_o [m=s]$	$RMSE_f [N=m^2]$
EnKF	0.055417	89.065175
DEnKF	0.031543	65.323912

**Figure 4.15:** Comparison EnKF with DEnKF through $RMSE_f$

Bias forecast

For the data assimilation with PIV data, it is important to know that the filter will be able to find a solution even if the body force field parameterization cannot exactly capture the true body force field. This additional bias in the forecast model is introduced using filter settings v4, where parameter y_o is fixed.

Table 4.7 gives an overview of some of the main results comparing the unbias forecast (v1) with the bias forecast (v4). The table includes the thrust in the x- and y-direction. Thrust is defined as sum of the force \mathbf{F} acting on each volume element in the fluid.

$$\mathbf{F} = \int_V \mathbf{f} dV \approx \mathbf{f}_i \Delta x_i \Delta y_i \quad (4.7)$$

From table 4.7, it can be seen that the filter has a very difficult time of consistently finding the optimal solution. The $RMSE_o$ and $RMSE_f$ have increased compared to the unbias forecast model. Also, the thrust calculations for v4 show a lot more results compared to the unbias forecast model. This could indicate multiple local minima. This table shows that the filter will still be able to find an approximation to body force field, even if the force field parameterization is sub-optimal, but also shows the high dependence on a good body force parameterization.

Table 4.7: Comparison results using unbias (v1) and bias (v4) forecast model, True thrust: $T_x = 9.6615$ mN, $T_y = -6.4410$ mN

	$RMSE_o$ [m/s]		$RMSE_f$ [N/m^2]		Tx [mN]		Ty [mN]	
	v1	v4	v1	v4	v1	v4	v1	v4
EnKF								
Prior2a $T_o = 1.0$	0.004753	0.070660	7.125184	257.651149	9.594057	5.950986	-6.344845	-4.438080
Prior2b $T_o = 1.0$	0.003012	0.065205	1.580311	314.999823	9.587875	0.835538	-6.419829	-0.615857
Prior2a $T_o = 0.5$	0.022300	0.063677	24.568347	209.277802	8.893971	6.686986	-6.036223	-4.572315
Prior2b $T_o = 0.5$	0.039801	0.066401	64.098989	217.633816	8.973406	6.900636	-5.509677	-5.186866
Prior2a $T_o = 0.75$	0.010355	0.072508	20.947549	192.945570	9.246991	7.639213	-5.924780	-5.070867
Prior2b $T_o = 0.75$	0.001867	0.063860	2.889824	138.691273	9.554835	9.143641	-6.372159	-7.047977
Prior3a $T_o = 1.0$	0.040553	0.015268	52.632515	664.764125	8.464730	30.026637	-6.084292	-20.747224
Prior3b $T_o = 1.0$	0.207523	0.293826	533.971447	323.262812	11.226887	3.607185	-60.049972	-1.599068
Prior3a $T_o = 0.5$	0.026560	0.064281	45.183572	405.651678	8.600225	9.979255	-5.518458	-4.338725
Prior3b $T_o = 0.5$	0.182317	0.205118	324.724818	312.024874	2.054722	0.000675	2.774787	0.001002
Prior3a $T_o = 0.75$	0.225693	0.248163	343.883838	313.534468	2.149835	-0.236246	-6.589707	-1.070156
Prior3b $T_o = 0.75$	0.076248	0.263169	83.671094	576.890477	7.315910	23.643758	-6.088595	-10.404322
DEnKF								
Prior2a $T_o = 1.0$	0.003192	0.062165	6.693855	212.43899	9.428300	7.928106	-6.309200	-5.888899
Prior2b $T_o = 1.0$	0.002131	0.062416	2.792824	231.54174	9.574122	7.272614	-6.368241	-5.328976
Prior2a $T_o = 0.5$	0.010472	0.057586	10.130504	274.33992	9.706522	4.912000	-6.426109	-3.615058
Prior2b $T_o = 0.5$	0.007717	0.055577	9.188797	234.90218	9.678292	6.686149	-6.350092	-4.618676
Prior2a $T_o = 0.75$	0.001428	0.059450	0.746923	228.07016	9.637648	7.443291	-6.429121	-5.600916
Prior2b $T_o = 0.75$	0.001856	0.058113	1.672794	215.88054	9.636508	7.074899	-6.418666	-5.760596
Prior3a $T_o = 1.0$	0.006215	0.171724	12.623451	310.270471	9.227268	0.533805	-6.243999	0.366590
Prior3b $T_o = 1.0$	0.140357	0.154811	457.74524	319.58075	9.889679	0.246162	-32.674926	2.886233
Prior3a $T_o = 0.5$	0.107581	0.148908	240.06895	322.28607	2.940730	4.452373	-2.757108	-1.204374
Prior3b $T_o = 0.5$	0.036023	0.174116	97.81616	1347.6445	7.355778	120.739288	-4.617651	-37.298858
Prior3a $T_o = 0.75$	0.172389	0.186952	313.58203	345.87434	1.117000	-1.942379	-0.629666	3.886130
Prior3b $T_o = 0.75$	0.001787	0.148742	3.031682	312.01778	9.551118	0.014515	-6.363651	-0.009230

Consistency of the method

The method of the SSS-EnKF is assumed to be able to find a the true minimum independent of the initial prior, assuming the filter is allowed to run for indefinite periods of time. The

addition of the *sponge layer*, removes this consistency of the method by introducing very high values of bias and finally causing the forecast model to force quit. If the filter could run for indefinite periods of time, all ensemble members are allowed to reach complete steady state, and the filter will be able to find a (local) true minimum. The only dependence of the results on the initial prior will be how long the filter needs to run before all ensemble member would reach complete steady state.

The current parameter mean estimations in the twin problem are assumed to be incorrect with regards to the theoretical EnKF in steady state due to mostly the non-steady state ensemble members. The non-steady state ensemble members also cause for an incorrect parameter error estimate. The current twin problem results show extremely low values for parameter error, not sufficiently large to capture the true parameter values. These results can be found in table D.1. If all ensemble member were allowed to reach complete steady state, the parameter error estimation would mostly be affected by the non-linearities caused by the forecast model, and would generate much better results.

Given the current forecast model used, the performance of the filter can therefore be determined by a relative constant $RMSE_o$ for longer periods of integration time. This allows for more ensemble members to reach steady state, and ensure a true minimum is found.

4.2.3 Optimal settings for PIV

The previous sections have shown some of the results of the sensitivity of the filter results to different filter settings. The range of parameter setting values studied is not sufficiently large to draw any definite conclusions about optimal settings. However, the best settings using the current range of parameter setting values recommended for the data assimilation with PIV data are given in table 4.8. For filter settings with low average sensitivity to the result, such as N and dT , the settings with lowest computational cost have been selected. This is not expected to cause significant additional bias error. The average results seemed not highly dependent on the filter type and the DEnKF is selected based on the slightly better average performance. The initial integration time selected is $T_o = 1.0\text{s}$ in an attempt to create mostly steady state ensemble members.

Table 4.8: Filter settings used for data assimilation with PIV

N	60
T_o	1.0 s
dT	0.05 ms
Filter	DEnKF

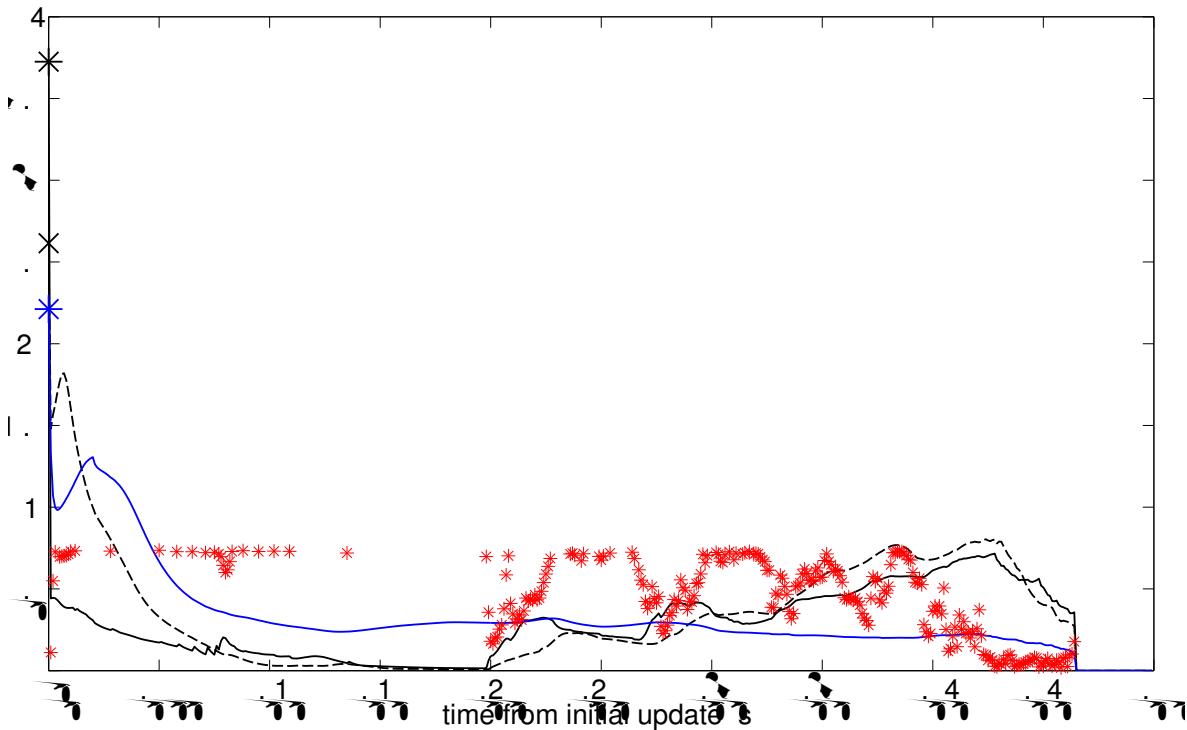


Figure 4.16: Semi steady State DEnKF results prior 2v1, $N = 60$, $T_o = 0.75$ s; $\frac{RMSE_o}{100} [m=s]$ (* black solid line), $\frac{RMSE_f}{100} [N=m^2]$ (x black dashed line), $\Sigma[m=s]$ (* blue line), Sponge ratio [-] (red *)

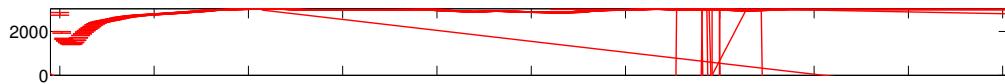


Figure 4.17: Semi steady State DEnKF parameter estimation development for prior 2v1, $N = 60$, $T_o = 0.75$ s; parameter mean and ± 3 error bars. Lines of constant values are equal to true parameter value.

Chapter 5

Semi Steady State Itering with PIV data

This chapter discusses the results obtained from data assimilation using the DEnKF in combination with PIV data of eight different quasi-steady flow fields generated by plasma actuator in quiescent flow. First, a short introduction of how initial priors are selected for the filtering with PIV data for the different scenarios is given. The last part of the chapter will discuss the results and compare thrust, a characteristic value of plasma actuator, to previous findings by [Kotsonis \(2012\)](#).

5.1 Initial Priors for PIV

As is shown in chapter 4, the results of the filter highly depend on the initial prior. The prior should therefore be picked using our best knowledge. The method used in determining the initial parameter values was to first obtain some characteristic values from [Shyy et al. \(2002\)](#). These values are scaled up or down to cover the range of cases to be studied. As is known from previous results ([Kotsonis, 2012](#)), the body force field is expected to increase in both size and intensity when the voltage is increased. In the case of increasing actuator frequency (f_{ac}), the body force field is expected to increase in intensity. Table 5.1 shows the settings used to set-up the priors. The variance used is equal to that of the twin problem. An ensemble size of $N = 60$ is used, and all initial priors are integrated to $T_o = 1.0$ s.

To increase the chances of a successful filter run, a single prior is used for different cases. The determination of which initial prior to re-use for which case is based on the comparison of the maximum mean flow velocity observed in the initial prior and that observed in the PIV data. Table 5.2 shows which priors are used with which case.

Table 5.1: Initial prior set-up; mean and variance of the parameters, (e^2)

#	$F_x [N=m^2]$	$F_y [N=m^2]$	$x_o [m]$	$y_o [m]$	$y_{offset} [m]$
1	2500 (500)	-1500 (500)	$1e-3 (1e^{-8})$	$1e-3 (1e^{-8})$	$1e^6 (1e^6)$
2	4000 (500)	-4000 (500)	$3e^{-4} (1e^{-8})$	$2.6e^{-4} (1e^{-8})$	$1.4e^6 (1e^6)$
3	1500 (500)	-1500 (500)	$1.0e^{-3} (1e^{-8})$	$1.2e^{-3} (1e^{-8})$	$2e^6 (1e^6)$
4	800 (500)	-800 (500)	$1.1e^{-3} (1e^{-8})$	$1.2e^{-3} (1e^{-8})$	$2e^6 (1e^6)$
5	5000 (500)	-5000 (500)	$1.1e^{-3} (1e^{-8})$	$1.3e^{-3} (1e^{-8})$	$2e^6 (1e^6)$
6	3200 (500)	-3200 (500)	$1.1e^{-3} (1e^{-8})$	$1.3e^{-3} (1e^{-8})$	$1.5e^6 (1e^6)$
7	8000 (500)	-8000 (500)	$1.1e^{-3} (1e^{-8})$	$1.5e^{-3} (1e^{-8})$	$1e^6 (1e^6)$
8	4700 (500)	-4700 (500)	$1.1e^{-3} (1e^{-8})$	$1.4e^{-3} (1e^{-8})$	$1e^6 (1e^6)$
9	6000 (500)	-6000 (500)	$1e^{-3} (1e^{-8})$	$1e^{-3} (1e^{-8})$	$1e^6 (1e^6)$

Table 5.2: Priors used for different cases

Prior	1	2	3	4	5	6	7	8	9
Case 1	x	x	x						
Case 2	x			x	x				
Case 3			x	x		x	x		
Case 4					x	x	x		
Case 5					x	x	x		
Case 6	x	x	x		x				
Case 7				x	x		x		
Case 8				x		x	x		

5.2 Parameter estimation

A list of the results obtained from the filtering using the priors as described in section 5.1 are given in tables E.1 and E.2 in Appendix E.1. The tables include the parameter mean and error estimations, as well as the $RMSE_o$ and the thrust in the x- and y-direction. The determination of the best filter results is based on the lowest $RMSE_o$ value. To further visualize the resulting body force fields, contour plots of all force fields found per case are given in Appendix E.2.

Table 5.3: Best parameter estimations per case

Case #		F_{xo} [$\frac{N}{m^2}$]	F_{yo} [$\frac{N}{m^2}$]	x_o [m]	y_o [m]	β [-]	y_{offset} [m]	$RMSE_o$	Tx [mN]	Ty [mN]
1	μ	9.63E+02	-9.73E+02	1.01E-03	-9.31E-04	1.29E+07	-6.23E-04	1.98E-02	7.58E-01	-7.66E-01
	σ^2	5.34E-01	6.67E+00	7.82E-13	5.13E-13	6.65E+08	3.65E-13	-	-	-
2	μ	1.48E+03	-1.30E+03	1.25E-03	-1.18E-03	4.57E+06	-8.81E-04	5.06E-02	2.29E+00	-2.00E+00
	σ^2	5.70E-01	4.53E+00	5.17E-13	4.87E-13	2.52E+07	2.54E-13	-	-	-
3	μ	2.65E+03	6.27E+03	1.85E-03	1.09E-03	3.20E+05	-2.28E-03	1.28E-01	7.41E+00	1.75E+01
	σ^2	4.19E+00	5.66E+01	1.23E-12	1.59E-12	3.63E+04	2.62E-13	-	-	-
4	μ	9.04E+02	1.65E+03	5.22E-03	4.27E-03	2.11E+06	-3.78E-03	2.76E-01	7.22E+00	1.32E+01
	σ^2	6.99E-02	1.82E+01	1.96E-11	2.29E-11	9.98E+05	2.29E-11	-	-	-
5	μ	2.26E+03	-1.87E+03	4.45E-03	4.08E-03	2.46E+06	-3.50E-03	3.52E-01	1.70E+01	-1.41E+01
	σ^2	2.07E-01	4.30E+01	4.46E-12	2.32E-12	8.12E+05	2.30E-12	-	-	-
6	μ	4.68E+03	-2.90E+03	-1.70E-04	1.51E-04	4.61E+05	-1.68E-03	5.16E-02	3.30E+00	-2.04E+00
	σ^2	1.03E+01	2.65E+02	3.13E-12	4.12E-14	6.22E+04	3.56E-13	-	-	-
7	μ	8.51E+03	-1.65E+04	-4.13E-04	-9.85E-05	4.30E+05	-1.71E-03	8.22E-02	6.00E+00	-1.16E+01
	σ^2	3.58E+00	8.67E+01	7.51E-13	2.32E-14	1.52E+04	2.02E-13	-	-	-
8	μ	2.64E+03	-2.67E+04	-9.99E-05	3.44E-03	6.38E+05	-4.36E-03	1.90E-01	7.48E+00	-7.57E+01
	σ^2	5.20E+00	1.08E+03	1.98E-12	1.18E-12	5.82E+04	1.13E-12	-	-	-

5.2.1 Best force field estimates

The results for the best parameter estimation per case is given in table 5.3. The force fields created by the estimated parameter means per case are given in figure 5.1. Some characteristics of the force field already determined by [Kotsonis \(2012\)](#) can be seen from this figure. The force field increases in size when voltage is increased, while when frequency is increased the field stays relatively similar in shape both increases in intensity. However, especially case 3 and 8 seem to have significant trouble with the estimation of the force in the y-direction. Case 3 even shows a positive force in the y-direction, while case 8 shows very high negative values.

5.2.2 Filter performance

A full overview of all divergence plots and parameter estimation development is given in Appendix E.2. An estimation of the filter performance can be concluded from these plots. A good run should have a $RMSE_o$ which remains about constant for longer integration times, signifying a minimum, while allowing the $RMSE_f$ to decrease as was seen in the twin problem. This is happening for the results from cases 1, 2, 6, and 7. These cases are therefore expected to be good estimations of the force field, given the current force parameterization. The results for these cases will mostly be influenced by bias of the force parameterization, non-Gaussianity in the ensemble, and the finite ensemble size. The cases 3, 4, 5, and 8 however, show an almost continuous increase of $RMSE_o$ after the initialization period. This is most likely due to high bias error introduced through the *sponge layer*, but can also be due to very high bias error in the force parameterization. The results for these cases are assumed to be poor estimations of the force fields.

For successful results (cases 1, 2, 6 and 7), it is expected that influence on the incorrect estimation due to non-Gaussianity and finite ensemble size can be neglected due to the very small variance in the ensemble. Therefore, the bias of the force model is expected to dominate the final $RMSE_o$ value. The low $RMSE_o$ values for the successful cases therefore indicate that



Figure 5.1: Force field when $RMSE_o$ is lowest. Left from top to bottom, cases 1 through 5, with increasing Voltage. Right from top to bottom case 6,2,7,8, with increasing f_{ac}

the current force parameterization is a good approximation to the force field for reconstructing the flow generated by the plasma actuators.

5.2.3 Thrust comparison

To somewhat validate the body force fields a comparison between thrust values obtained using the proposed semi-steady state filtering results and results from [Kotsonis \(2012\)](#) are used. [Kotsonis \(2012\)](#) uses two benchmark methods in the form of a load cell for the horizontal thrust and an integral of momentum balance over a control volume using the flow velocities for thrust measurements in the x- and y-direction. It should be noted that these two methods include the shear stress at the wall and are therefore expected to yield higher values in thrust for the x-direction. Two more methods are used to calculate the thrust: the *reduced method*, and the *gradient method*. The *reduced method* is based on the assumption that, if initial conditions of zero velocity are used, the convective, viscous and pressure gradient terms in (3.29) can be assumed relative small and can be neglected. This assumption reduces (3.29) to:

$$\frac{\mathbf{U}}{t} = \mathbf{f} \quad (5.1)$$

The *reduced method* can only be used for the first moments of actuation.

For the *gradient method*, the Navier-Stokes equation is differentiated in time after which is it

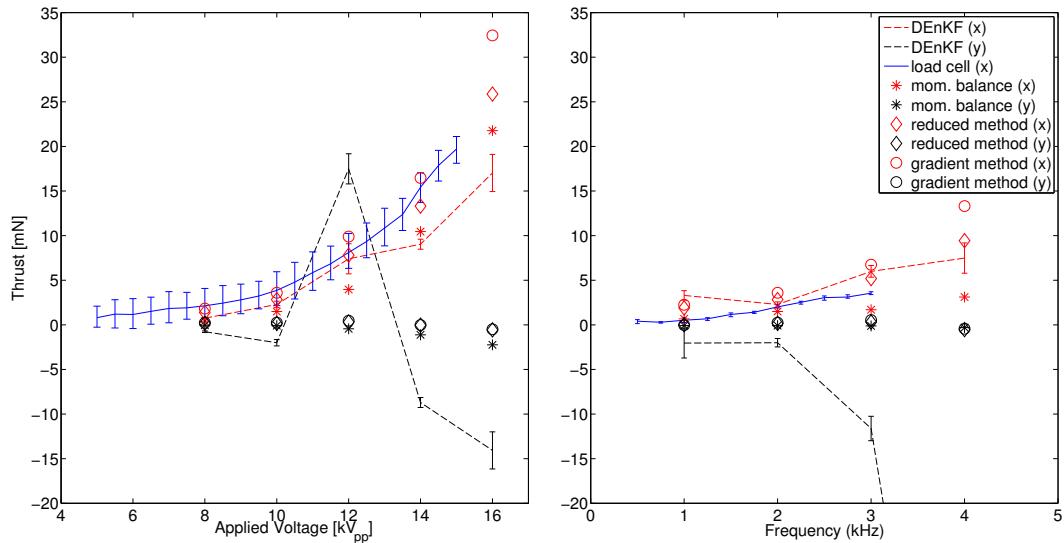


Figure 5.2: Comparison between results obtained by [Kotsonis \(2012\)](#) and the SSS-EnKF method.
NOTE: SSS-EnKF results are plotted with errorbars of $100 \cdot 10^{-2}$

integrated from the initial conditions of zeros flow velocity and zero pressure gradient. Both methods compute the complete body force field and thrust can be calculated using (4.7).

The comparison between results from [Kotsonis \(2012\)](#) and the filtering method are shown in figure 5.2. An error estimation for the thrust can be computed using the parameter error estimates given in terms of variance. This thrust variance is calculated using the Monte Carlo method with ensemble size of $N = 100$. The calculated thrust variance times 100 is shown in figure 5.2 as errorbars. The multiplication is used to be able to show the relative estimated variance between cases. These low values for thrust error show the potential of the current method of high accurate thrust estimations.

Figure 5.2 gives a comparison of the EnKF method results and results obtain by ([Kotsonis, 2012](#)). The estimated thrust in the x-direction seems to show similar values as the benchmark values. Even the filter runs which were identified as having poor results, still show good agreement with the benchmark results for the thrust in the x-direction.

The thrust in the y-direction seems to be incorrectly estimated for most cases. Especially case 3, which has a positive value for the thrust in the y-direction, seems to be completely incorrect. whereas the thrust in the y-direction seems to be be completely off for cases with higher flow velocities. Even for case 7 ($f_{ac} = 3$ kHz) the thrust in the y-direction seems to be overestimated.

5.2.4 Pressure correction

This incorrect estimation of the thrust force can be partially due to the existence of a steady pressure gradient, counteracting parts of the body force. The current parameterization of the

body force might allow for such a steady state pressure gradient as describe in section 3.4.1. This pressure gradient might not be physical and solely dependents on the force parameterization chosen. A new definition of total thrust in both the x and y-direction can be created incorporating both the pressure gradient as well as the force field.

$$\mathbf{F} = \int_V (\mathbf{f} + \nabla p) dV \quad (5.2)$$

The pressure field used to calculate the integrated force is generated by a single forecast run to steady state using the mean of the estimated parameters. This method of acquiring the pressure field however does not allow for a thrust error estimation using the ensemble method due to very high computational costs, and will therefore not be given when using (5.2). The calculated force fields with the pressure correction are shown in figure 5.3. For comparison, the force fields calculated using the *gradient method* by Kotsonis (2012), are shown in figure 5.4. It can be seen that the force fields calculated for cases which were expected to be successful (cases 1 ($8kV_{pp}$), 2 ($10kV_{pp}/f_{ac} = 2$ kHz), 6 ($f_{ac} = 1$ kHz) and 7($f_{ac} = 4$ kHz)), show good agreement now with the results of the *gradient method*. Main differences can be contributed to relative simple parameterization of the force field used by the EnKF method.

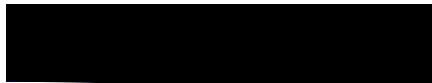


Figure 5.3: $\mathbf{f} - \nabla p$. Left from top to bottom, cases 1 through 5, with increasing Voltage. Right from top to bottom case 6,2,7,8, with increasing f_{ac}



Figure 5.4: Force fields using *gradient method* by Kotsonis (2012). Left from top to bottom, cases 1 through 5, with increasing Voltage. Right from top to bottom case 6,2,7,8, with increasing f_{ac}

Figure 5.5 shows the results using (5.2) for the calculation of thrust. It can be noted that the thrust in the x-direction is not much affected by the incorporation of the pressure gradient. The thrust in the y-direction however does change significantly. Case 7 ($f_{ac} = 3 \text{ kHz}$), which previously overestimated the thrust in the y-direction now shows fairly good agreement with the benchmark values. Also the thrust estimation for case 3 ($12kV_{pp}$) and case 5 ($16kV_{pp}$) moved towards the benchmark results.

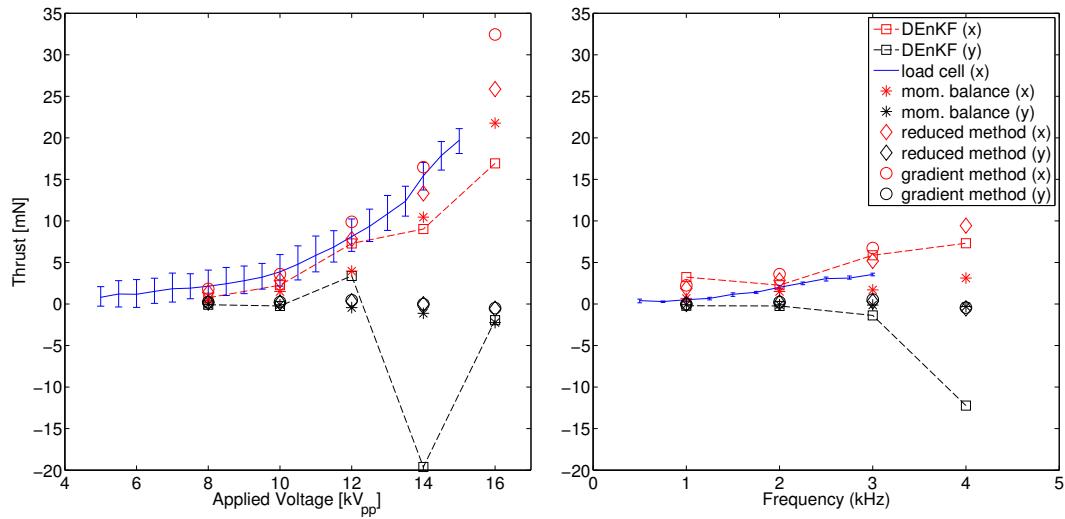


Figure 5.5: Comparison between results obtained by [Kotsonis \(2012\)](#) and the SSS-EnKF method. Incorporation the pressure gradient in the thrust calculation

Chapter 6

Conclusion

An initial study into the use of the EnKF, and derivatives of this method, for the parameter estimation describing the force field generated by a plasma actuator was conducted in the form of a twin problem and an application to quasi-steady plasma actuator configurations. A finite difference solver was created solving the full laminar Navier-Stokes equation to use as forecast model. Observations were used in the form of high resolution time averaged PIV data for eight different plasma actuator configurations. Initial priors were created based on zero flow conditions with an initial estimation of parameter mean and variance. A slight modification to the EnKF was made, resulting in the Semi Steady State EnKF (SSS-EnKF), to correct for sub-optimal update steps. The correction is necessary due to the loss of connectivity between parameters and the observed flow variables caused by non-Gaussianity in the initial priors. The idea of the SSS-EnKF is to integrate the full ensemble state forward in time by a given time step, in order to let information of the parameters to flow back into the flow variables, ensuring that all ensemble members are at steady state for the corresponding parameter values.

The semi steady state filter method in the current configuration could prove useful to estimate the body force field generated by a plasma actuator. Different force model parameterizations could be tested using this method including the error estimation for the parameters describing the force model used. The study of the twin problem showed that most filter settings as integration between updates, type of filter (EnKF vs DEnKF) and ensemble size, do not greatly affect the solution. However, the current set-up, especially with regards to the forecast model used, causes the SSS-EnKF to become highly sensitive to the initial prior. If the initial prior contains members far away from the final solution, a lot of bias will be introduced into the system to keep the ensemble members stable. When this happens, the estimation of the parameter mean and variance will be dominated by bias error, resulting in poor results. Parameter estimation with EnKF using PIV data could theoretically yield force fields with better accuracy than current techniques using velocity based experimental measurements. This thesis is therefore only a preliminary study into the application of parameter estimation with EnKF using PIV data. Further research could improve upon the current method to

make it more robust and even estimate time dependent parameters.

Recommendations

If further research in the area of parameter estimation using data assimilation for quasi-steady state characteristics of the plasma actuator were to be of interest the following recommendation would be made to improve the method.

For EnKF parameter estimation with plasma actuators in general:

1. **More robust forecast model.** Use of a forecast model in which the initial start-up vortex leaves the domain completely, reducing the chance of very high flow velocities in the far field and making the use of the *sponge layer* obsolete. This will most likely cause the method to become far less dependent on the initial prior.
2. **Improvement of the linearity of the initial prior.** Using a different initial integration time per ensemble member, allowing each ensemble member to reach steady state independently.
3. **Body fore parameterization** The parameterization should contain the lowest possible parameters (smoother response surface), while describing the body force as accurate as possible (reducing bias in the forecast). Improvements could be made by solving for an approximate electrical field with only a parameterization of the charge distribution.
4. **Time dependent periodic body forceeld.** Use the EnKF to determine the temporal body force field. This allow the EnKF to use its ability to increase the accuracy of the flow field not only spatially, but also in time.

Semi steady state filtering with quasi steady state flow field generated by plasma actuator:

1. **New type of in action.** In order keep the true parameter values inside the range of the ensemble members, use can be made of some sort of limiting inflation, ensuring the total variance only decreases by a limited amount, never losing the true solution inside the range of ensemble members.
2. **Replacing unstable ensemble members.** Ensemble members with high flow velocity, causing the introduction of bias through the *sponge layer*, can be replaced by a different stable ensemble member, or some interpolation between random stable ensemble members.
3. **Ensure steady state.** Integrate each ensemble member with variable integration time in between updates. The variable integration time is then determined by the time for the ensemble member to reach steady state.

Bibliography

A practical guide to Particle Image Velocimetry. Springer, 2007.

- T. Albrecht, T. Weier, G. Gerbeth, H. Metzkes, and J. Stiller. A method to estimate the planar, instantaneous body force distribution from velocity field measurements. *Physics of Fluids*, 23(2):021702, 2011.
- J. L. Anderson. An Ensemble Adjustment Kalman Filter for Data Assimilation. *Monthly Weather Review*, 129:2884–2903, 2001.
- J. D. Annan, J. C. Hargreaves, N. R. Edwards, and R. Marsh. Parameter estimation in an intermediate complexity earth system model using an ensemble Kalman filter. *Ocean Modelling*, 8(1-2):135–154, January 2005.
- G. Burgers, P. J. van Leeuwen, and G. Evensen. Analysis Scheme in the Ensemble Kalman Filter. *Monthly Weather Review*, 126(6):1719–1724, 1998.
- Y. Chen and D. Zhang. Data assimilation for transient flow in geologic formations via ensemble Kalman filter. *Advances in Water Resources*, 29(8):1107–1122, August 2006.
- A. Cristofolini, C. A. Borghi, and G. Neretti. Charge distribution on the surface of a dielectric barrier discharge actuator for the fluid-dynamic control. *Journal of Applied Physics*, 113(14):143307, 2013.
- S. Dong, G. E. Karniadakis, and C. Chryssostomidis. A robust and accurate outflow boundary condition for incompressible flow simulations on severely-truncated unbounded domains. *Journal of Computational Physics*, January 2014.
- A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.
- C. L. Enloe, T. E. McLaughlin, R. D. Vandyken, and K. D. Kachner. Mechanisms and Responses of a Single Dielectric Barrier Plasma Actuator : Plasma Morphology. *AIAA*, 42(3):599–594, 2004a.
- C. L. Enloe, T. E. McLaughlin, R. D. Vandyken, and K. D. Kachner. Mechanisms and Responses of a Single Dielectric Barrier Plasma Actuator : Geometric Effects. *AIAA*, 42(3):594–604, 2004b.

- G. Evensen. eve94a.pdf. *Journal of Geophysical Research*, 99(C5):143–10, 1994.
- G. Evensen. The Ensemble Kalman Filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53(4):343–367, November 2003.
- G. Evensen. The Ensemble Kalman Filter for Combined State and Parameter Estimation. *IEEE Control Systems magazing*, (June):83–104, 2009.
- H.J. Hendricks Franssen and W. Kinzelbach. Ensemble Kalman filtering versus sequential self-calibration for inverse modelling of dynamic groundwater flow systems. *Journal of Hydrology*, 365(3-4):261–274, February 2009.
- J. H. Hacker and C. Snyder. Ensemble Kalman Filter Assimilation of Fixed Screen-Height Observations in a Parameterized PBL. *Monthly Weather Review*, 133:3260–3275, 2005.
- J. Harlim and A. J. Majda. Catastrophic Filter Divergence in Filtering Nonlinear Dissipative Systems. *Communications in Mathematical Sciences*, 8(1), 2010.
- H. J. Hendricks Franssen and W. Kinzelbach. Real-time groundwater flow modeling with the Ensemble Kalman Filter: Joint estimation of states and parameters and the filter inbreeding problem. *Water Resources Research*, 44(9):n/a–n/a, September 2008.
- B. Jayaraman and W. Shyy. Modeling of dielectric barrier discharge-induced fluid dynamics and heat transfer. *Progress in Aerospace Sciences*, 44(3):139–191, April 2008.
- A. H. Jazwinski. *Stochastic processes and filtering theory*. Academic press, 1970.
- R E Kalman. A New Approach to Linear Filtering and Prediction Problems. *Basic Engineering*, 82(Series D):35–45, 1960.
- M. Kotsonis. *Dielectric Barrier Discharge Actuators for Flow Control; Diagnostics, Modeling, Application*. PhD thesis, Technische Universiteit Delft, Delft, 2012.
- H. Koyama and M. Watanabe. Reducing Forecast Errors Due to Model Imperfections Using Ensemble Kalman Filtering. *Monthly Weather Review*, 138(8):3316–3332, August 2010.
- J. Kriegseis, C. Schwarz, C. Tropea, and S. Grundmann. Velocity-information-based force-term estimation of dielectric-barrier discharge plasma actuators. *Journal of Physics D: Applied Physics*, 46(5):055202, February 2013.
- M. Neumann, C. Friedrich, J. Czarske, J. Kriegseis, and S. Grundmann. Determination of the phase-resolved body force produced by a dielectric barrier discharge plasma actuator. *Journal of Physics D: Applied Physics*, 46(4):042001, January 2013.
- G. H. D. Olivares. *Cost effective groundwater quality sampling network design*. PhD thesis, University of Vermont, Burlington, 1998.
- D. F. Opaits, M. N. Shneider, Richard B. Miles, A. V. Likhanskii, and S. O. Macheret. Surface charge in dielectric barrier discharge plasma actuators. *Physics of Plasmas*, 15(7):073505, 2008.
- B. Orescanin, B. Rajkovic, M. Zupanski, and D. Zupanski. Soil model parameter estimation with ensemble data. *Atmospheric science letters*, 131(April):127–131, 2009.

- D. M. Orlov and T. C. Corke. Electric Circuit Model for Aerodynamic Plasma Actuator. *AIAA*, Paper No., 2006.
- J. R. Roth, D. M. Sherman, S. P. Wilkinson, B. Drive, J. Reece, and M. Sherman. Flow control with boundary a one atmosphere uniform discharge plasma glow. *AIAA*, 28(03), 1998.
- P. Sakov and P. R. Oke. A deterministic formulation of the ensemble Kalman filter: an alternative to ensemble square root filters. *Tellus A*, 60(2):361–371, March 2008.
- P. Sakov, G. Evensen, and L. Bertino. Asynchronous data assimilation with the EnKF. *Tellus A*, 62(1):24–29, January 2010.
- B. Sanderse and B. Koren. Accuracy analysis of explicit Runge Kutta methods applied to the incompressible NavierStokes equations. *Journal of Computational Physics*, 231(8):3041–3063, April 2012.
- F. Scarano. *Experimental Aerodynamics*. Delft University of Technology, 2007.
- D. Schiavazzi, F. Coletti, G. Iaccarino, and J. K. Eaton. A matching pursuit approach to solenoidal filtering of three-dimensional velocity measurements. pages 127–138, 2012.
- A. Sciacchitano, R. P. Dwight, and F. Scarano. Navier-Stokes simulations in gappy PIV data. pages 1–24, 2011.
- W. Shyy, B. Jayaraman, and A. Andersson. Modeling of glow discharge-induced fluid dynamics. *Journal of Applied Physics*, 92(11):6434, 2002.
- K. P. Singh and S. Roy. Force approximation for a plasma actuator operating in atmospheric air. *Journal of Applied Physics*, 103(1):013305, 2008.
- S. Skachko, J.-M. Brankart, F. Castruccio, P. Brasseur, and J. Verron. Improved Turbulent AirSea Flux Bulk Parameters for Controlling the Response of the Ocean Mixed Layer: A Sequential Data Assimilation Approach. 26(3):538–555, March 2009.
- J. V. T. Sørensen. Efficient Kalman filter techniques for the assimilation of tide gauge data in three-dimensional modeling of the North Sea and Baltic Sea system. *Journal of Geophysical Research*, 109(C3):C03017, 2004.
- P. J. van Leeuwen. Particle Filtering in Geophysical Systems. *Monthly Weather Review*, 137(12):4089–4114, December 2009.
- X. Yang and T. Delsole. Using the ensemble Kalman filter to estimate multiplicative model parameters. *Tellus A*, 61(5):601–609, October 2009.

Appendix A

PIV setup

Table A.1: PIV setup

camera	Photron Fastcam SA1 high speed CMOS
sensor size	1024 x 512
acquisition rate	10 kHz
lens	Nikkor 105 mm
f-stop	4 + extension tubes
magnification	0.8
FOV	15x6 mm
seeding particles	olive oil droplets (1 m diameter)
laser	Quantronix Darwin-Duo 80 W at 3 kHz
light sheet thickness	2 mm
Analizing software	Davis 7.4 (Lavision GmbH)
interrogation window size	12x12 pixels
overlap	75 %

Appendix B

Twin problem initial prior statistics

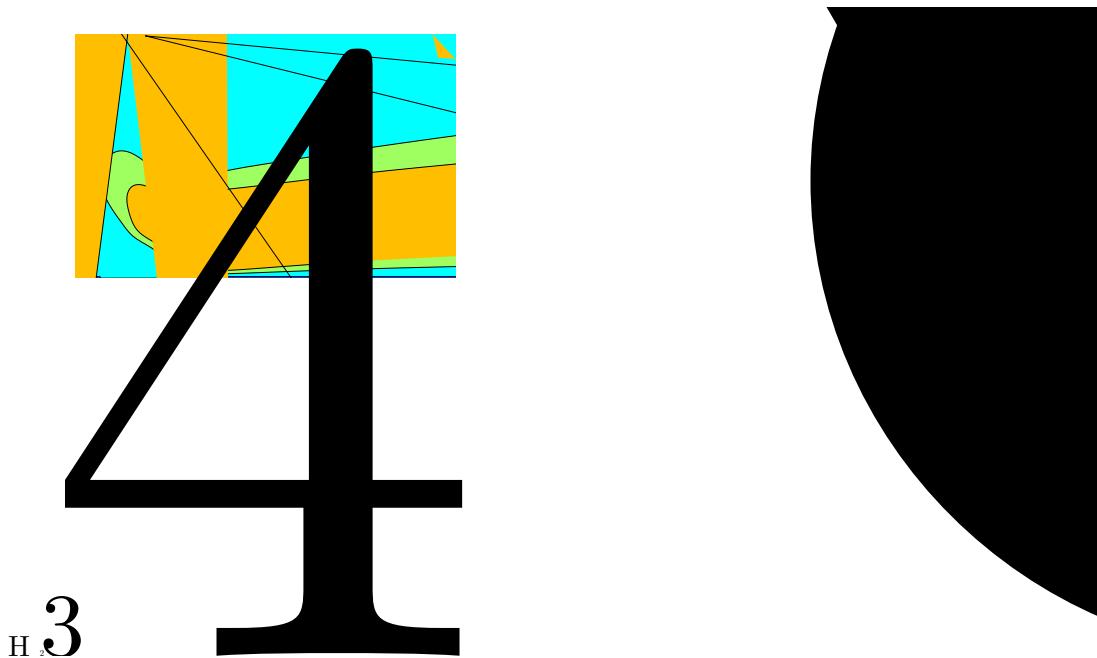


Figure B.1: Uni-varient statistics of prior 2a x-velocity, $To = 50$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

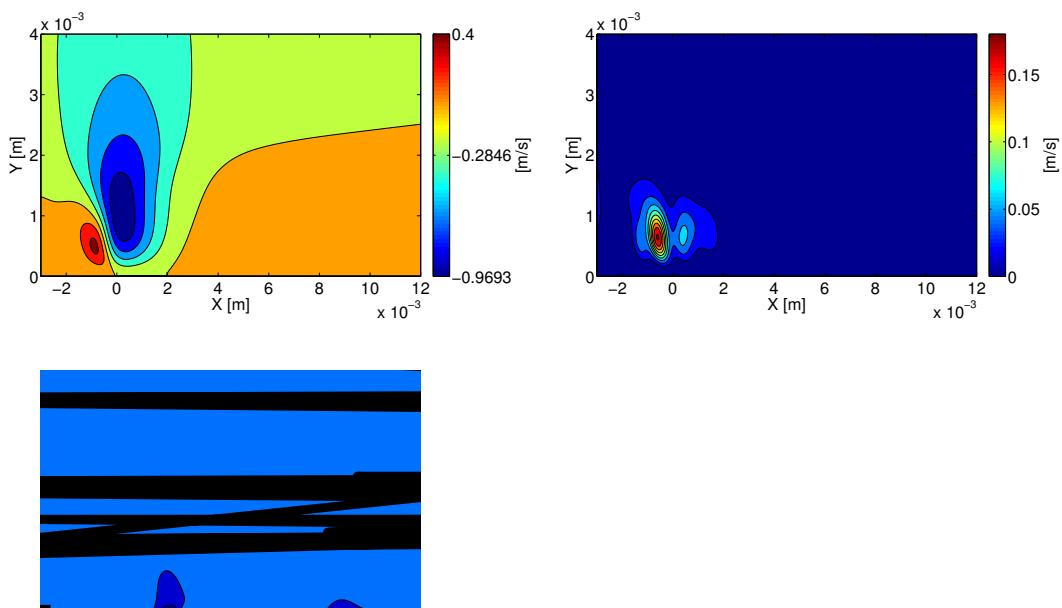


Figure B.2: Uni-varient statistics of prior 2a y-velocity, $To = 50$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

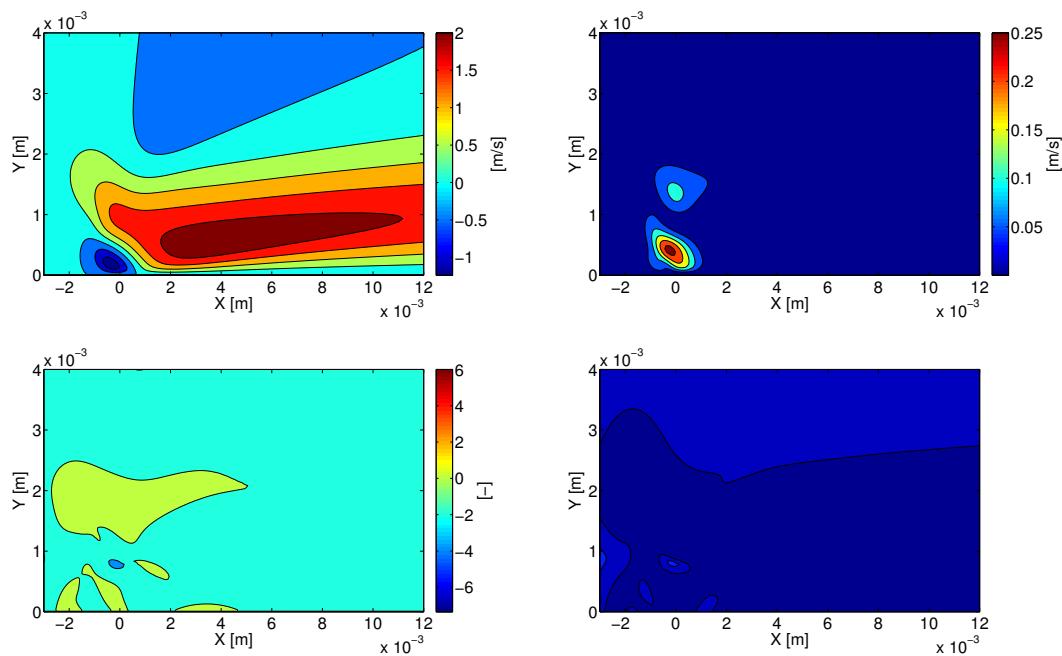


Figure B.3: Uni-variant statistics of prior 2a x-velocity, $To = 75$; V^2 (top right), σ^2 (top left), σ_k (bottom left), and k (bottom right)

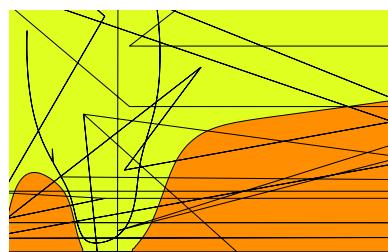


Figure B.4: Uni-variant statistics of prior 2a y-velocity, $To = 75$; V^2 (top right), σ^2 (top left), σ_k (bottom left), and k (bottom right)

Figure B.5: Uni-varient statistics of prior 2a x-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

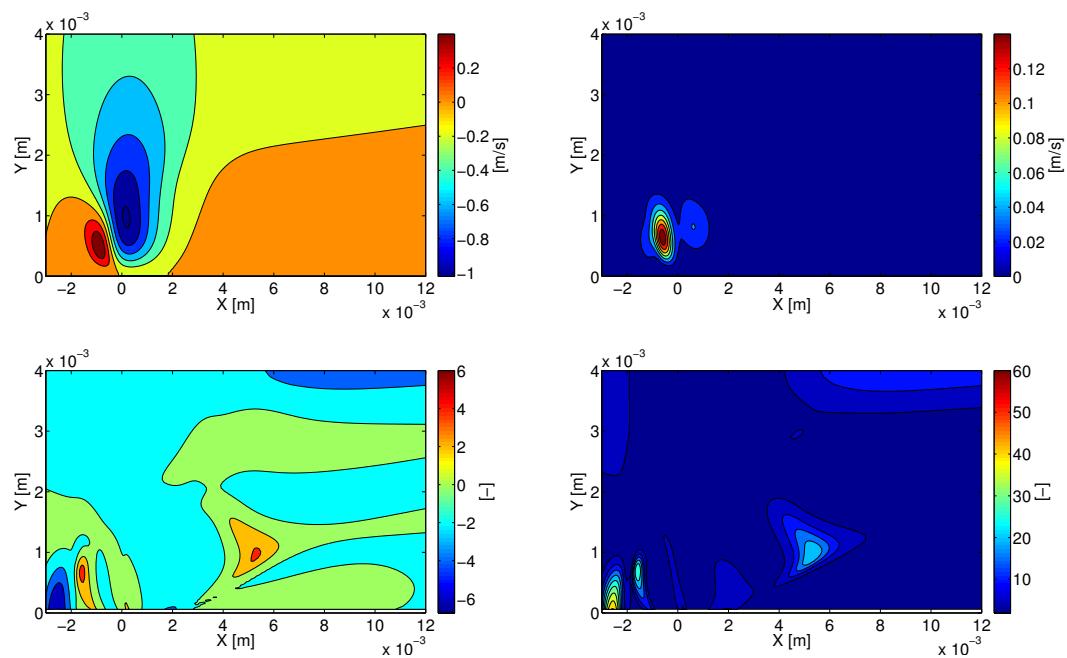


Figure B.6: Uni-varient statistics of prior 2a y-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

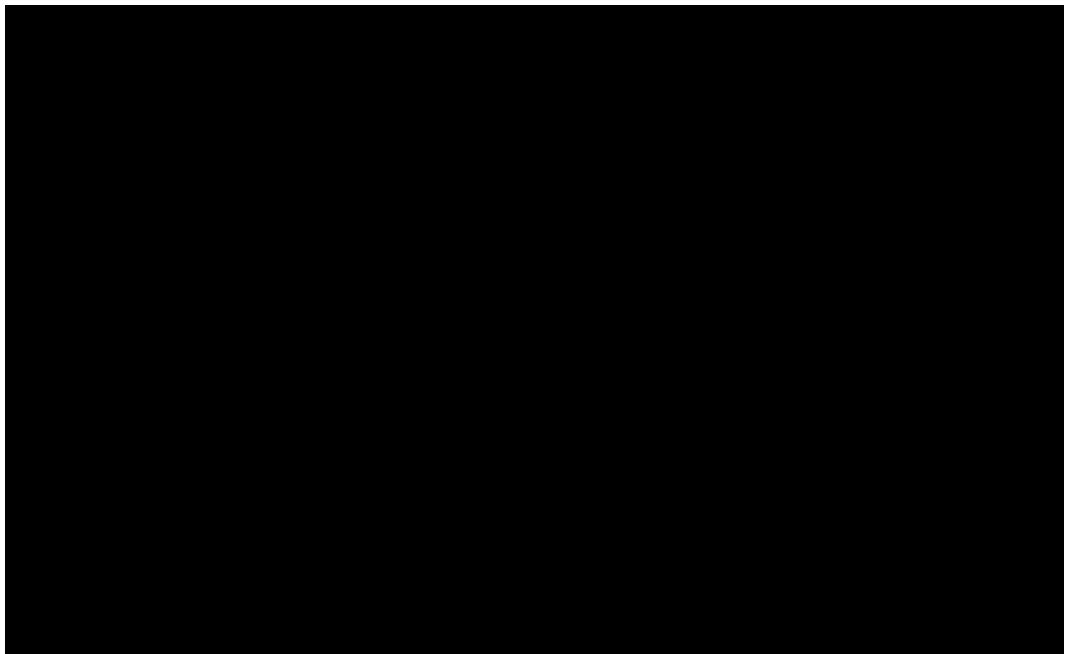


Figure B.7: Uni-varient statistics of prior 2b x-velocity, $To = 50$; σ (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

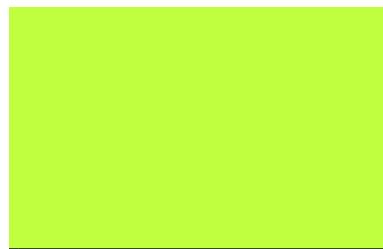


Figure B.8: Uni-varient statistics of prior 2b y-velocity, $To = 50$; σ (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

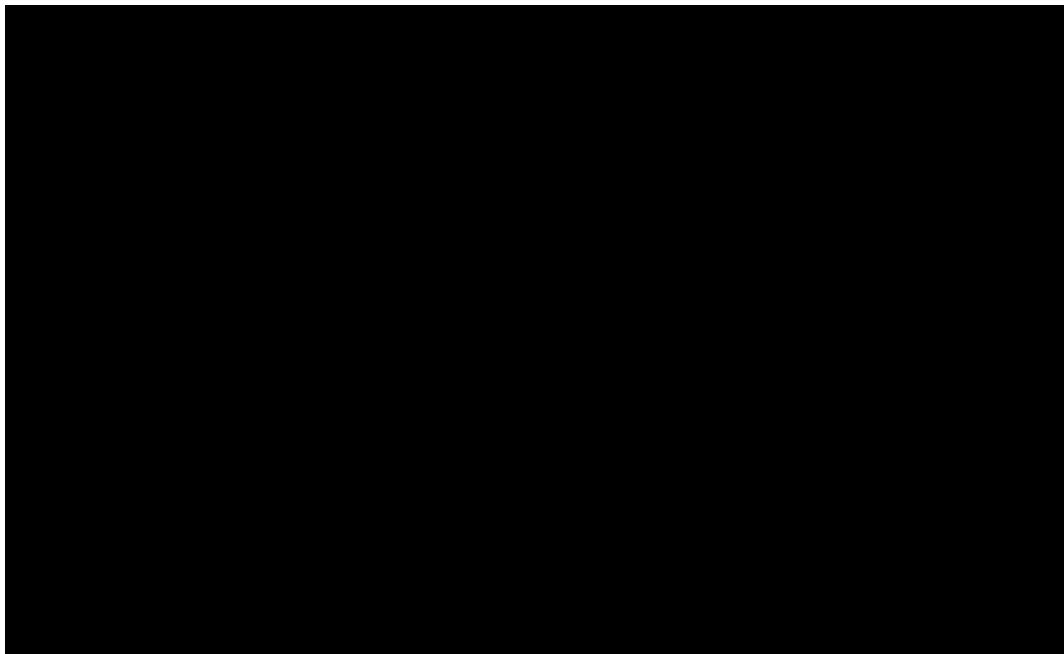


Figure B.9: Uni-variant statistics of prior 2b x-velocity, $To = 75$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

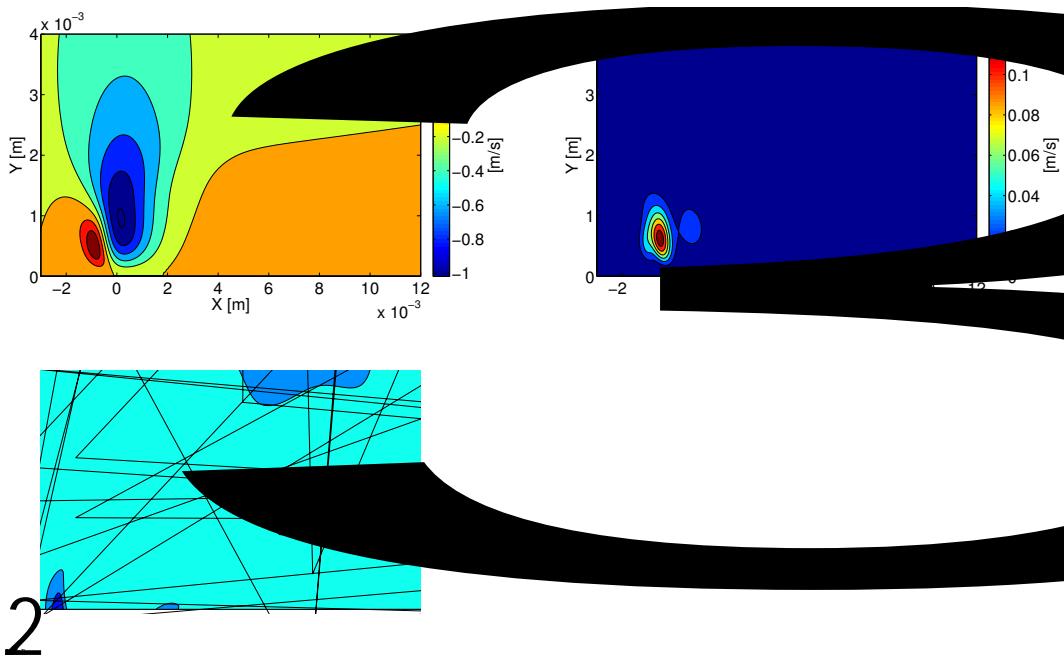


Figure B.10: Uni-varient statistics of prior 2b y-velocity, $To = 75$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

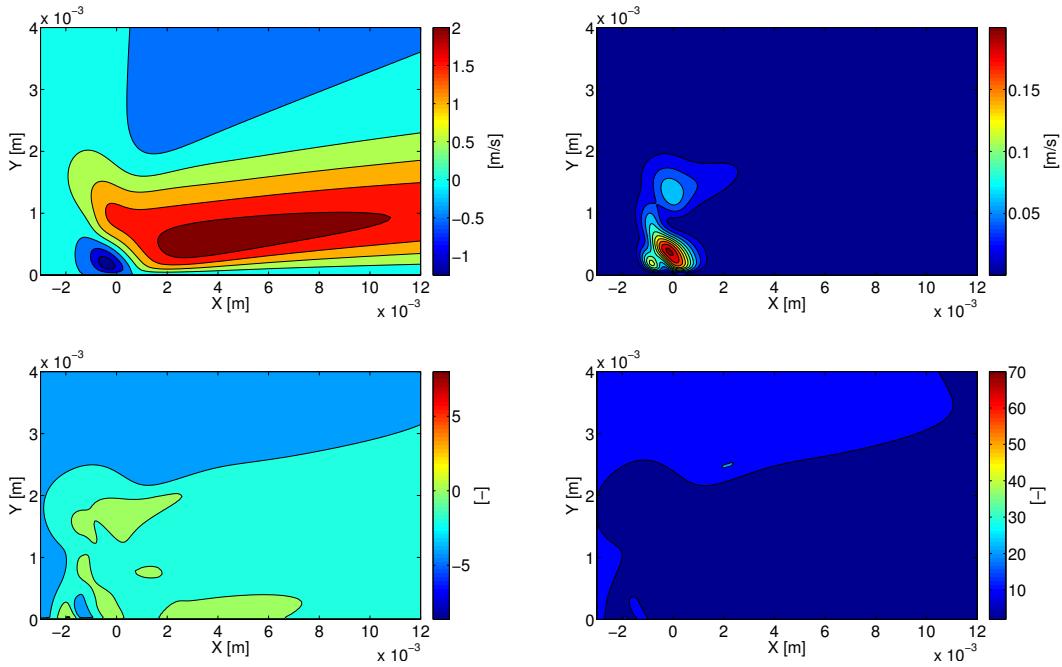


Figure B.11: Uni-variance statistics of prior 2b x-velocity, $To = 100$; σ_x^2 (top left), $\sigma_{x^2}^2$ (top right), sk_x (bottom left), and k_x (bottom right)

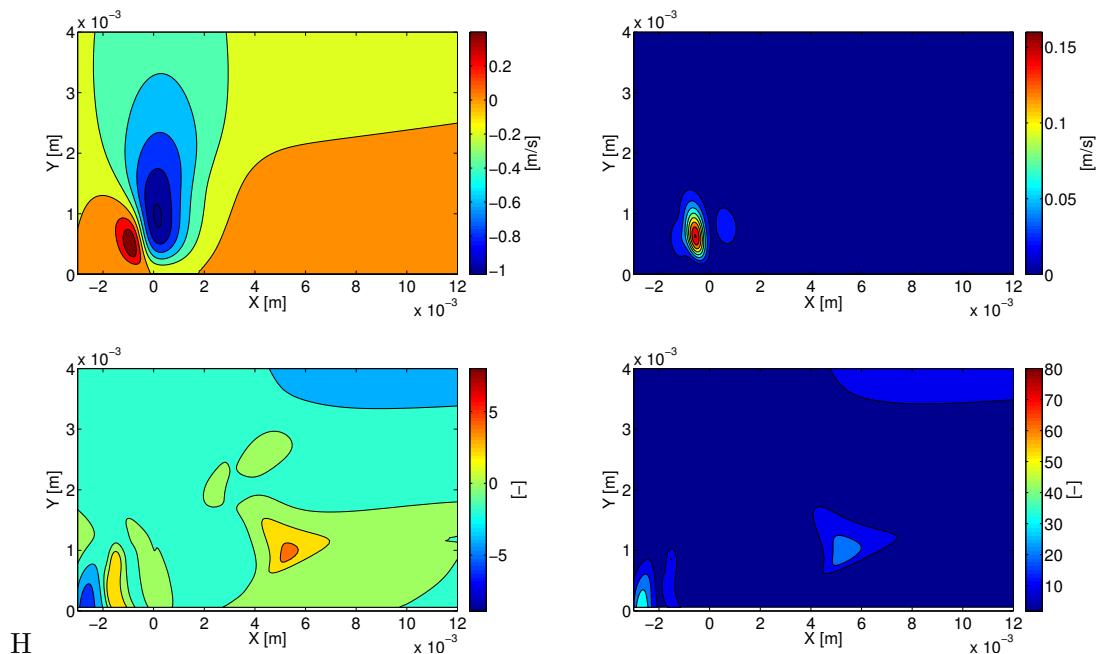


Figure B.12: Uni-variance statistics of prior 2b y-velocity, $To = 100$; σ_y^2 (top left), $\sigma_{y^2}^2$ (top right), sk_y (bottom left), and k_y (bottom right)

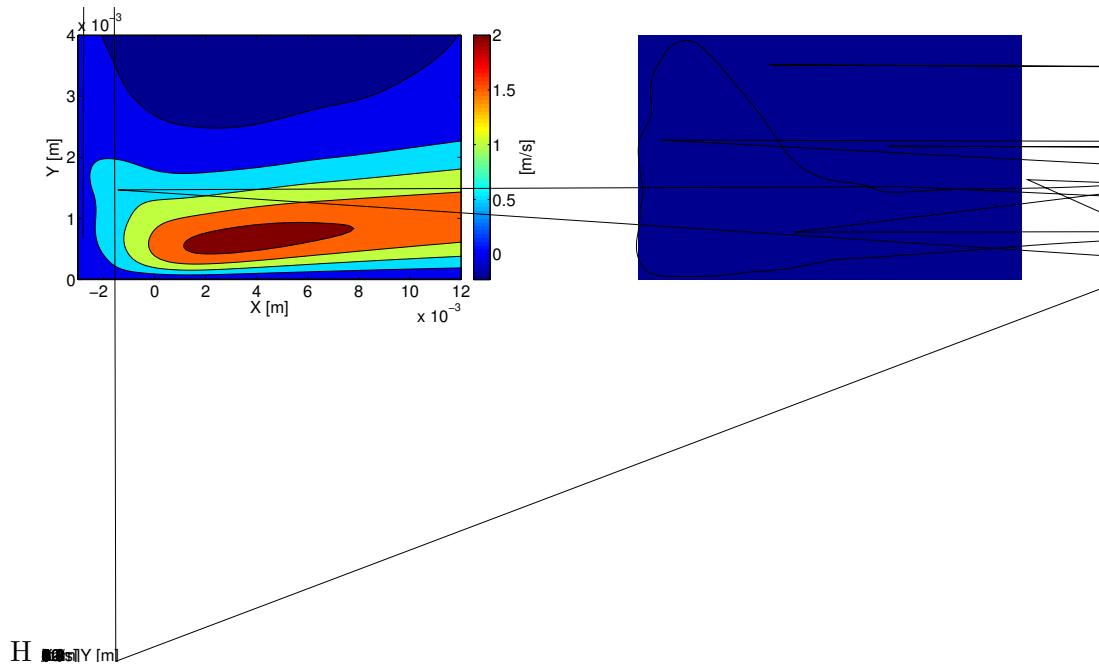


Figure B.13: Uni-variant statistics of prior 3a x-velocity, $To = 50$; $|v|$ (top left), $|v|^2$ (top right), sk (bottom left), and k (bottom right)

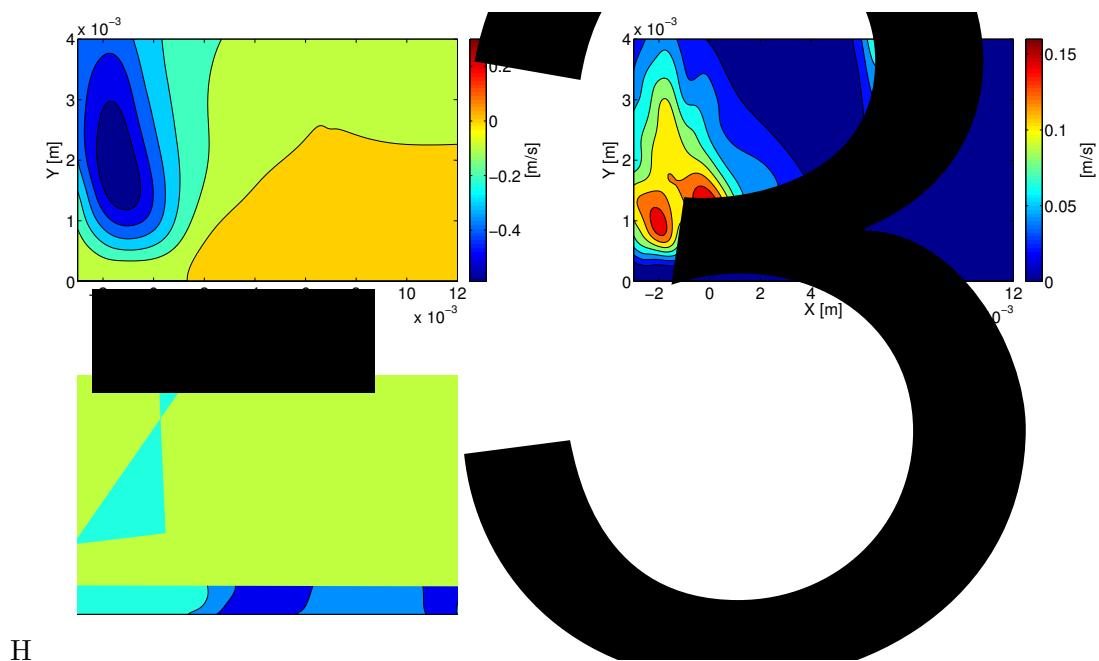


Figure B.14: Uni-variant statistics of prior 3a y-velocity, $To = 50$; $|v|$ (top left), $|v|^2$ (top right), sk (bottom left), and k (bottom right)

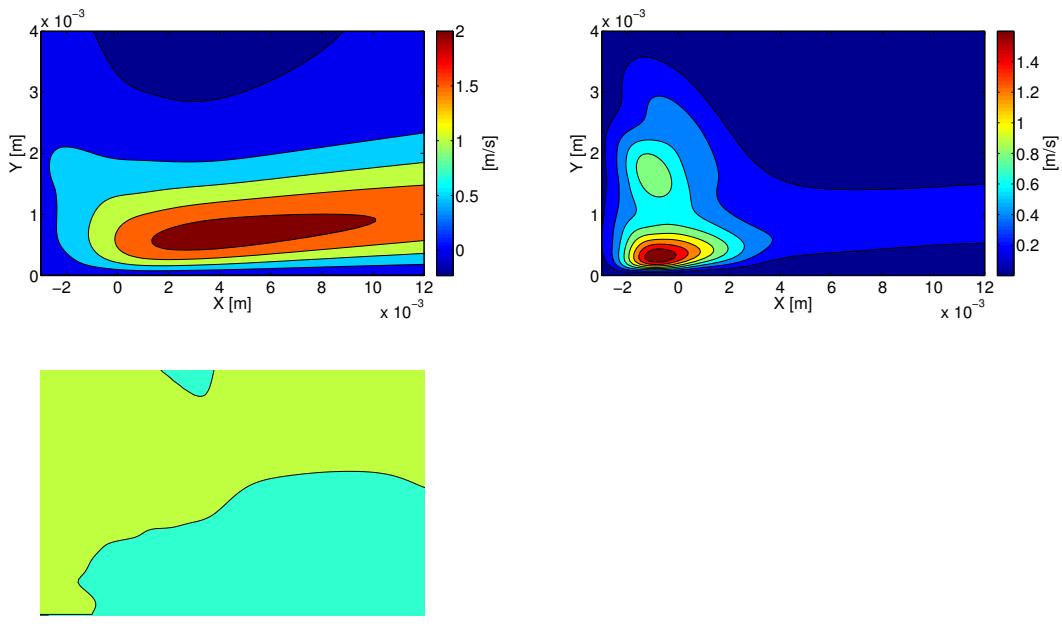


Figure B.15: Uni-variant statistics of prior 3a x-velocity, $To = 75$; $|u|$ (top left), $|u|^2$ (top right), sk (bottom left), and k (bottom right)

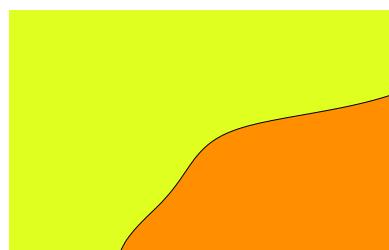
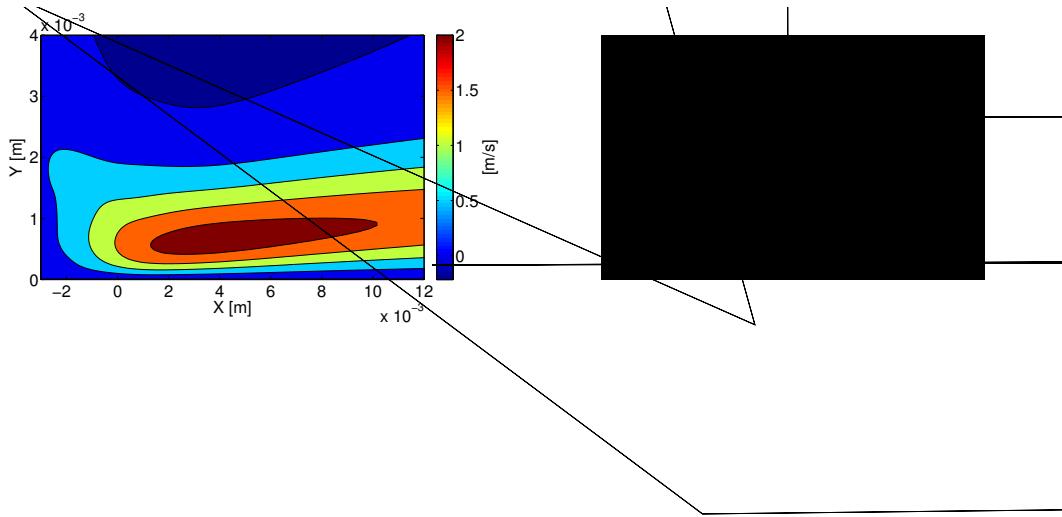


Figure B.16: Uni-variant statistics of prior 3a y-velocity, $To = 7$; $|v|$ (top left), $|v|^2$ (top right), sk (bottom left), and k (bottom right)



H sk $Y [m]$

Figure B.17: Uni-variant statistics of prior 3a x-velocity, $To = 100$; $|v|^2$ (top right), sk (bottom left), and k (bottom right)

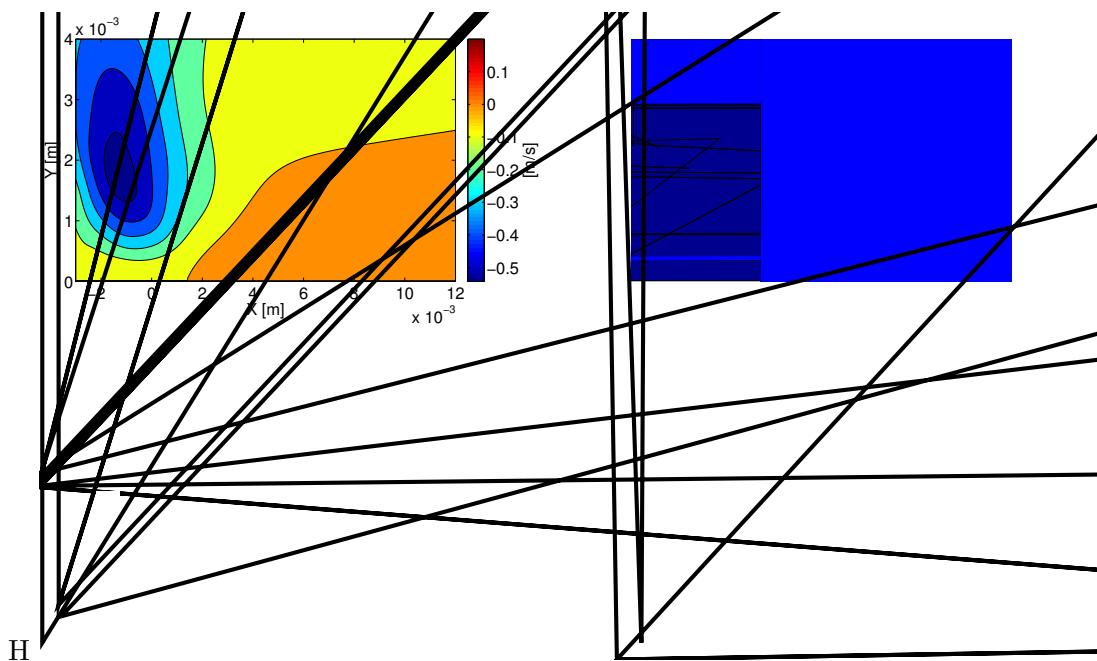


Figure B.18: Uni-variant statistics of prior 3a y-velocity, $To = 100$; $|v|^2$ (top right), sk (bottom left), and k (bottom right)

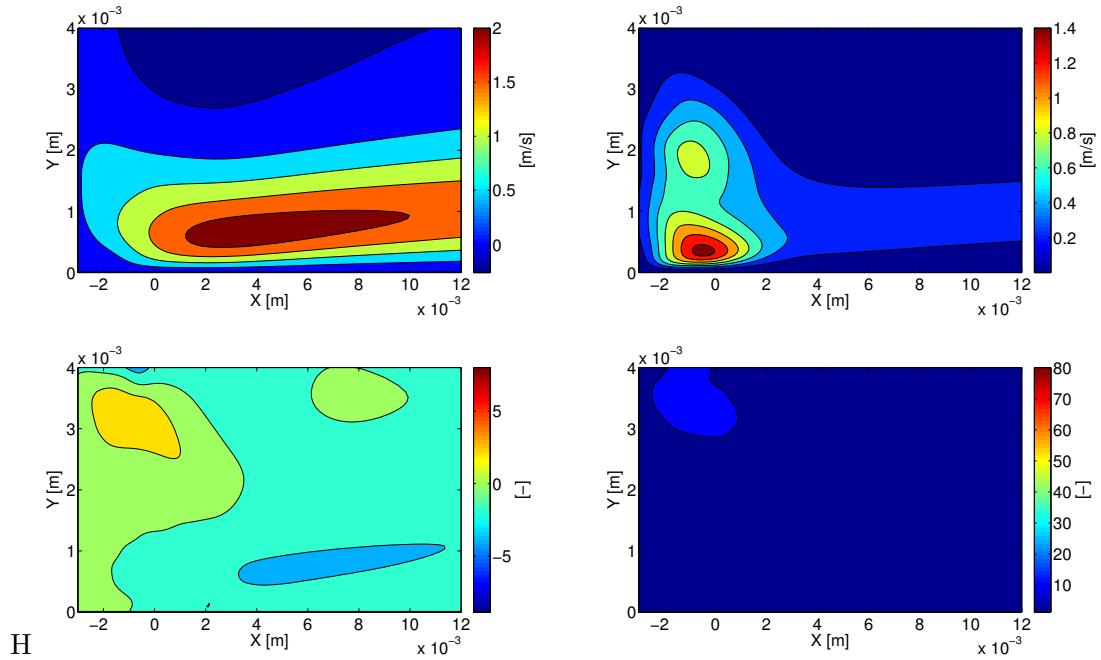


Figure B.19: Uni-variate statistics of prior 3b x-velocity, $To = 50$; σ^2 (top right), σ_k (bottom left), and k (bottom right)

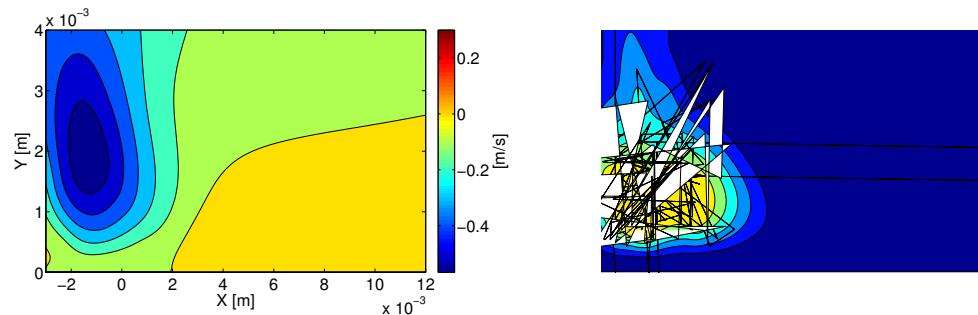


Figure B.20: Uni-variate statistics of prior 3b y-velocity, $To = 50$; σ^2 (top right), σ_k (bottom left), and k (bottom right)

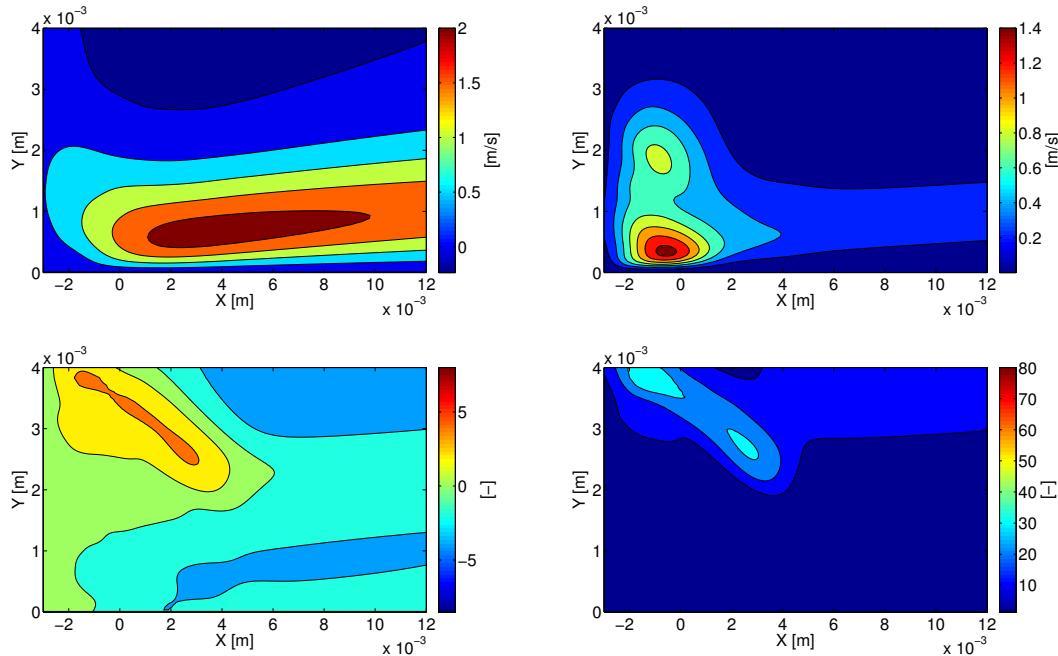


Figure B.21: Uni-variate statistics of prior 3b x-velocity, $To = 75$; \bar{u} (top left), σ_u (top right), sk (bottom left), and k (bottom right)

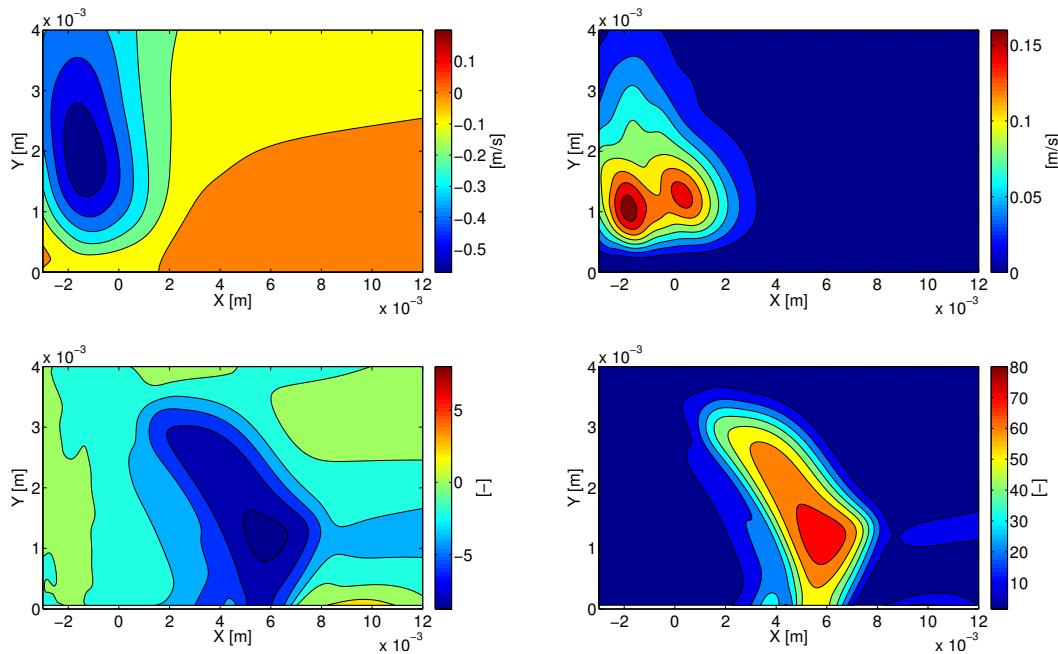


Figure B.22: Uni-variate statistics of prior 3b y-velocity, $To = 75$; \bar{u} (top left), σ_u (top right), sk (bottom left), and k (bottom right)

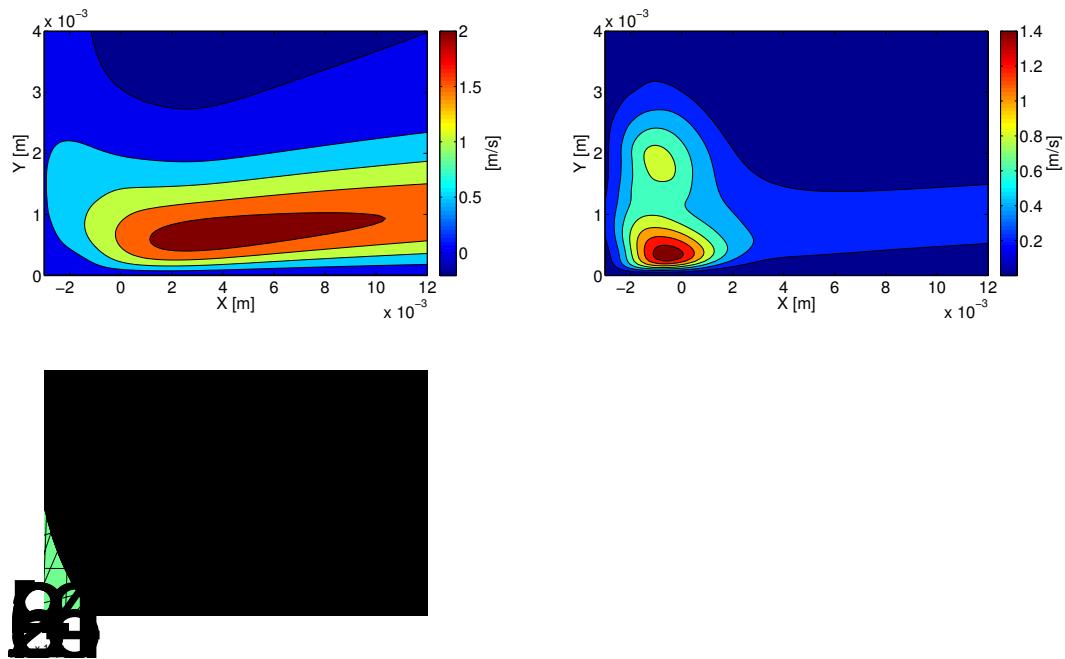


Figure B.23: Uni-varianc statistics of prior 3b x-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

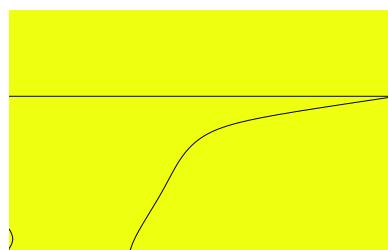


Figure B.24: Uni-varianc statistics of prior 3b y-velocity, $To = 100$; σ^2 (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

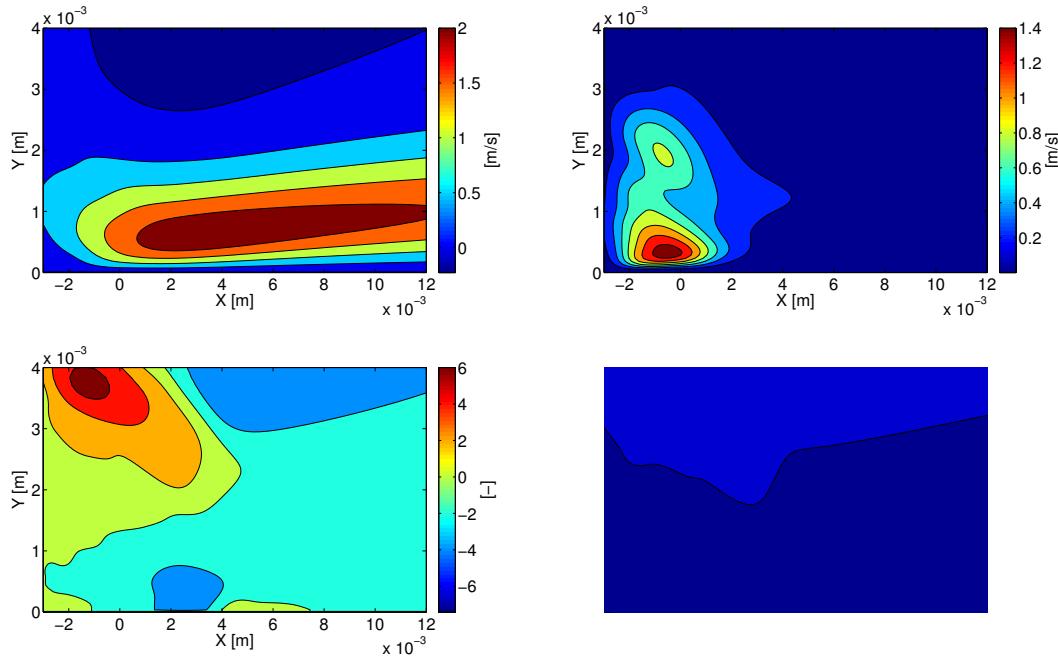


Figure B.25: Uni-variate statistics of prior 4 x-velocity, $T_0 = 100$; \bar{v} (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

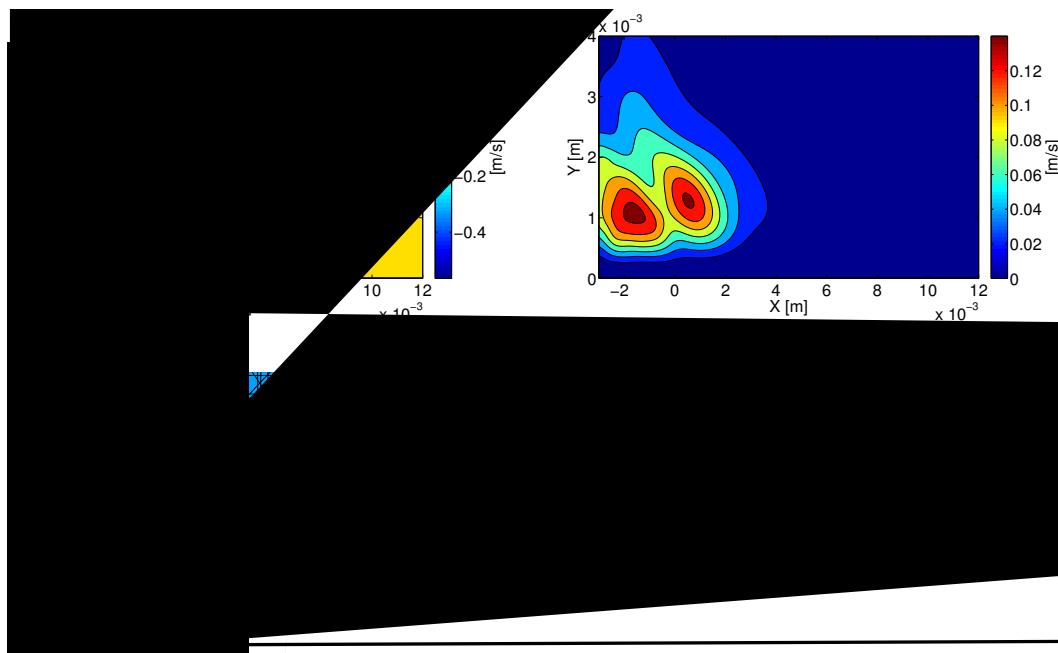


Figure B.26: Uni-variate statistics of prior 4 y-velocity, $T_0 = 100$; \bar{v} (top left), σ^2 (top right), sk (bottom left), and k (bottom right)

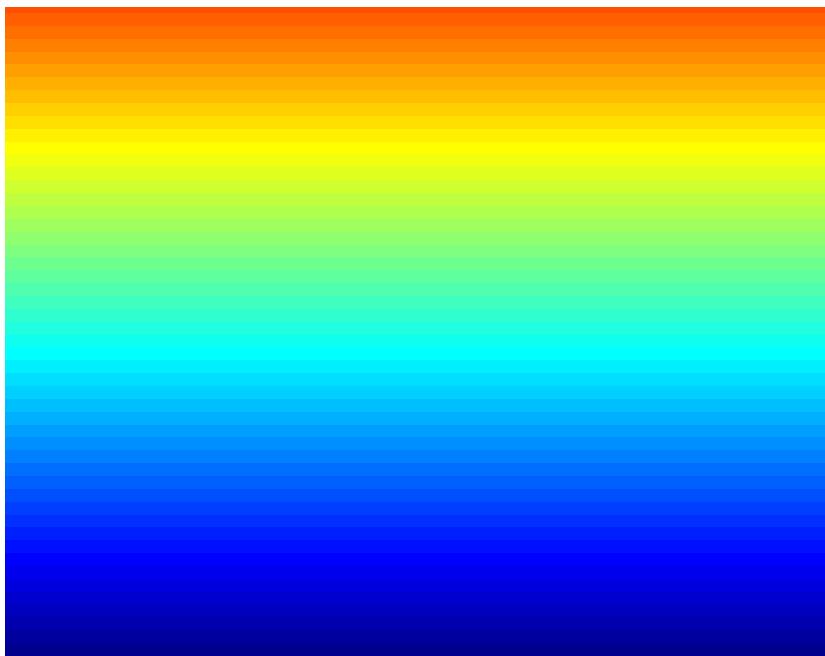


Figure B.27: Uni-varient statistics of prior 5 x-velocity, $To = 100$; σ^2 (top left), sk (bottom left), and k (bottom right)

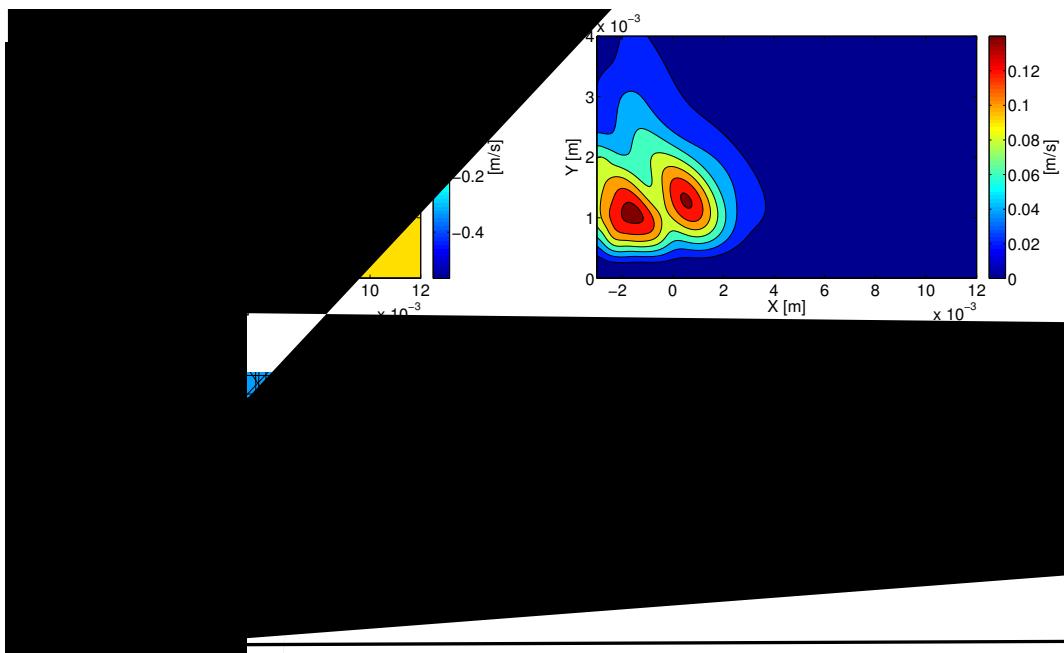


Figure B.28: Uni-varient statistics of prior 5 y-velocity, $To = 100$; σ^2 (top left), sk (bottom left), and k (bottom right)

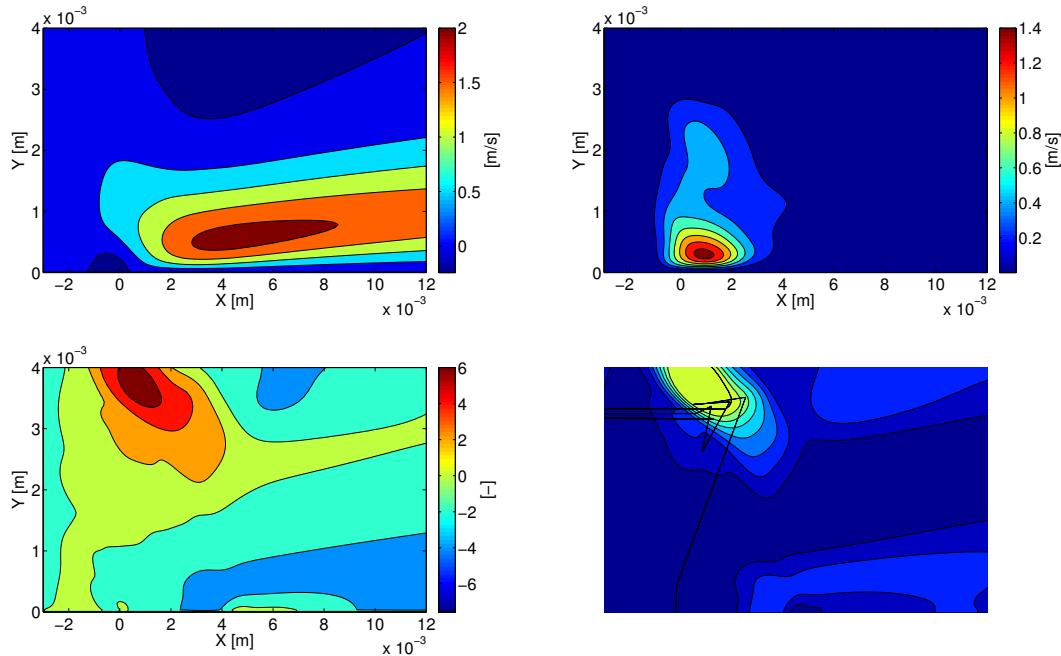


Figure B.29: Uni-variate statistics of prior 6 x-velocity, $T\sigma = 100$; σ^2 (top right), σ_k (bottom left), and k (bottom right)

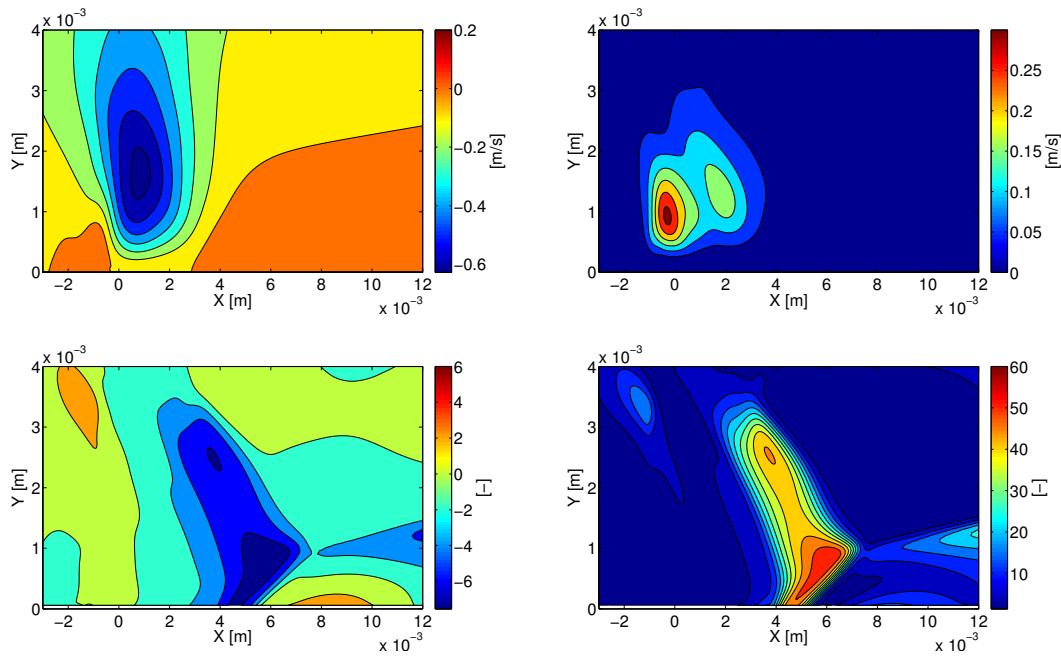


Figure B.30: Uni-variate statistics of prior 6 y-velocity, $T\sigma = 100$; σ^2 (top right), σ_k (bottom left), and k (bottom right)

Appendix C

SSEnKF Results

C.1 Parameter development SSEnKF

For all figures: parameter mean and ± 3 error bars. Lines of constant values are equal to true parameter value.

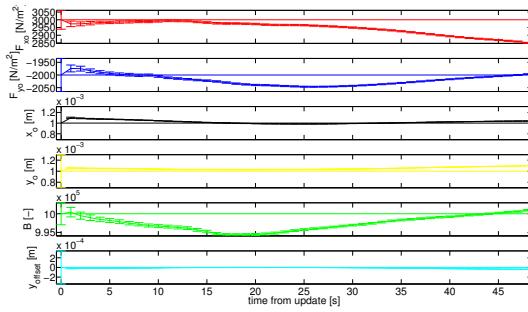


Figure C.1: Steady State EnKF; prior 1a, $T_o = 0.5$ s

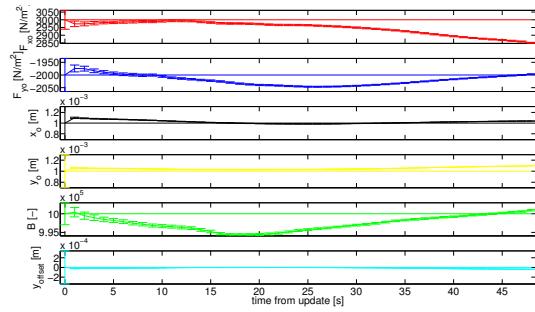


Figure C.2: Steady State EnKF; prior 1a, $T_o = 0.5$ s

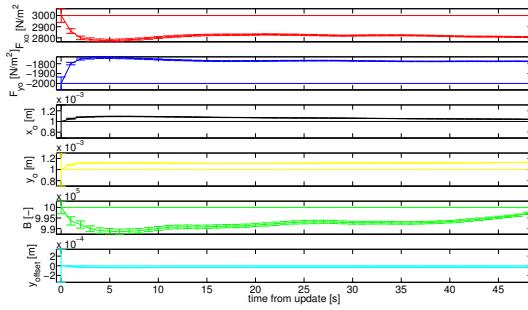


Figure C.3: Steady State EnKF; prior 1a, $T_o = 1.0$ s

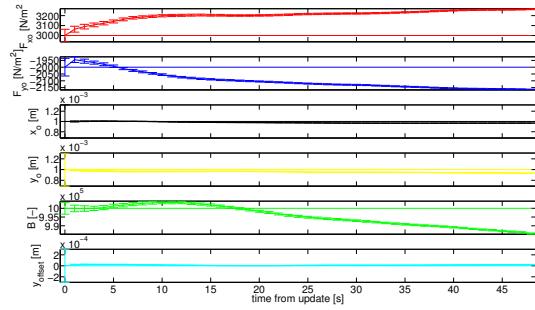


Figure C.4: Steady State EnKF; prior 1b, $T_o = 0.5$ s

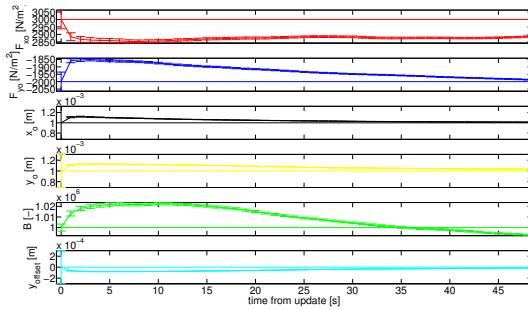


Figure C.5: Steady State EnKF; prior 1b, $T_o = 0.75$ s

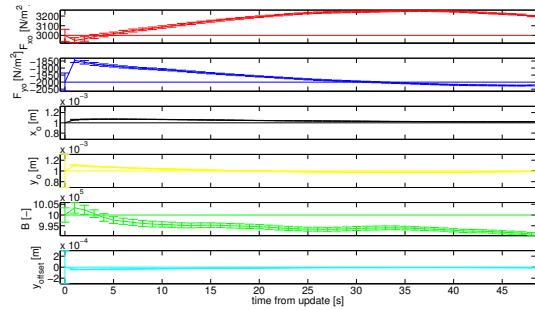


Figure C.6: Steady State EnKF; prior 1b, $T_o = 1.0$ s

C.2 Divergence plots SSEnKF

For all figures: $\frac{RMSE_o}{100} [m=s]$ (black solid line), $\frac{RMSE_f}{100} [N=m^2]$ (black dashed line), $S [m=s]$ (blue line)

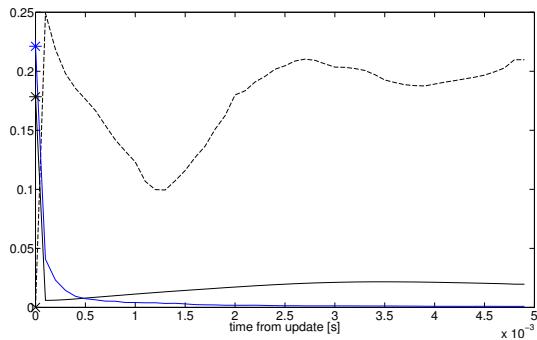


Figure C.7: Steady State EnKF; prior 1a, $T_o = 0.5$ s

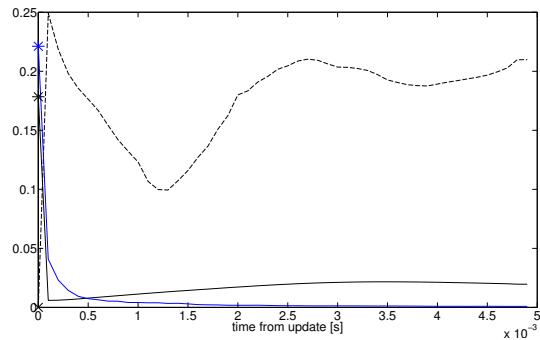


Figure C.8: Steady State EnKF; prior 1a, $T_o = 0.5$ s

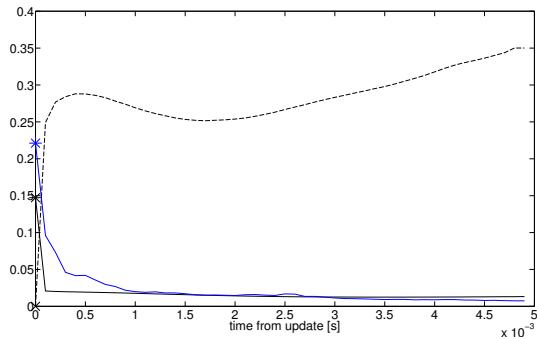


Figure C.9: Steady State EnKF; prior 1a, $T_o = 1.0$ s

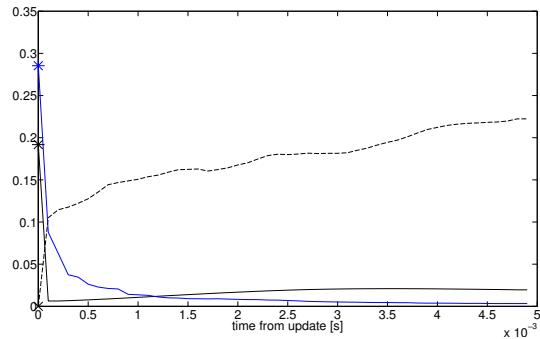


Figure C.10: Steady State EnKF; prior 1b, $T_o = 0.5$ s

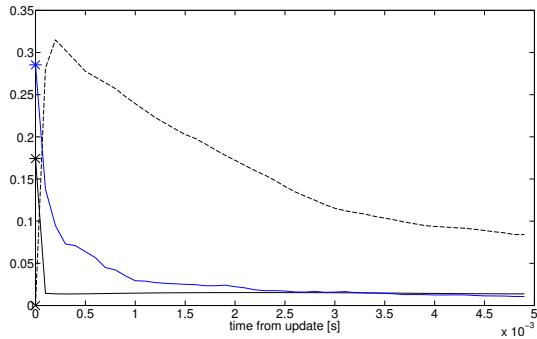


Figure C.11: Steady State EnKF; prior 1b, $T_o = 0.75$ s

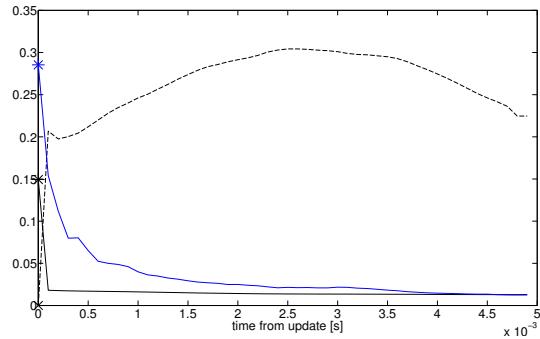


Figure C.12: Steady State EnKF; prior 1b, $T_o = 1.0$ s

Appendix D

Results Twin problem

D.1 Final parameter estimation

Table D.1: Full results of the data assimilation for the twin problem

EnKF		$F_{xo} \left[\frac{N}{m^2} \right]$	$F_{yo} \left[\frac{N}{m^2} \right]$	$x_o \text{ [m]}$	$y_o \text{ [m]}$	$\beta \text{ [-]}$	$y_{offset} \text{ [m]}$	$RMSE_o \text{ [m/s]}$	$RMSE_f \text{ [N/m}^2\text{]}$
Prior 1 v1	μ	3.0126E+03	-1.9926E+03	9.9126E-04	1.0016E-03	1.0140E+06	-8.6250E-07	0.008	4.049
$T_o = 100 \text{ N} = 60$	σ^2	1.9760E-01	1.4520E-01	2.6598E-14	5.3771E-14	6.0442E+04	1.7859E-14	0.008	4.049
Prior 1 v1	μ	3.0086E+03	-1.9714E+03	9.9566E-04	9.9871E-04	1.0053E+06	7.6392E-06	0.009	5.790
$T_o = 100 \text{ N} = 80$	σ^2	2.4677E-01	1.6394E-01	2.9758E-14	7.1843E-14	1.0968E+05	1.9351E-14	0.009	5.790
Prior 1 v1	μ	3.0110E+03	-1.9925E+03	1.0070E-03	1.0084E-03	1.0071E+06	-3.1693E-06	0.003	1.937
$T_o = 50 \text{ N} = 60$	σ^2	1.5782E-01	1.0055E-01	2.4502E-14	3.5011E-14	3.4525E+04	1.7855E-14	0.003	1.937
Prior 1 v1	μ	3.0134E+03	-2.0016E+03	1.0019E-03	9.9965E-04	9.9282E+05	-2.5464E-06	0.004	2.595
$T_o = 50 \text{ N} = 80$	σ^2	2.3165E-01	1.2694E-01	2.3261E-14	4.1728E-14	6.1652E+04	1.4356E-14	0.004	2.595
Prior 1 v1	μ	3.0219E+03	-1.9604E+03	1.0092E-03	1.0146E-03	1.0280E+06	-2.4286E-06	0.004	4.853
$T_o = 75 \text{ N} = 60$	σ^2	2.3247E-01	1.2199E-01	2.3862E-14	4.4165E-14	5.3470E+04	1.8991E-14	0.004	4.853
Prior 1 v1	μ	3.0036E+03	-2.0040E+03	1.0008E-03	1.0048E-03	1.0093E+06	7.2885E-07	0.004	2.185
$T_o = 75 \text{ N} = 80$	σ^2	2.4199E-01	1.7155E-01	3.0374E-14	6.0933E-14	9.1075E+04	1.8546E-14	0.004	2.185
Prior 2 v1	μ	2.9576E+03	-1.9559E+03	1.0140E-03	1.0335E-03	1.0423E+06	-9.7229E-06	0.005	7.125
$T_o = 100 \text{ N} = 60$	σ^2	2.3953E-01	8.5326E-02	1.5488E-14	3.2902E-14	6.4169E+04	1.2549E-14	0.005	7.125
Prior 2 v1	μ	2.9927E+03	-2.0039E+03	1.0011E-03	9.9750E-04	1.0054E+06	1.9699E-06	0.003	1.580
$T_o = 100 \text{ N} = 80$	σ^2	2.3383E-01	9.8953E-02	1.8143E-14	5.0615E-14	9.0497E+04	1.3166E-14	0.003	1.580
Prior 2 v1	μ	2.8715E+03	-1.9488E+03	1.0012E-03	1.0196E-03	1.1094E+06	1.9595E-05	0.022	24.568
$T_o = 50 \text{ N} = 60$	σ^2	1.3492E-01	8.6711E-02	1.9512E-14	2.5494E-14	8.7779E+04	1.4621E-14	0.022	24.568
Prior 2 v1	μ	2.5913E+03	-1.5911E+03	1.0527E-03	1.2864E-03	1.3682E+06	-1.2174E-04	0.040	64.099
$T_o = 50 \text{ N} = 80$	σ^2	1.3128E-01	8.9035E-02	3.0674E-14	4.0438E-14	1.0804E+05	3.0161E-14	0.040	64.099
Prior 2 v1	μ	2.9576E+03	-1.8950E+03	1.0006E-03	1.0160E-03	1.0838E+06	1.7188E-05	0.010	20.948
$T_o = 75 \text{ N} = 60$	σ^2	1.5152E-01	7.7782E-02	2.1362E-14	3.7214E-14	7.0260E+04	1.7004E-14	0.010	20.948
Prior 2 v1	μ	2.9867E+03	-1.9919E+03	9.9854E-04	9.9970E-04	1.0117E+06	2.1420E-06	0.002	2.890
$T_o = 75 \text{ N} = 80$	σ^2	2.1477E-01	1.2624E-01	2.0730E-14	4.9359E-14	1.2212E+05	1.3312E-14	0.002	2.890
Prior 2 v2	μ	2.9449E+03	-1.9959E+03	9.9632E-04	1.0023E-03	1.0404E+06	4.1915E-06	0.004	8.552
$T_o = 100 \text{ N} = 60$	σ^2	1.7010E-01	1.1922E-01	1.9125E-14	6.0237E-14	1.8917E+05	1.4610E-14	0.004	8.552
Prior 2 v2	μ	3.0037E+03	-1.9928E+03	1.0066E-03	1.0013E-03	1.0154E+06	2.9780E-07	0.003	2.753
$T_o = 100 \text{ N} = 80$	σ^2	4.0158E-01	1.4695E-01	2.5300E-14	7.0459E-14	1.4912E+05	1.8054E-14	0.003	2.753
Prior 2 v3	μ	2.9500E+03	-1.9325E+03	1.0071E-03	1.0270E-03	1.0435E+06	-3.1447E-06	0.006	10.366
$T_o = 100 \text{ N} = 60$	σ^2	1.3912E-01	8.0790E-02	1.8592E-14	4.0800E-14	7.8559E+04	1.5028E-14	0.006	10.366
Prior 2 v3	μ	2.9591E+03	-1.9811E+03	9.9351E-04	1.0062E-03	1.0326E+06	3.4626E-06	0.004	8.570
$T_o = 100 \text{ N} = 80$	σ^2	1.6867E-01	9.1939E-02	1.5762E-14	4.2789E-14	7.9750E+04	1.1621E-14	0.004	8.570
Prior 2 v4	μ	1.8817E+03	-1.4033E+03	1.4518E-03	1.2822E-03	1.4527E+06	-7.2040E-04	0.071	257.651
$T_o = 100 \text{ N} = 60$	σ^2	5.0274E-02	3.7219E-02	2.7110E-14	4.1940E-13	6.9151E+04	2.5921E-15	0.071	257.651
Prior 2 v4	μ	1.9301E+03	-1.4226E+03	1.4367E-03	1.8658E-04	1.5610E+06	-7.3515E-04	0.065	315.000
$T_o = 100 \text{ N} = 80$	σ^2	5.6770E-02	4.9009E-02	2.9501E-14	9.4091E-10	1.4016E+05	3.8264E-15	0.065	315.000
Prior 2 v4	μ	1.9092E+03	-1.3054E+03	1.3878E-03	1.5295E-03	1.8003E+06	-7.3679E-04	0.064	209.278
$T_o = 50 \text{ N} = 60$	σ^2	4.5157E-02	4.2306E-02	2.5204E-14	1.5446E-13	6.5838E+04	2.6033E-15	0.064	209.278
Prior 2 v4	μ	1.8623E+03	-1.3998E+03	1.4913E-03	1.5138E-03	1.5578E+06	-7.4178E-04	0.066	217.634
$T_o = 50 \text{ N} = 80$	σ^2	7.0020E-02	5.3906E-02	3.9057E-14	8.9619E-13	1.2959E+05	3.1739E-15	0.066	217.634
Prior 2 v4	μ	1.9560E+03	-1.2984E+03	1.4224E-03	1.5566E-03	1.5011E+06	-7.1658E-04	0.073	192.946
$T_o = 75 \text{ N} = 60$	σ^2	3.9796E-02	7.6137E-02	3.4668E-14	8.7189E-14	1.4488E+05	3.4406E-15	0.073	192.946
Prior 2 v4	μ	1.9192E+03	-1.4794E+03	1.4937E-03	1.8532E-03	1.4701E+06	-7.3902E-04	0.064	138.691
$T_o = 75 \text{ N} = 80$	σ^2	6.1603E-02	5.2989E-02	3.3285E-14	5.8849E-12	1.4510E+05	4.7200E-15	0.064	138.691
Prior 3 v1	μ	2.3524E+03	-1.6909E+03	1.0535E-03	-1.1895E-03	1.0967E+06	-8.8688E-05	0.041	52.633
$T_o = 100 \text{ N} = 60$	σ^2	7.6604E-02	4.3389E-02	3.6882E-14	5.3482E-14	6.3320E+04	2.8755E-14	0.041	52.633
Prior 3 v1	μ	3.5268E+02	-1.8864E+03	-3.2204E-04	-6.1261E-03	2.2331E+05	-4.2472E-03	0.208	533.971
$T_o = 100 \text{ N} = 80$	σ^2	7.8437E-03	9.2543E-02	2.8175E-13	1.0702E-12	6.2228E+03	8.3967E-13	0.208	533.971
Prior 3 v1	μ	2.7167E+03	-1.7432E+03	1.0895E-03	-1.1546E-03	1.3289E+06	-3.2218E-05	0.027	45.184
$T_o = 50 \text{ N} = 60$	σ^2	1.6083E-01	9.4547E-02	2.4813E-14	4.8973E-14	7.7887E+04	2.8666E-14	0.027	45.184
Prior 3 v1	μ	4.0748E+02	5.5028E+02	-1.6810E-05	-2.4781E-03	2.3173E+06	-1.5272E-03	0.182	324.725
$T_o = 50 \text{ N} = 80$	σ^2	3.8001E-02	3.9107E-01	3.4473E-13	2.4614E-13	2.7505E+04	2.0753E-13	0.182	324.725
Prior 3 v1	μ	3.2136E+02	-9.8504E+02	-4.4417E-03	-5.3945E-03	1.9941E+06	-2.0576E-03	0.226	343.884
$T_o = 75 \text{ N} = 60$	σ^2	7.3170E-03	2.0774E-01	2.4792E-13	9.5731E-13	8.2131E+04	7.9772E-13	0.226	343.884
Prior 3 v1	μ	1.8947E+03	-1.5769E+03	1.1281E-03	-1.2773E-03	1.0890E+06	-1.9027E-04	0.076	83.671
$T_o = 75 \text{ N} = 80$	σ^2	5.9622E-02	5.4352E-02	5.7474E-14	6.6276E-14	1.1323E+05	5.6980E-14	0.076	83.671
Prior 3 v2	μ	2.5951E+03	-1.8941E+03	1.1122E-03	-1.2301E-03	1.0671E+06	-1.5788E-04	0.023	40.507
$T_o = 100 \text{ N} = 60$	σ^2	1.6759E-01	1.5848E-01	4.1973E-14	8.9051E-14	7.4635E+04	4.7538E-14	0.023	40.507
Prior 3 v2	μ	4.1481E+02	-2.0845E+03	-9.1139E-04	-7.2871E-03	1.3167E+05	-5.2345E-03	0.130	669.516
$T_o = 100 \text{ N} = 80$	σ^2	1.2666E-02	2.5872E-01	3.7841E-13	1.5691E-12	4.7785E+04	1.2736E-12	0.130	669.516

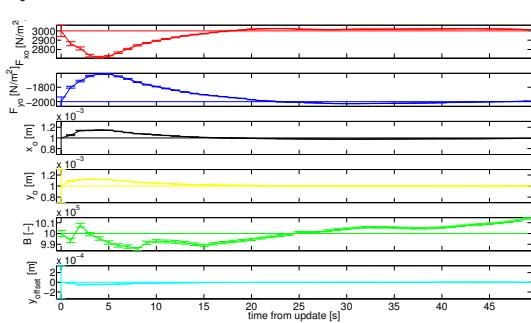
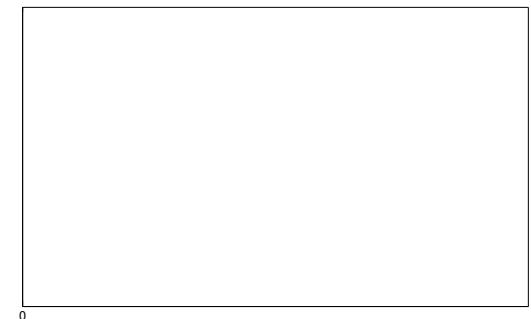
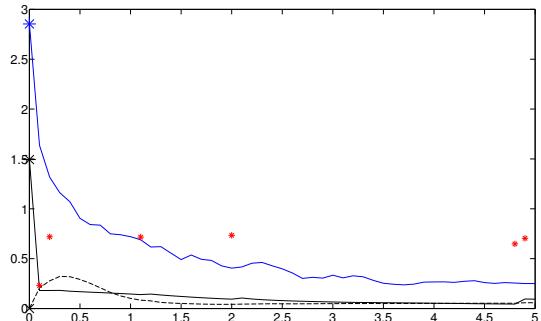
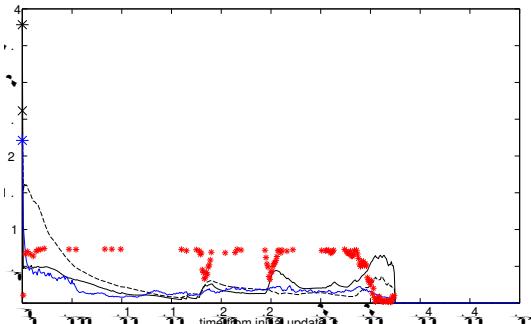
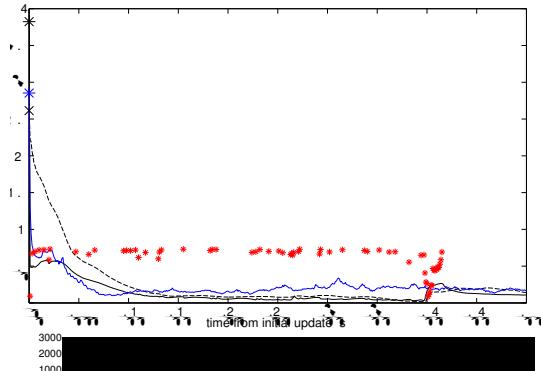
Prior 3 v3	μ	7.6817E+02	-1.1405E+03	2.9400E-04	-2.5454E-03	2.7230E+06	-8.3383E-04	0.229	265.275
$T_o = 100$ N = 60	σ^2	8.9783E-03	1.7343E-02	6.3906E-14	1.7901E-13	6.0144E+04	1.5477E-13	0.229	265.275
Prior 3 v3	μ	2.7366E+02	-5.9511E+02	-5.7645E-04	-3.1276E-03	1.6896E+06	-9.3452E-04	0.277	307.879
$T_o = 100$ N = 80	σ^2	3.4396E-03	5.3000E-02	1.9679E-13	6.0344E-13	1.7995E+05	5.7465E-13	0.277	307.879
Prior 3 v4	μ	2.8857E+03	-1.9939E+03	9.7170E-04	-4.1513E-03	1.1148E+06	2.6620E-05	0.015	664.764
$T_o = 100$ N = 60	σ^2	8.8599E-02	9.5540E-02	1.0278E-14	6.2386E-11	3.2938E+04	3.8782E-15	0.015	664.764
Prior 3 v4	μ	7.3174E+02	-3.2438E+02	-2.8493E-03	-2.2077E-03	1.6603E+06	4.6819E-04	0.294	323.263
$T_o = 100$ N = 80	σ^2	3.2266E-02	7.6441E-02	5.3392E-14	2.2804E-12	7.9822E+04	5.5821E-15	0.294	323.263
Prior 3 v4	μ	3.3400E+03	-1.4522E+03	1.0075E-03	-5.3874E-03	1.1588E+06	7.9396E-05	0.064	405.652
$T_o = 50$ N = 60	σ^2	1.1942E-01	7.0626E-02	1.0717E-14	6.3044E-12	5.0777E+04	2.9213E-15	0.064	405.652
Prior 3 v4	μ	5.7458E+03	8.5216E+03	-6.0923E-03	-2.6760E-03	1.7852E+06	-4.8789E-03	0.205	312.025
$T_o = 50$ N = 80	σ^2	2.9163E+01	2.7725E+01	4.6033E-11	2.1486E-12	7.4540E+04	7.8348E-13	0.205	312.025
Prior 3 v4	μ	-5.6000E+01	-2.5367E+02	-3.9510E-03	-3.0828E-03	2.4598E+06	2.3585E-04	0.248	313.535
$T_o = 75$ N = 60	σ^2	1.0053E-02	4.1836E-02	2.1049E-14	1.2505E-12	8.8239E+04	1.7616E-14	0.248	313.535
Prior 3 v4	μ	4.3007E+03	-1.8925E+03	-5.9310E-03	-5.2030E-03	1.4291E+06	-2.3841E-03	0.263	576.891
$T_o = 75$ N = 80	σ^2	9.3670E+00	6.7844E+01	1.3068E-12	2.7281E-12	1.1097E+05	2.1173E-13	0.263	576.891
Prior 4 v1	μ	2.9155E+03	-2.0510E+03	9.9356E-04	-9.9678E-04	9.9961E+05	1.1458E-06	0.009	8.705
$T_o = 100$ N = 60	σ^2	1.6886E-01	1.1002E-01	2.0260E-14	5.2720E-14	9.8281E+04	1.2194E-14	0.009	8.705
Prior 4 v1	μ	1.5036E+03	-7.7034E+02	1.3676E-03	-2.1003E-03	2.3833E+06	-7.5776E-04	0.129	179.548
$T_o = 50$ N = 60	σ^2	2.4245E-02	4.9130E-02	4.7008E-14	7.9321E-14	5.4885E+04	7.7393E-14	0.129	179.548
Prior 4 v1	μ	3.2955E+01	-5.0317E+02	-6.5733E-04	-2.7411E-03	1.9234E+06	-1.0242E-03	0.251	300.940
$T_o = 75$ N = 60	σ^2	7.1748E-03	3.6014E-02	4.9535E-14	8.2932E-14	2.6944E+04	6.9638E-14	0.251	300.940
Prior 5 v1	μ	-1.6337E+03	1.4553E+03	-1.6740E-03	-1.3823E-03	1.9992E+06	-3.1568E-04	0.235	389.002
$T_o = 100$ N = 60	σ^2	2.2914E-01	6.5543E-02	2.6934E-14	2.0905E-14	3.8293E+02	1.7040E-14	0.235	389.002
Prior 5 v1	μ	-2.0181E+03	3.5575E+02	-1.6175E-03	-1.7485E-03	2.0116E+06	-6.2235E-04	0.228	387.637
$T_o = 50$ N = 60	σ^2	1.5466E-01	2.9731E-02	1.9487E-14	2.4768E-14	2.8609E+02	1.8531E-14	0.228	387.637
Prior 5 v1	μ	-1.7616E+03	1.1270E+03	-1.3312E-03	-1.7522E-03	2.0136E+06	-6.8686E-04	0.228	414.931
$T_o = 75$ N = 60	σ^2	9.8658E-02	1.0489E-02	1.0770E-14	8.8741E-15	5.2772E+01	6.2561E-15	0.228	414.931
Prior 6 v1	μ	2.9861E+03	-1.9961E+03	1.0016E-03	1.0021E-03	1.0129E+06	-1.1805E-06	0.003	2.294
$T_o = 100$ N = 60	σ^2	1.7765E-01	8.0858E-02	2.5528E-14	6.6592E-14	5.9803E+04	2.3536E-14	0.003	2.294
Prior 6 v1	μ	2.5587E+03	-1.2237E+03	1.0865E-03	1.4075E-03	1.9967E+06	-1.6323E-04	0.059	111.683
$T_o = 50$ N = 60	σ^2	9.9123E-02	8.9494E-02	4.7685E-14	5.4417E-14	2.3645E+04	5.0181E-14	0.059	111.683
Prior 6 v1	μ	2.9319E+03	-1.9767E+03	9.9833E-04	1.0160E-03	1.0254E+06	-4.3207E-06	0.007	7.708
$T_o = 75$ N = 60	σ^2	1.6539E-01	8.5067E-02	2.4722E-14	5.2229E-14	6.2541E+04	2.0842E-14	0.007	7.708

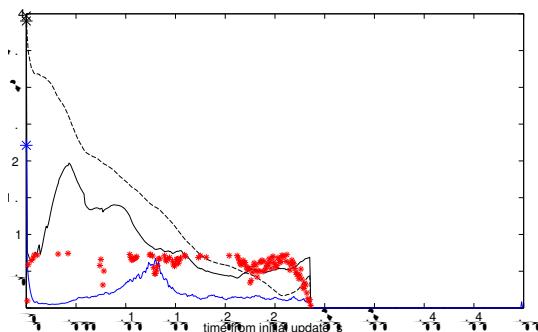
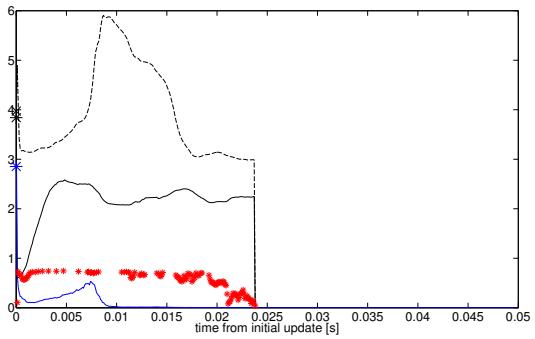
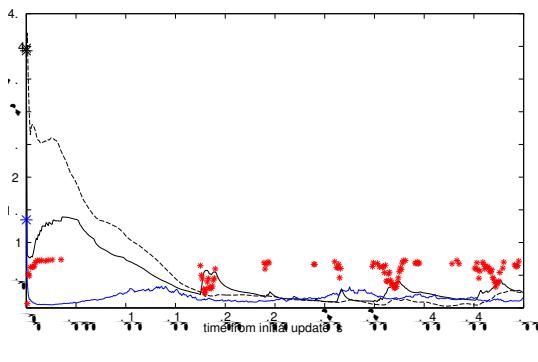
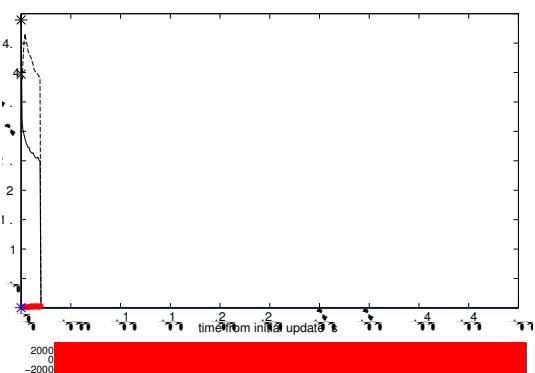
DEnKF									
Prior 1 v1	μ	2.9922E+03	-1.9746E+03	1.0033E-03	1.0108E-03	1.0294E+06	5.7322E-07	0.006	4.998
$T_o = 100$ N = 60	σ^2	3.7347E-01	2.7014E-01	5.0880E-14	9.1718E-14	1.1719E+05	3.0586E-14	0.006	4.998
Prior 1 v1	μ	3.0187E+03	-1.9812E+03	1.0017E-03	1.0050E-03	1.0046E+06	-1.1039E-06	0.003	2.686
$T_o = 100$ N = 80	σ^2	4.3770E-01	3.0294E-01	4.9447E-14	1.0045E-13	1.8252E+05	3.3691E-14	0.003	2.686
Prior 1 v1	μ	2.9918E+03	-2.0037E+03	1.0047E-03	1.0021E-03	9.9773E+05	-2.8339E-06	0.004	2.309
$T_o = 50$ N = 60	σ^2	5.0752E-01	3.2925E-01	5.9137E-14	1.0139E-13	1.5361E+05	3.9233E-14	0.004	2.309
Prior 1 v1	μ	3.0040E+03	-1.9928E+03	1.0035E-03	1.0036E-03	1.0014E+06	-2.8019E-06	0.002	1.477
$T_o = 50$ N = 80	σ^2	5.4466E-01	3.7996E-01	6.2751E-14	1.1246E-13	2.2540E+05	4.0597E-14	0.002	1.477
Prior 1 v1	μ	3.0208E+03	-1.9792E+03	1.0024E-03	1.0064E-03	9.9825E+05	-3.3690E-06	0.003	3.093
$T_o = 75$ N = 60	σ^2	4.6925E-01	3.0933E-01	5.6440E-14	1.0528E-13	1.5430E+05	3.5287E-14	0.003	3.093
Prior 1 v1	μ	2.9907E+03	-1.9893E+03	1.0037E-03	1.0075E-03	1.0120E+06	-2.1271E-06	0.003	1.707
$T_o = 75$ N = 80	σ^2	4.7600E-01	3.4520E-01	5.7180E-14	1.1021E-13	2.0589E+05	3.8457E-14	0.003	1.707
Prior 2 v1	μ	2.9711E+03	-1.9882E+03	9.9982E-04	1.0069E-03	1.0389E+06	2.6306E-06	0.003	6.694
$T_o = 100$ N = 60	σ^2	3.0176E-01	1.5395E-01	3.0833E-14	6.9019E-14	9.5072E+04	2.3051E-14	0.003	6.694
Prior 2 v1	μ	2.9848E+03	-1.9853E+03	1.0062E-03	1.0068E-03	1.0186E+06	-1.5973E-06	0.002	2.793
$T_o = 100$ N = 80	σ^2	3.5242E-01	1.8313E-01	3.4555E-14	7.5159E-14	1.1908E+05	2.3378E-14	0.002	2.793
Prior 2 v1	μ	2.9728E+03	-1.9681E+03	1.0439E-03	1.0412E-03	1.0428E+06	-1.8253E-05	0.010	10.131
$T_o = 50$ N = 60	σ^2	3.1764E-01	1.5975E-01	3.4249E-14	6.6874E-14	1.4865E+05	3.0396E-14	0.010	10.131
Prior 2 v1	μ	2.9386E+03	-1.9281E+03	1.0262E-03	1.0500E-03	1.0410E+06	-1.5542E-05	0.008	9.189
$T_o = 50$ N = 80	σ^2	3.4863E-01	2.1464E-01	4.3704E-14	7.6838E-14	1.6716E+05	3.7389E-14	0.008	9.189
Prior 2 v1	μ	2.9967E+03	-1.9991E+03	1.0024E-03	1.0019E-03	1.0056E+06	-5.1548E-07	0.001	0.747
$T_o = 75$ N = 60	σ^2	3.1478E-01	1.6736E-01	3.1471E-14	7.1058E-14	1.2824E+05	2.2124E-14	0.001	0.747
Prior 2 v1	μ	2.9881E+03	-1.9903E+03	1.0047E-03	1.0089E-03	1.0121E+06	-2.9797E-06	0.002	1.673
$T_o = 75$ N = 80	σ^2	3.8720E-01	1.9069E-01	3.7429E-14	7.6135E-14	1.3454E+05	2.7808E-14	0.002	1.673
Prior 2 v2	μ	2.9862E+03	-1.9967E+03	1.0053E-03	1.0084E-03	1.0531E+06	-1.2390E-07	0.003	6.181
$T_o = 100$ N = 60	σ^2	4.3758E-01	2.5353E-01	4.2038E-14	9.0952E-14	1.6623E+05	3.3361E-14	0.003	6.181
Prior 2 v2	μ	3.0052E+03	-1.9998E+03	1.0024E-03	9.9971E-04	1.0112E+06	-1.0721E-07	0.002	1.844
$T_o = 100$ N = 80	σ^2	4.9118E-01	2.9041E-01	4.9936E-14	1.1272E-13	2.2727E+05	3.5569E-14	0.002	1.844
Prior 2 v3	μ	2.9400E+03	-1.9162E+03	1.0162E-03	1.0447E-03	1.0569E+06	-8.4135E-06	0.007	12.516

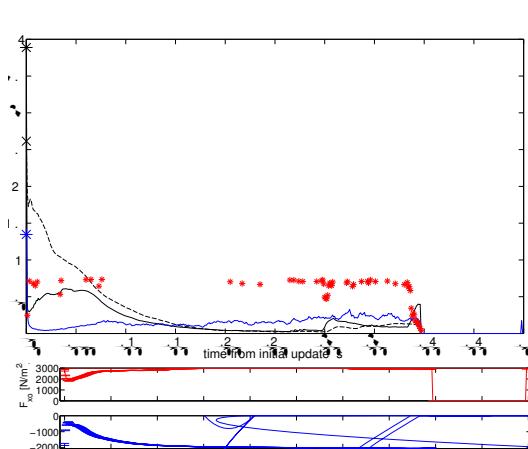
$T_o = 100$	$N = 60$	σ^2	2.5529E-01	1.1495E-01	2.9186E-14	6.1024E-14	7.2659E+04	2.4036E-14	0.007	12.516
Prior 2 v3	μ	2.9703E+03	-1.9846E+03	1.0010E-03	1.0061E-03	1.0205E+06	8.1813E-07	0.002	4.501	
$T_o = 100$	$N = 80$	σ^2	2.9200E-01	1.4651E-01	2.8608E-14	6.6064E-14	8.7642E+04	1.9678E-14	0.002	4.501
Prior 2 v4	μ	1.9164E+03	-1.4235E+03	1.4663E-03	1.5063E-03	1.2174E+06	-7.5654E-04	0.062	212.439	
$T_o = 100$	$N = 60$	σ^2	9.1053E-02	1.2439E-01	6.6259E-14	8.6844E-13	1.4430E+05	8.5693E-15	0.062	212.439
Prior 2 v4	μ	1.9284E+03	-1.4131E+03	1.4441E-03	1.4145E-03	1.2683E+06	-7.5219E-04	0.062	231.542	
$T_o = 100$	$N = 80$	σ^2	9.2183E-02	1.2274E-01	5.4803E-14	1.7775E-12	1.5855E+05	7.1540E-15	0.062	231.542
Prior 2 v4	μ	1.8851E+03	-1.3873E+03	1.3424E-03	1.2270E-03	1.9125E+06	-7.4134E-04	0.058	274.340	
$T_o = 50$	$N = 60$	σ^2	1.3271E-01	1.9490E-01	7.3312E-14	5.8337E-13	2.9058E+05	7.1594E-15	0.058	274.340
Prior 2 v4	μ	1.9477E+03	-1.3454E+03	1.3686E-03	1.3620E-03	1.4189E+06	-7.2742E-04	0.056	234.902	
$T_o = 50$	$N = 80$	σ^2	1.4456E-01	1.9575E-01	8.7073E-14	1.1084E-12	4.3621E+05	8.2194E-15	0.056	234.902
Prior 2 v4	μ	1.9401E+03	-1.4599E+03	1.4420E-03	1.4280E-03	1.2537E+06	-7.5056E-04	0.059	228.070	
$T_o = 75$	$N = 60$	σ^2	1.0042E-01	1.7380E-01	7.3789E-14	4.9468E-13	2.3095E+05	8.2857E-15	0.059	228.070
Prior 2 v4	μ	1.8184E+03	-1.4806E+03	1.4691E-03	1.5241E-03	1.4208E+06	-7.6207E-04	0.058	215.881	
$T_o = 75$	$N = 80$	σ^2	1.3113E-01	1.3806E-01	6.6877E-14	1.7684E-12	4.2263E+05	8.7664E-15	0.058	215.881
Prior 3 v1	μ	2.9082E+03	-1.9680E+03	9.9401E-04	-1.0069E-03	1.0392E+06	5.9220E-06	0.006	12.623	
$T_o = 100$	$N = 60$	σ^2	2.7285E-01	1.4869E-01	3.2622E-14	6.9626E-14	9.3320E+04	2.4297E-14	0.006	12.623
Prior 3 v1	μ	7.1237E+02	-2.3536E+03	-2.0068E-03	-6.7496E-03	1.2504E+06	-5.0396E-03	0.140	457.745	
$T_o = 100$	$N = 80$	σ^2	2.0349E-02	6.5169E-01	8.3338E-13	1.3812E-12	3.5265E+05	1.2766E-12	0.140	457.745
Prior 3 v1	μ	1.0421E+03	-9.7704E+02	5.2418E-04	-1.3354E-03	2.2113E+06	4.6023E-05	0.108	240.069	
$T_o = 50$	$N = 60$	σ^2	1.3406E-01	1.6254E-01	1.2020E-13	1.6847E-13	2.8300E+05	9.9833E-14	0.108	240.069
Prior 3 v1	μ	2.2868E+03	-1.4355E+03	1.0596E-03	-1.3089E-03	1.6376E+06	-1.0717E-04	0.036	97.816	
$T_o = 50$	$N = 80$	σ^2	2.1592E-01	1.6574E-01	7.9844E-14	1.0753E-13	4.8037E+05	9.1269E-14	0.036	97.816
Prior 3 v1	μ	2.7501E+02	-1.5503E+02	-3.2687E-03	-2.5015E-03	2.6126E+06	4.6036E-04	0.172	313.582	
$T_o = 75$	$N = 60$	σ^2	1.9169E-02	2.1875E-01	1.0185E-12	3.0684E-12	7.2713E+05	2.6944E-12	0.172	313.582
Prior 3 v1	μ	2.9851E+03	-1.9889E+03	1.0013E-03	-1.0016E-03	1.0146E+06	2.0835E-06	0.002	3.032	
$T_o = 75$	$N = 80$	σ^2	3.5421E-01	1.9419E-01	3.4571E-14	7.6417E-14	1.7257E+05	2.5135E-14	0.002	3.032
Prior 3 v2	μ	2.9719E+03	-1.9790E+03	1.0028E-03	-1.0043E-03	1.0456E+06	1.0070E-06	0.004	7.350	
$T_o = 100$	$N = 60$	σ^2	4.0601E-01	2.9271E-01	5.3829E-14	1.0301E-13	1.8197E+05	4.3955E-14	0.004	7.350
Prior 3 v2	μ	4.3537E+02	-1.1533E+03	-5.2146E-04	-5.4132E-03	1.2868E+06	-3.9312E-03	0.105	289.617	
$T_o = 100$	$N = 80$	σ^2	2.8460E-02	1.0397E+00	6.3879E-12	1.2188E-11	2.2099E+06	1.1590E-11	0.105	289.617
Prior 3 v3	μ	2.8300E+03	-1.9128E+03	1.0101E-03	-1.0545E-03	1.0715E+06	-9.0199E-06	0.012	20.779	
$T_o = 100$	$N = 60$	σ^2	1.9533E-01	9.7657E-02	3.1531E-14	5.7339E-14	7.2424E+04	2.6217E-14	0.012	20.779
Prior 3 v3	μ	5.5636E+02	-1.5425E+03	-1.3984E-03	-5.7613E-03	1.5861E+06	-4.1062E-03	0.184	338.234	
$T_o = 100$	$N = 80$	σ^2	2.4974E-02	4.7670E-01	5.1037E-13	8.0857E-13	2.5855E+05	7.3022E-13	0.184	338.234
Prior 3 v4	μ	6.2025E+02	4.2596E+02	-3.1066E-04	-1.7435E-04	1.4428E+06	-9.8063E-05	0.172	310.270	
$T_o = 100$	$N = 60$	σ^2	9.6817E-02	3.0617E-01	2.5000E-13	6.0692E-12	8.2383E+05	5.9308E-14	0.172	310.270
Prior 3 v4	μ	3.6186E+01	4.2428E+02	-2.5977E-03	2.5548E-03	1.1479E+06	-3.2749E-04	0.155	319.581	
$T_o = 100$	$N = 80$	σ^2	7.2640E-02	2.1668E-01	1.3504E-12	2.1379E-11	2.0170E+06	1.6949E-13	0.155	319.581
Prior 3 v4	μ	6.3455E+02	-1.7165E+02	-1.3382E-03	-3.2388E-03	1.8691E+06	2.3382E-04	0.149	322.286	
$T_o = 50$	$N = 60$	σ^2	7.3763E-02	1.4842E-01	5.5049E-14	8.6181E-12	2.4599E+05	9.2321E-15	0.149	322.286
Prior 3 v4	μ	7.6520E+03	-2.3639E+03	-6.9729E-03	-1.6416E-02	1.5240E+06	-1.5578E-02	0.174	1347.644	
$T_o = 50$	$N = 80$	σ^2	3.7606E+02	3.0987E+02	8.7309E-10	2.7946E-10	1.0425E+06	3.7381E-10	0.174	1347.644
Prior 3 v4	μ	-1.7865E+03	3.5743E+03	-5.7623E-04	-7.9261E-03	2.8949E+06	-8.4950E-03	0.187	345.874	
$T_o = 75$	$N = 60$	σ^2	7.2798E+01	4.4856E+02	7.1822E-11	5.5531E-11	1.1326E+06	1.0329E-10	0.187	345.874
Prior 3 v4	μ	1.4800E+04	-9.4108E+03	-1.8852E-02	-5.8287E-03	1.5216E+06	-7.3195E-03	0.149	312.018	
$T_o = 75$	$N = 80$	σ^2	6.0430E+02	8.9723E+02	2.1683E-10	2.7776E-11	1.9086E+06	1.7441E-11	0.149	312.018
Prior 4 v1	μ	2.9223E+03	-1.9530E+03	1.0202E-03	-1.0408E-03	1.0508E+06	-1.2596E-05	0.007	8.634	
$T_o = 100$	$N = 60$	σ^2	2.8345E-01	1.7482E-01	3.5700E-14	6.8658E-14	1.3532E+05	3.0264E-14	0.007	8.634
Prior 4 v1	μ	8.8545E+02	2.8991E+02	1.0043E-03	-3.3435E-03	1.7109E+06	-2.0498E-03	0.138	278.316	
$T_o = 50$	$N = 60$	σ^2	4.4286E-02	2.7224E-01	5.2946E-13	5.3867E-13	4.3652E+04	4.8070E-13	0.138	278.316
Prior 4 v1	μ	2.4068E+03	-1.2764E+03	1.3138E-03	-1.4034E-03	1.0742E+06	-2.5741E-04	0.075	84.612	
$T_o = 75$	$N = 60$	σ^2	1.0151E-01	1.1262E-01	5.3250E-14	8.1023E-14	1.3878E+05	6.0515E-14	0.075	84.612
Prior 5 v1	μ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.953	312.025	
$T_o = 100$	$N = 60$	σ^2	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.953	312.025
Prior 5 v1	μ	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.953	312.025	
$T_o = 75$	$N = 60$	σ^2	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.953	312.025
Prior 6 v1	μ	2.9957E+03	-1.9838E+03	1.0052E-03	1.0075E-03	1.0179E+06	-2.6688E-06	0.003	2.322	
$T_o = 100$	$N = 60$	σ^2	3.0478E-01	1.5169E-01	3.2427E-14	6.6411E-14	9.2324E+04	2.1561E-14	0.003	2.322
Prior 6 v1	μ	2.9593E+03	-2.0765E+03	1.0316E-03	1.0193E-03	9.6376E+05	-1.3325E-05	0.016	15.525	
$T_o = 50$	$N = 60$	σ^2	2.2225E-01	1.2199E-01	3.3465E-14	6.1278E-14	6.2235E+04	2.4834E-14	0.016	15.525
Prior 6 v1	μ	2.9434E+03	-1.9950E+03	1.0101E-03	1.0147E-03	1.0400E+06	-3.3534E-06	0.005	6.000	
$T_o = 75$	$N = 60$	σ^2	2.8609E-01	1.5868E-01	3.2039E-14	7.1410E-14	1.2415E+05	2.6664E-14	0.005	6.000

D.2 Parameter estimation development and divergence plots

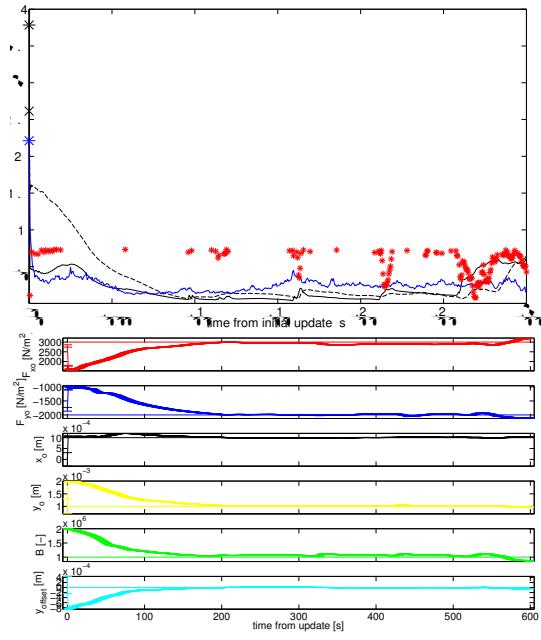
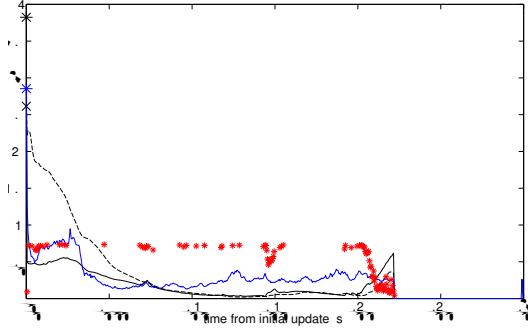
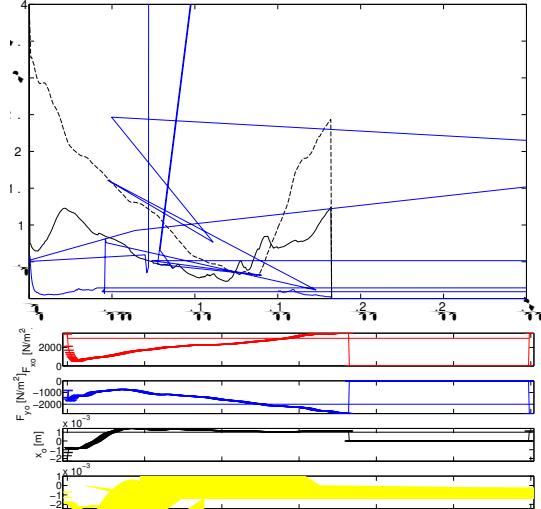
Only results using initial priors with $T_o = 1.0$ s are shown. SSSEnKF results; TOP: $RMSE_o * 100[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{10}[m=s]$ (* blue line). BOTTOM: parameter mean and ± 3 error bars. Lines of constant values are equal to true parameter value.

**Figure D.1:** EnKF: prior 1v1, $N = 60$ **Figure D.2:** EnKF: prior 1v1, $N = 80$ **Figure D.3:** EnKF: prior 2v1, $N = 60$ **Figure D.4:** EnKF: prior 2v1, $N = 80$

**Figure D.5:** EnKF: prior 3v1, $N = 60$ **Figure D.6:** EnKF: prior 3v1, $N = 80$ **Figure D.7:** EnKF: prior 4v1, $N = 60$ **Figure D.8:** EnKF: prior 5v1, $N = 60$



\$50 \times 10^{-4}

Figure D.9: EnKF: prior 6v1, $N = 60$ **Figure D.10:** EnKF: prior 2v2, $N = 60$ **Figure D.11:** EnKF: prior 2v2, $N = 80$ **Figure D.12:** EnKF: prior 3v2, $N = 60$

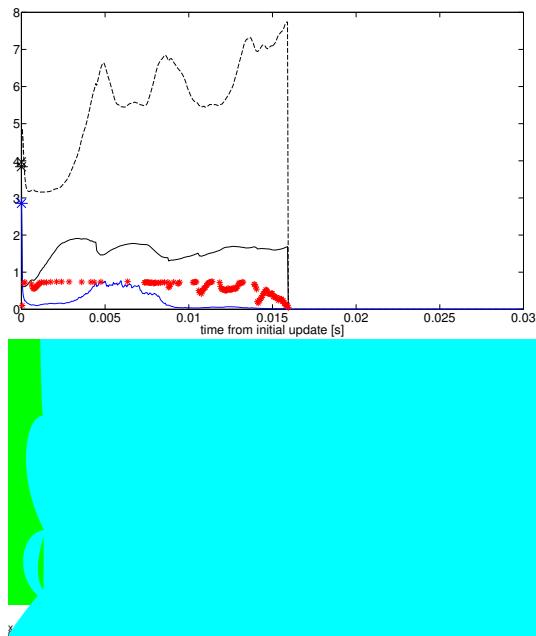


Figure D.13: EnKF: prior 3v2, $N = 80$

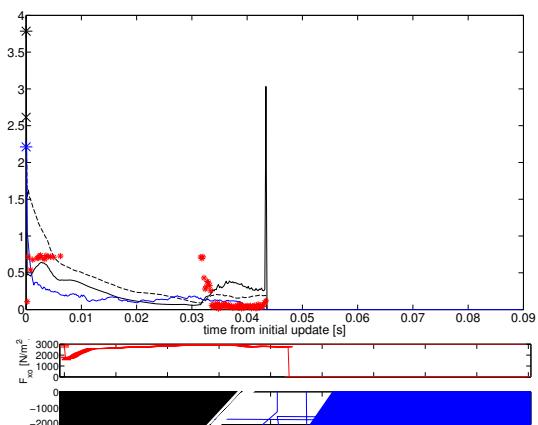


Figure D.14: EnKF: prior 2v3, $N = 60$

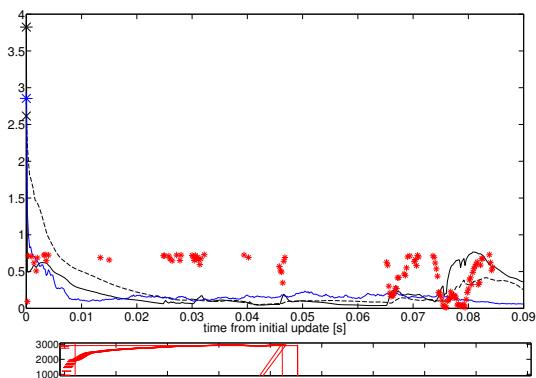


Figure D.15: EnKF: prior 2v3, $N = 80$

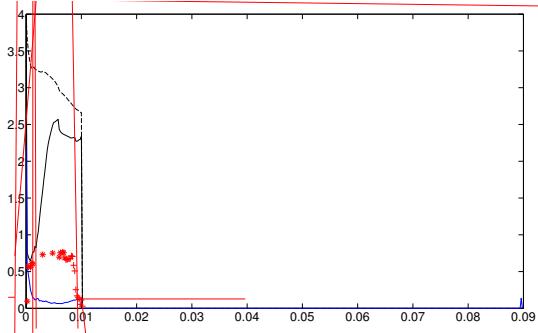


Figure D.16: EnKF: prior 3v3, $N = 60$

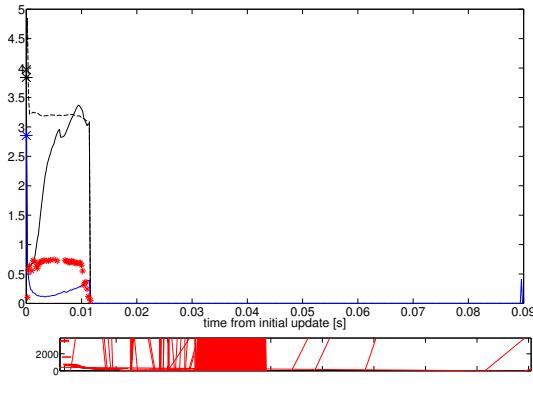


Figure D.17: EnKF: prior 3v3, $N = 80$

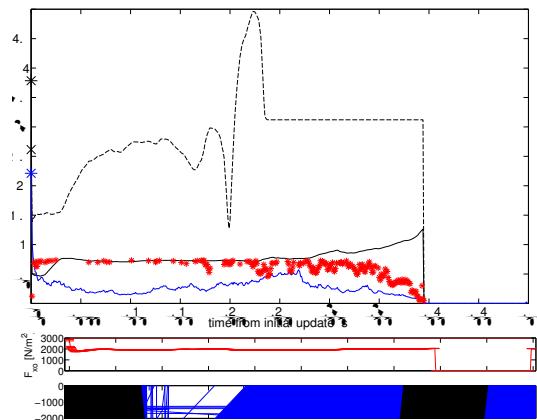


Figure D.18: EnKF: prior 2v4, $N = 60$

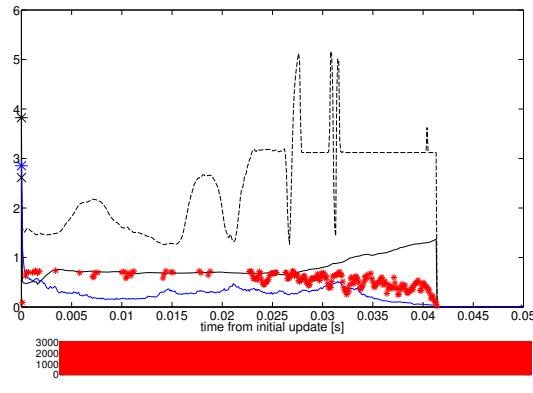


Figure D.19: EnKF: prior 2v4, $N = 80$

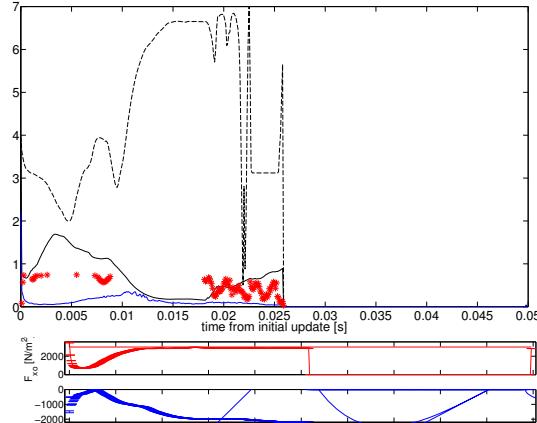


Figure D.20: EnKF: prior 3v4, $N = 60$

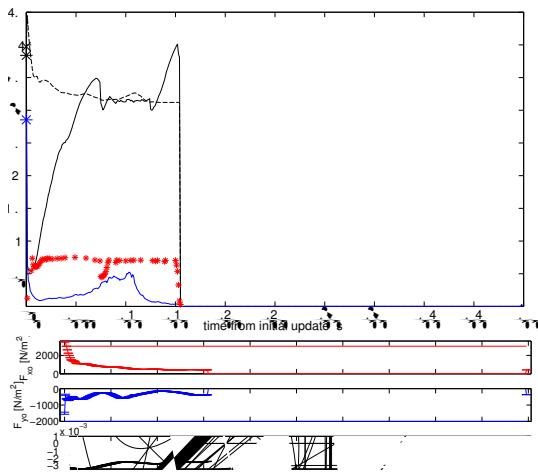


Figure D.21: EnKF: prior 3v4, $N = 80$

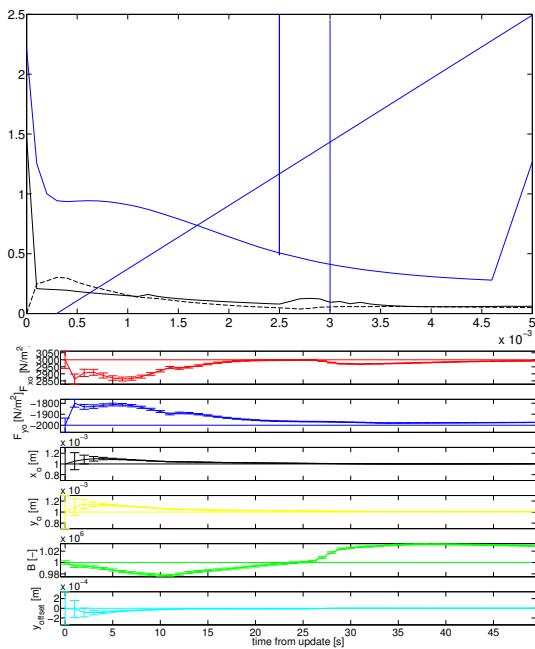


Figure D.22: DEnKF: prior 1v1,roblem46 9.9626 T78.64843 0 Td N1,robl1520 9.9626 Tf1.85943 0 Td =1,ro 0 50.85 0 -

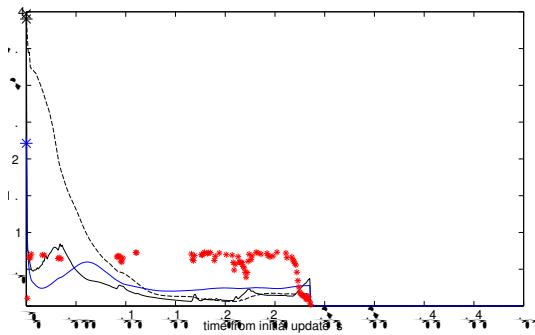


Figure D.26: DEnKF: prior 3v1, $N = 60$

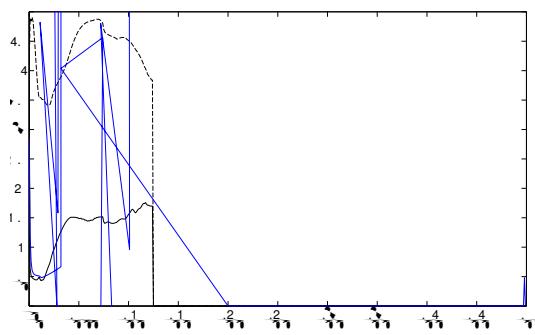


Figure D.27: DEnKF: prior 3v1, $N = 80$

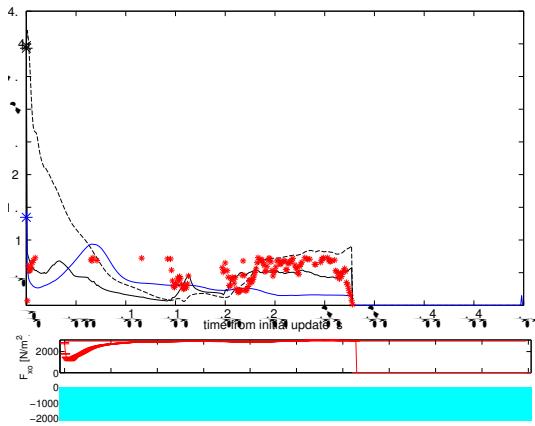


Figure D.28: DEnKF: prior 4v1, $N = 60$

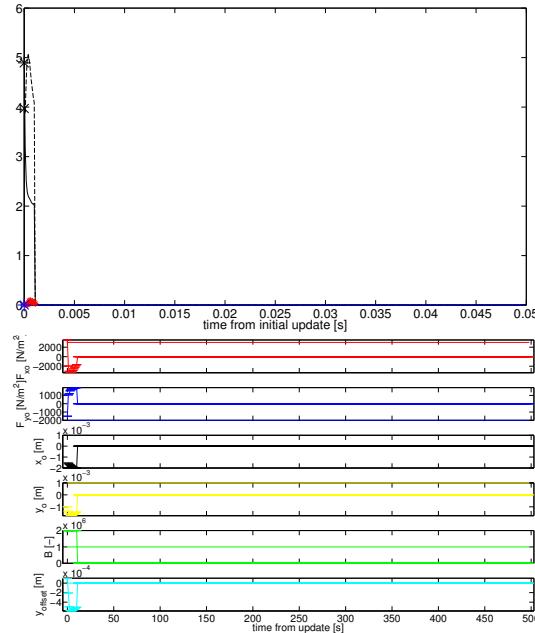


Figure D.29: DEnKF: prior 5v1, $N = 60$

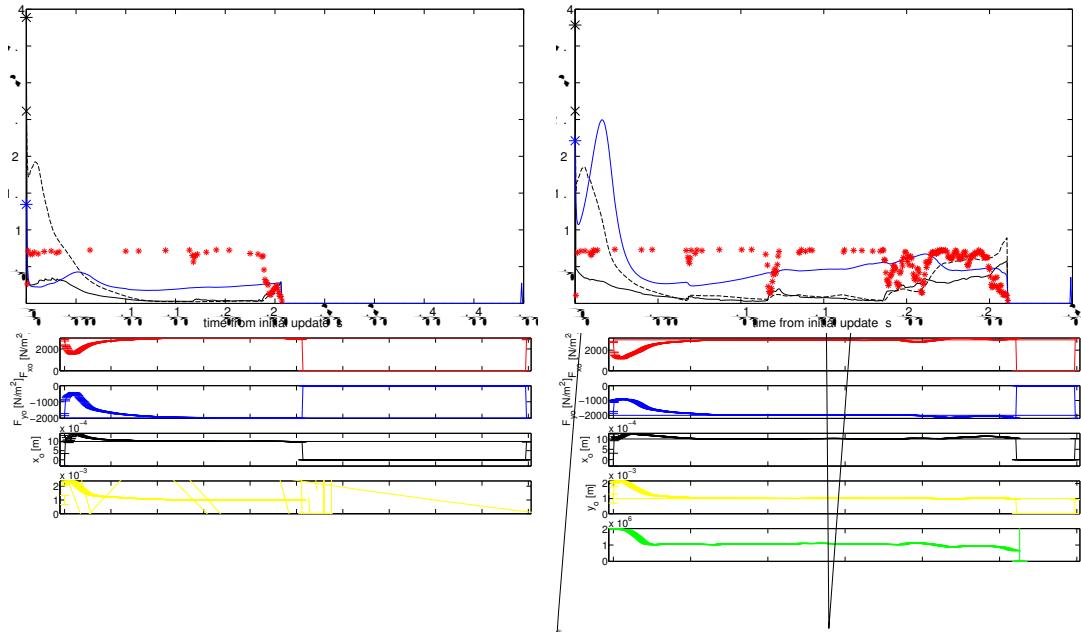


Figure D.30: DEnKF: prior 6v1, $N = 60$

Figure D.31: DEnKF: prior 2v2, $N = 60$

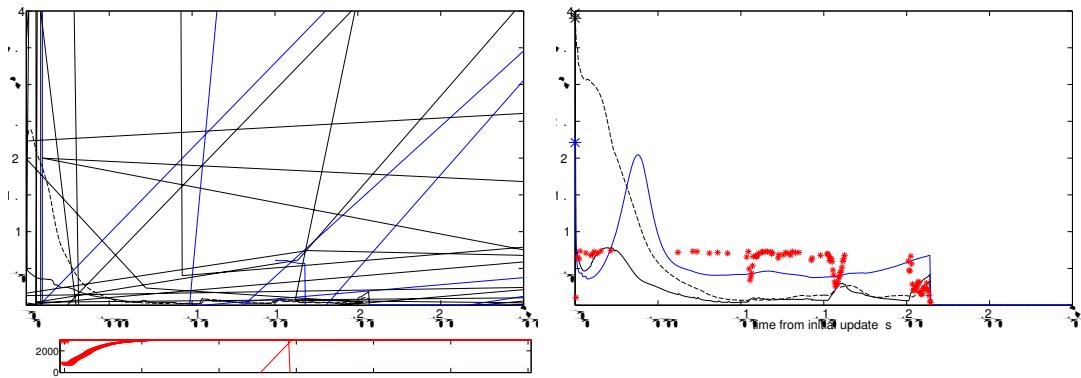


Figure D.32: DEnKF: prior 2v2, $N = 80$

Figure D.33: DEnKF: prior 3v2, $N = 60$

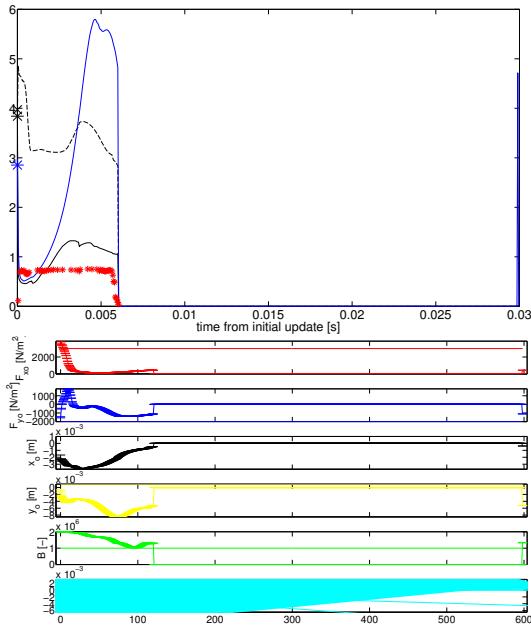


Figure D.34: DEnKF: prior 3v2, $N = 80$

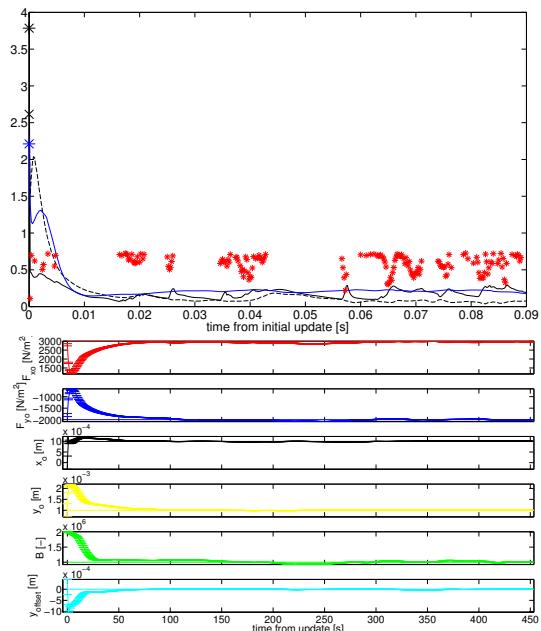


Figure D.35: DEnKF: prior 2v3, $N = 60$

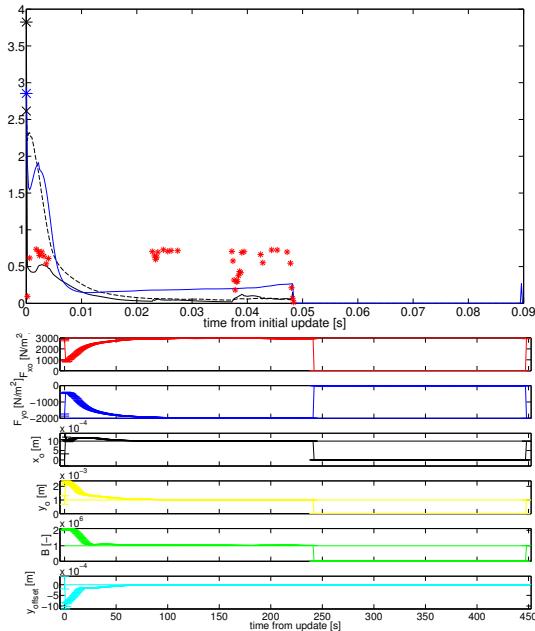


Figure D.36: DEnKF: prior 2v3, $N = 80$

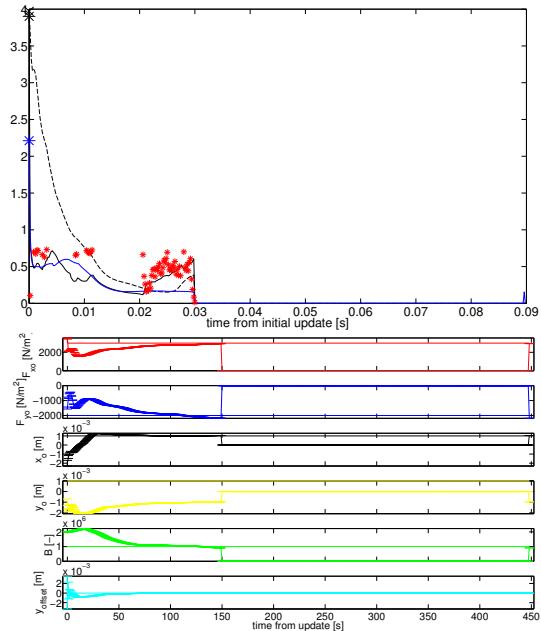


Figure D.37: DEnKF: prior 3v3, $N = 60$

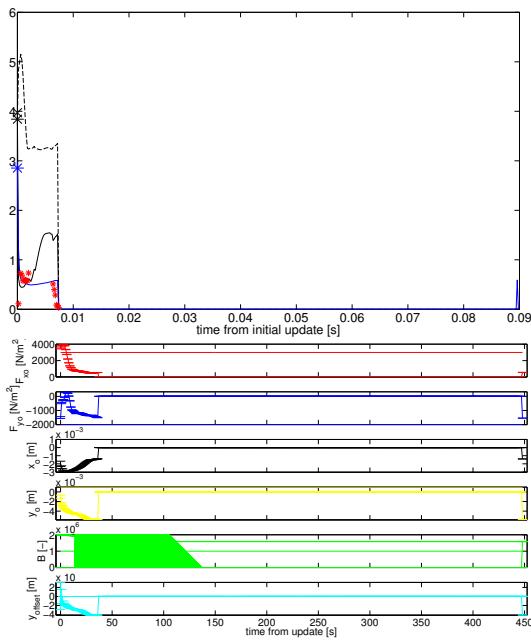


Figure D.38: DEnKF: prior 3v3, $N = 80$

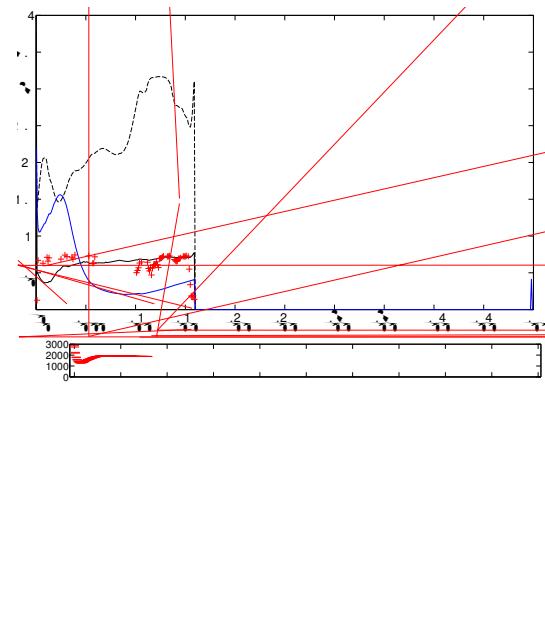


Figure D.39: DEnKF: prior 2v4, $N = 60$

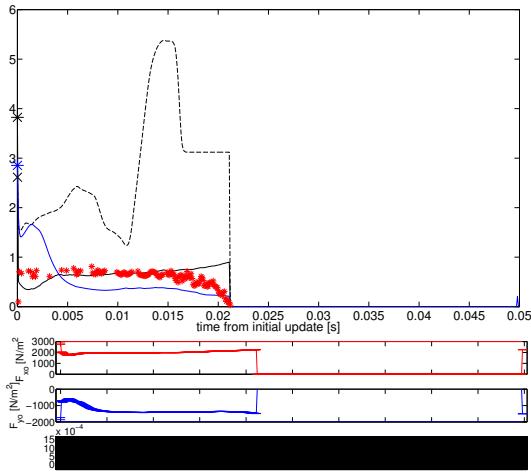


Figure D.40: DEnKF: prior 2v4, $N = 80$

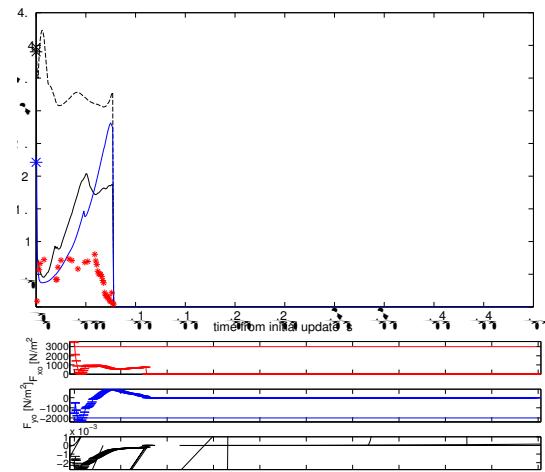


Figure D.41: DEnKF: prior 3v4, $N = 60$

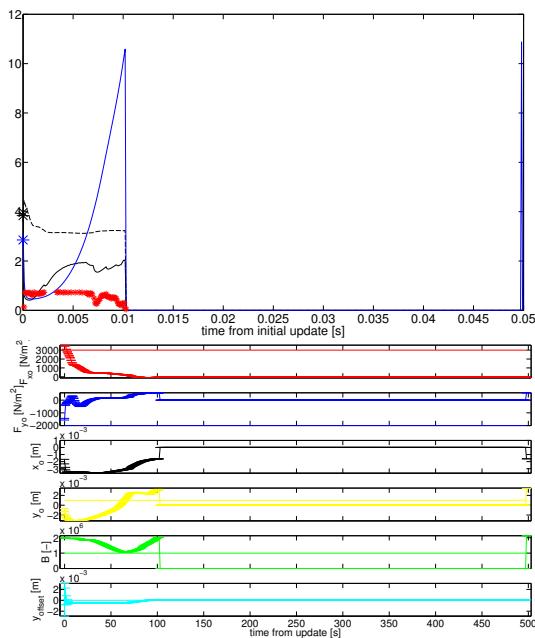


Figure D.42: DEnKF: prior 3v4, $N = 80$

Appendix E

PIV Itering with DEnKF results

E.1 Parameter estimation results

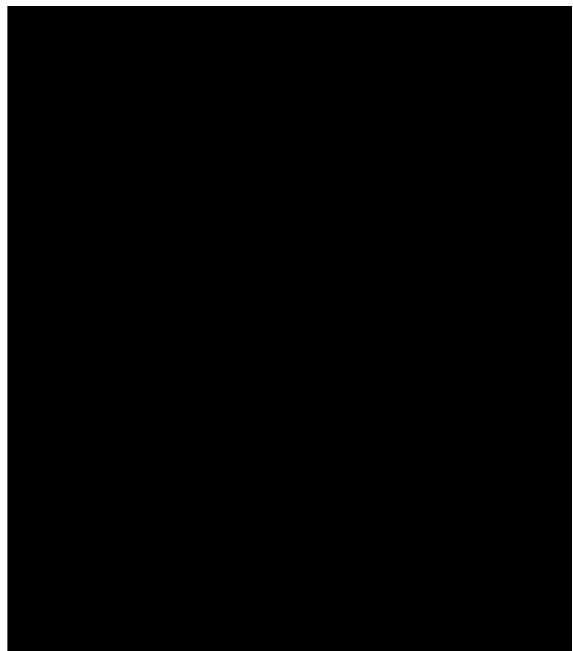
Table E.1: Results DEnKF with PIV data, case 1-4

Prior #	Case 1						$RMSE_o$	Tx [N]	Ty [N]
	$F_{xo} \left[\frac{N}{m^2} \right]$	$F_{yo} \left[\frac{N}{m^2} \right]$	x_o [m]	y_o [m]	β [-]	y_{offset} [m]			
2	μ	9.63E+02	-9.73E+02	1.01E-03	-9.31E-04	1.29E+07	-6.23E-04	1.98E-02	7.58E-01
	σ^2	5.34E-01	6.67E+00	7.82E-13	5.13E-13	6.65E+08	3.65E-13	-	-7.66E-01
3	μ	1.10E+03	-1.29E+03	7.56E-04	7.62E-04	2.80E+06	-7.97E-04	2.02E-02	1.08E+00
	σ^2	7.98E+00	3.94E+01	4.21E-12	3.04E-12	1.28E+07	5.54E-13	-	-1.27E+00
4	μ	-1.30E+04	-7.66E+03	7.21E-03	6.50E-03	2.60E+06	-7.95E-03	3.22E-02	-9.63E-02
	σ^2	4.02E+03	7.52E+04	4.48E-10	4.04E-10	1.96E+08	3.50E-10	-	-5.66E-02
	Case 2								
	$F_{xo} \left[\frac{N}{m^2} \right]$	$F_{yo} \left[\frac{N}{m^2} \right]$	x_o [m]	y_o [m]	β [-]	y_{offset} [m]	$RMSE_o$	Tx [N]	Ty [N]
1	μ	1.48E+03	-1.30E+03	1.25E-03	-1.18E-03	4.57E+06	-8.81E-04	5.06E-02	2.29E+00
	σ^2	5.70E-01	4.53E+00	5.17E-13	4.87E-13	2.52E+07	2.54E-13	-	-2.00E+00
3	μ	-4.72E+02	9.65E+02	3.69E-03	-1.90E-03	2.85E+04	-1.04E-03	7.24E-02	-2.31E+01
	σ^2	6.84E-02	1.30E-01	8.93E-13	1.50E-12	6.59E+03	8.94E-13	-	4.73E+01
5	μ	1.47E+03	-1.37E+03	1.15E-03	1.15E-03	3.82E+06	-8.71E-04	5.14E-02	2.37E+00
	σ^2	5.56E-01	5.02E+00	6.65E-13	5.20E-13	1.16E+07	3.00E-13	-	-2.22E+00
6	μ	2.20E+03	3.91E+03	1.32E-03	5.86E-04	6.21E+05	-1.34E-03	5.50E-02	3.22E+00
	σ^2	3.06E+00	3.05E+01	8.65E-13	1.16E-12	2.81E+05	2.59E-13	-	5.74E+00
	Case 3								
	$F_{xo} \left[\frac{N}{m^2} \right]$	$F_{yo} \left[\frac{N}{m^2} \right]$	x_o [m]	y_o [m]	β [-]	y_{offset} [m]	$RMSE_o$	Tx [N]	Ty [N]
10	μ	2.65E+03	6.27E+03	1.85E-03	1.09E-03	3.20E+05	-2.28E-03	1.28E-01	7.41E+00
	σ^2	4.19E+00	5.66E+01	1.23E-12	1.59E-12	3.63E+04	2.62E-13	-	1.75E+01
5	μ	1.20E+03	-1.78E+03	3.01E-03	3.26E-04	1.62E+06	3.82E-03	2.08E-01	1.53E+00
	σ^2	2.06E+00	2.63E+00	7.13E-14	1.58E-14	9.59E+05	2.88E-14	-	-2.27E+00
6	μ	5.30E+02	3.02E+02	2.39E-03	-1.68E-05	1.37E+06	1.04E-03	1.91E-01	6.14E-01
	σ^2	6.50E-01	1.29E+00	4.17E-13	1.44E-12	3.27E+06	1.70E-13	-	3.50E-01
8	μ	6.63E+02	2.33E+02	1.61E-03	1.10E-05	1.24E+06	1.18E-03	1.93E-01	8.66E-01
	σ^2	2.25E-01	3.53E-01	2.86E-13	4.57E-14	3.58E+05	1.01E-13	-	3.05E-01
	Case 4								
	$F_{xo} \left[\frac{N}{m^2} \right]$	$F_{yo} \left[\frac{N}{m^2} \right]$	x_o [m]	y_o [m]	β [-]	y_{offset} [m]	$RMSE_o$	Tx [N]	Ty [N]
9	μ	9.04E+02	1.65E+03	5.22E-03	4.27E-03	2.11E+06	-3.78E-03	2.76E-01	7.22E+00
	σ^2	6.99E-02	1.82E+01	1.96E-11	2.29E-11	9.98E+05	2.29E-11	-	1.32E+01
7	μ	1.18E+03	-9.04E+03	2.47E-03	7.16E-03	7.55E+05	-7.62E-03	2.83E-01	9.87E+00
	σ^2	5.64E-01	1.37E+02	1.10E-11	7.44E-12	2.75E+05	8.09E-12	-	-7.59E+01
8	μ	1.86E+03	-1.79E+03	2.82E-03	2.97E-03	3.26E+06	-2.42E-03	2.30E-01	9.04E+00
	σ^2	2.37E-01	6.28E+00	1.13E-12	8.31E-13	2.22E+06	8.04E-13	-	-8.71E+00

Table E.2: Results DEnKF with PIV data, case 5-8

Prior #	μ	Case 5						$RMSE_o$	Tx [N]	Ty [N]	
		F_{xo} [$\frac{N}{m^2}$]	F_{yo} [$\frac{N}{m^2}$]	x_o [m]	y_o [m]	β [-]	y_{offset} [m]				
9	σ^2	2.26E+03	-1.87E+03	4.45E-03	4.08E-03	2.46E+06	-3.50E-03	3.52E-01	1.70E+01	-1.41E+01	
		2.07E-01	4.30E+01	4.46E-12	2.32E-12	8.12E+05	2.30E-12	-	-	-	
7	μ	5.48E+03	-1.65E+04	3.20E-04	3.36E-03	9.17E+04	-5.87E-03	3.74E-01	5.62E+01	-1.69E+02	
		1.43E+01	2.34E+02	4.57E-12	3.39E-12	2.78E+03	2.89E-12	-	-	-	
8	μ	2.31E+03	-2.71E+03	4.95E-03	4.93E-03	2.67E+06	-4.31E-03	3.57E-01	2.06E+01	-2.41E+01	
		2.14E-01	7.22E+00	1.27E-12	1.15E-12	7.81E+05	1.11E-12	0	0	0	
		Case 6						$RMSE_o$	Tx [mN]	Ty [mN]	
		F_{xo} [$\frac{N}{m^2}$]	F_{yo} [$\frac{N}{m^2}$]	x_o [m]	y_o [m]	β [-]	y_{offset} [m]				
1	μ	8.92E+02	-5.21E+04	1.09E+00	8.22E-01	1.78E+06	-8.21E-01	6.56E-02	5.36E+00	-3.13E+02	
		2.60E+00	1.93E+04	4.69E-06	1.38E-06	1.18E+07	1.38E-06	-	-	-	
2	μ	4.68E+03	-2.90E+03	-1.70E-04	1.51E-04	4.61E+05	-1.68E-03	5.16E-02	3.30E+00	-2.04E+00	
		1.03E+01	2.65E+02	3.13E-12	4.12E-14	6.22E+04	3.56E-13	-	-	-	
3	μ	-8.44E+04	-2.42E+06	5.15E-03	2.83E-03	3.06E+08	-2.94E-03	6.86E-02	-1.30E-01	-3.74E+00	
		1.84E+04	3.33E+06	4.21E-12	1.16E-11	1.73E+10	1.17E-11	-	-	-	
6	μ	8.18E+02	-8.34E+02	1.20E-03	-1.56E-03	1.16E+06	-1.61E-03	5.30E-02	2.39E+00	-2.43E+00	
		1.47E+00	1.09E+01	4.04E-12	2.55E-12	7.46E+05	6.41E-13	-	-	-	
		Case 7						$RMSE_o$	Tx [mN]	Ty [mN]	
		F_{xo} [$\frac{N}{m^2}$]	F_{yo} [$\frac{N}{m^2}$]	x_o [m]	y_o [m]	β [-]	y_{offset} [m]				
9	μ	3.95E+03	-1.12E+04	-1.82E-04	-3.16E-04	5.75E+05	-1.22E-03	9.06E-02	4.67E+00	-1.33E+01	
		8.75E-01	2.06E+01	2.79E-13	5.02E-14	4.65E+04	6.72E-14	-	-	-	
5	μ	2.46E+03	3.84E+03	1.29E-03	6.71E-04	2.95E+05	-2.02E-03	1.05E-01	5.78E+00	9.02E+00	
		1.71E+00	2.13E+01	1.12E-12	1.62E-12	5.05E+04	4.07E-13	-	-	-	
6	μ	8.51E+03	-1.65E+04	-4.13E-04	-9.85E-05	4.30E+05	-1.71E-03	8.22E-02	6.00E+00	-1.16E+01	
		3.58E+00	8.67E+01	7.51E-13	2.32E-14	1.52E+04	2.02E-13	-	-	-	
		Case 8						$RMSE_o$	Tx [N]	Ty [N]	
		F_{xo} [$\frac{N}{m^2}$]	F_{yo} [$\frac{N}{m^2}$]	x_o [m]	y_o [m]	β [-]	y_{offset} [m]				
5	μ	9.88E+02	-9.43E+02	-4.31E-04	2.50E-03	2.67E+06	1.20E-03	2.73E-01	4.75E+00	-4.54E+00	
		1.29E-01	1.76E-01	3.77E-13	5.04E-13	3.86E+06	6.54E-13	-	-	-	
7	μ	2.64E+03	-2.67E+04	-9.99E-05	3.44E-03	6.38E+05	-4.36E-03	1.90E-01	7.48E+00	-7.57E+01	
		5.20E+00	1.08E+03	1.98E-12	1.18E-12	5.82E+04	1.13E-12	-	-	-	
8	μ	3.10E+03	-3.87E+03	4.60E-04	-5.32E-04	8.64E+05	-9.03E-04	1.99E-01	4.57E+00	-5.70E+00	
		1.23E+01	1.56E+01	4.04E-13	4.24E-13	4.89E+05	8.45E-14	-	-	-	

E.2 Force field plots all cases



θ Tr

Figure E.1: Force field contours for case 1



Figure E.2: Force field contours for case 2

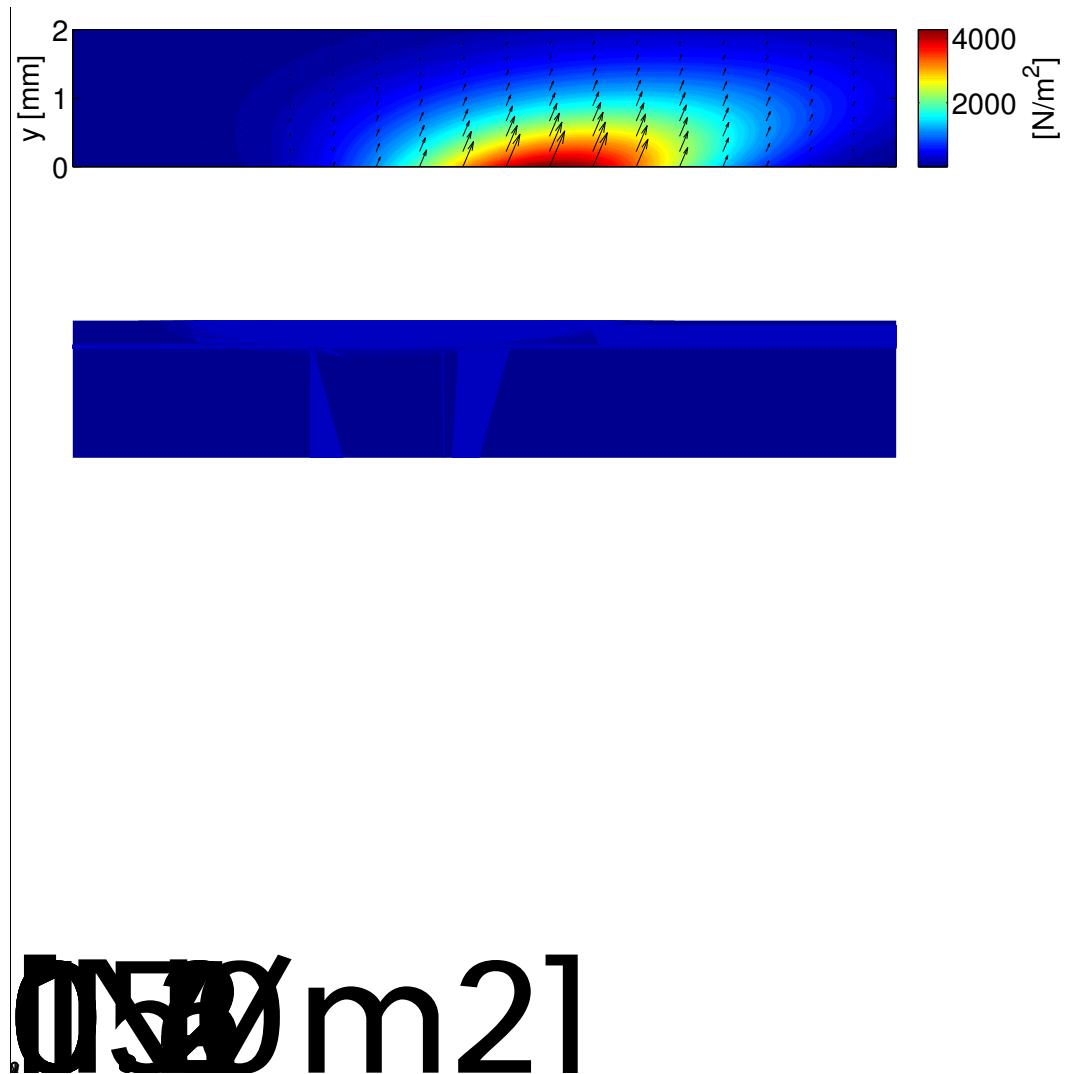


Figure E.3: Force field contours for case 3



Figure E.4: Force field contours for case 4



Figure E.5: Force field contours for case 5

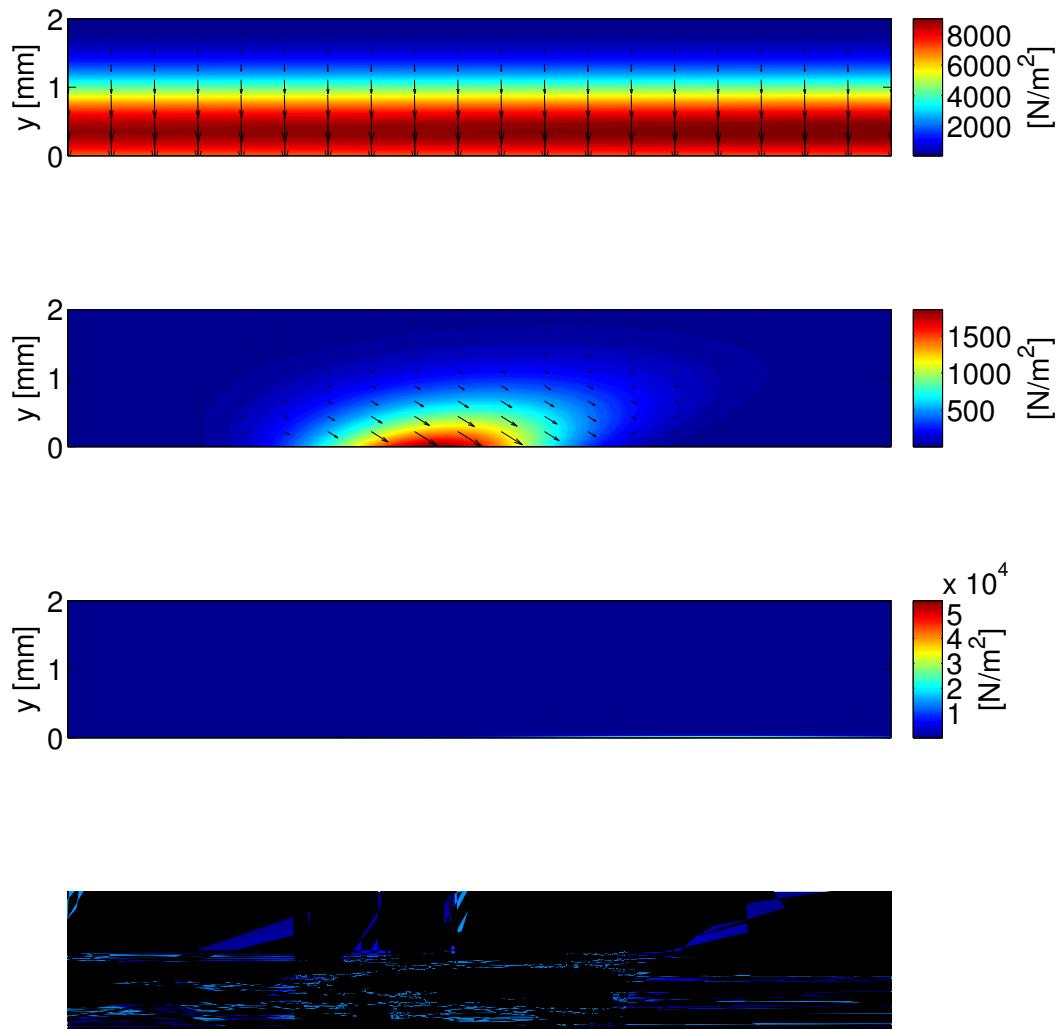


Figure E.6: Force field contours for case 6

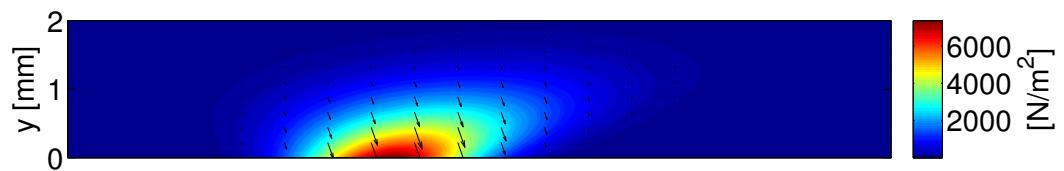


Figure E.7: Force field contours for case 7

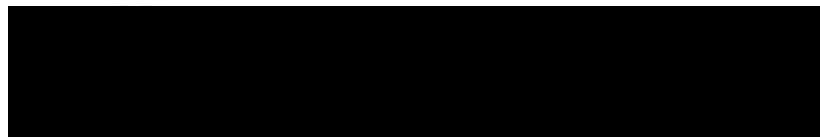
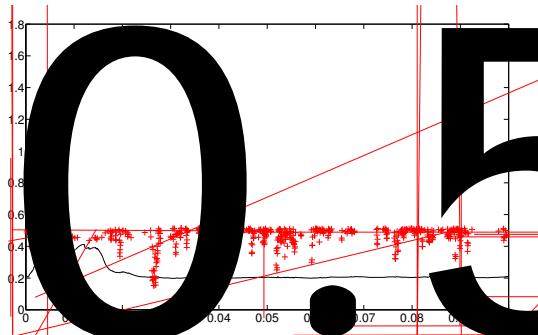
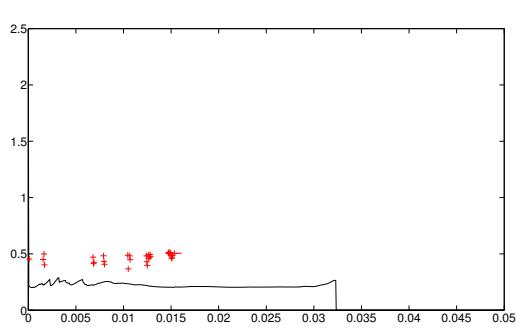
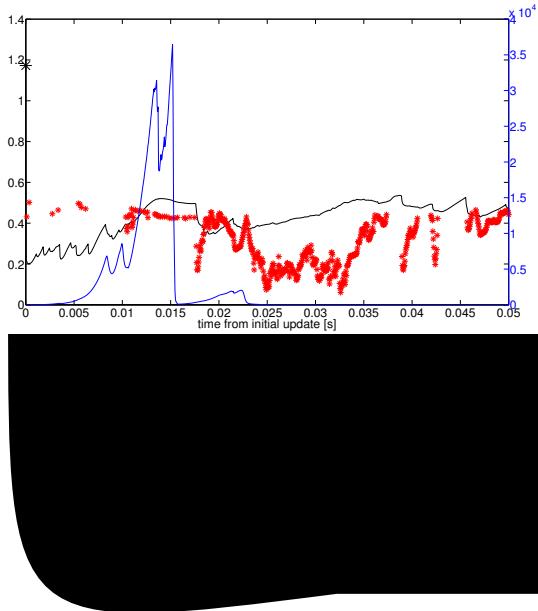
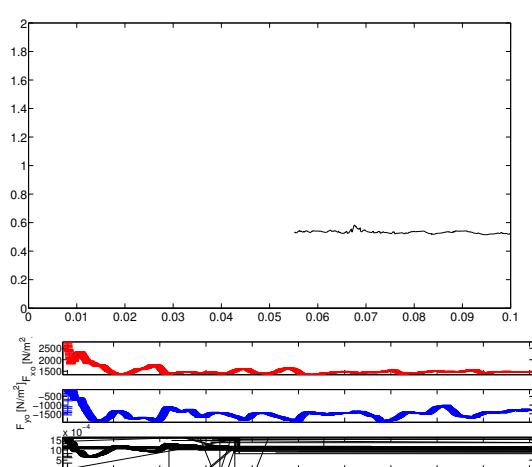
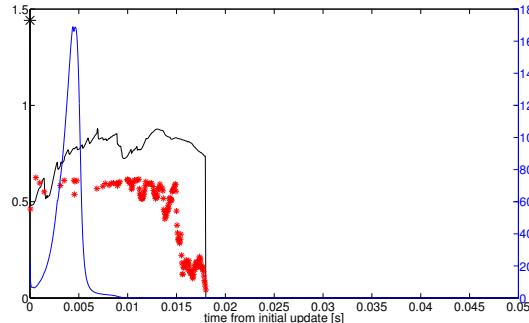
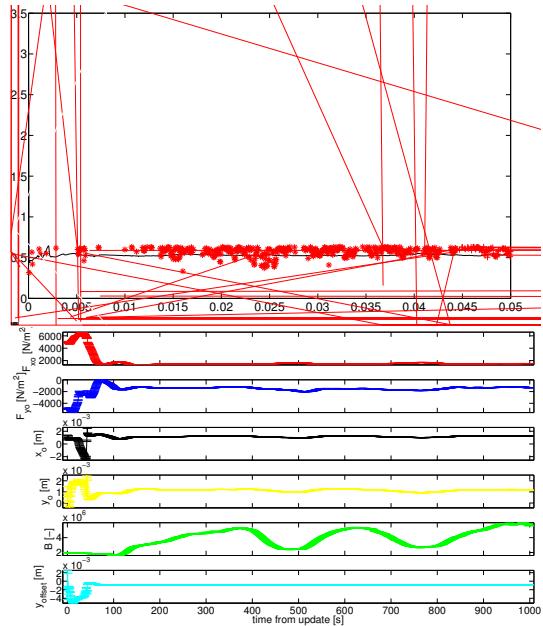
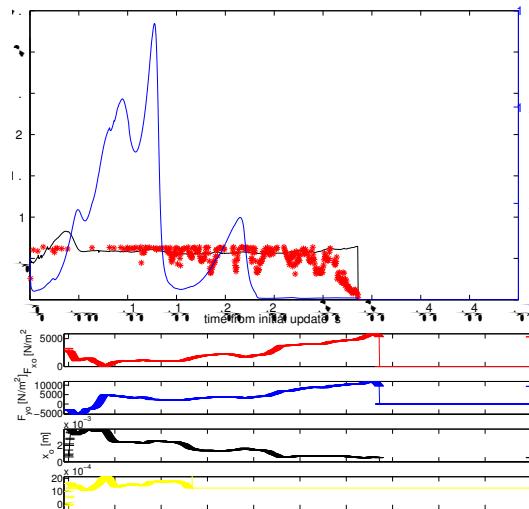
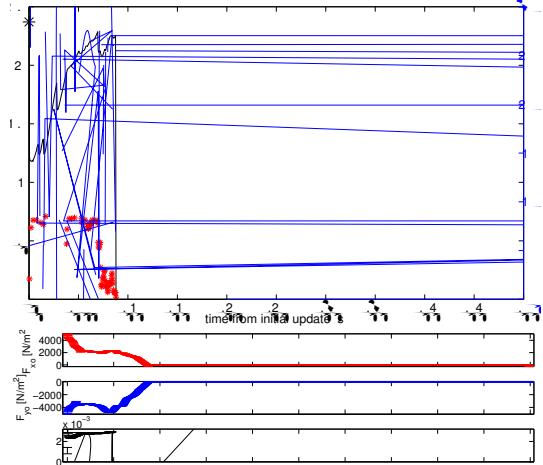


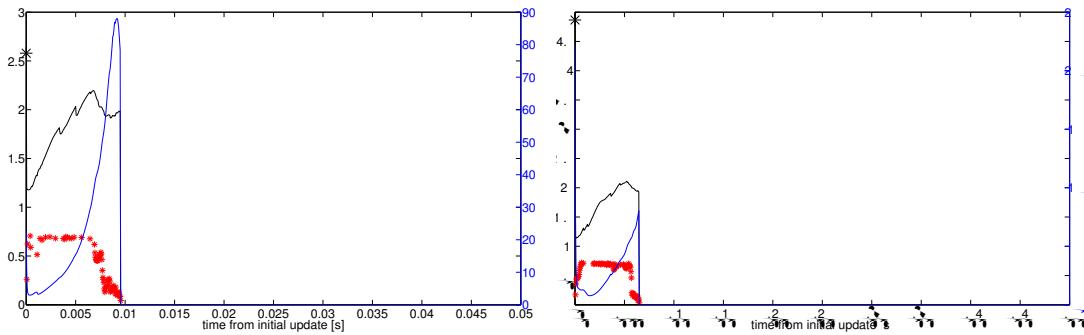
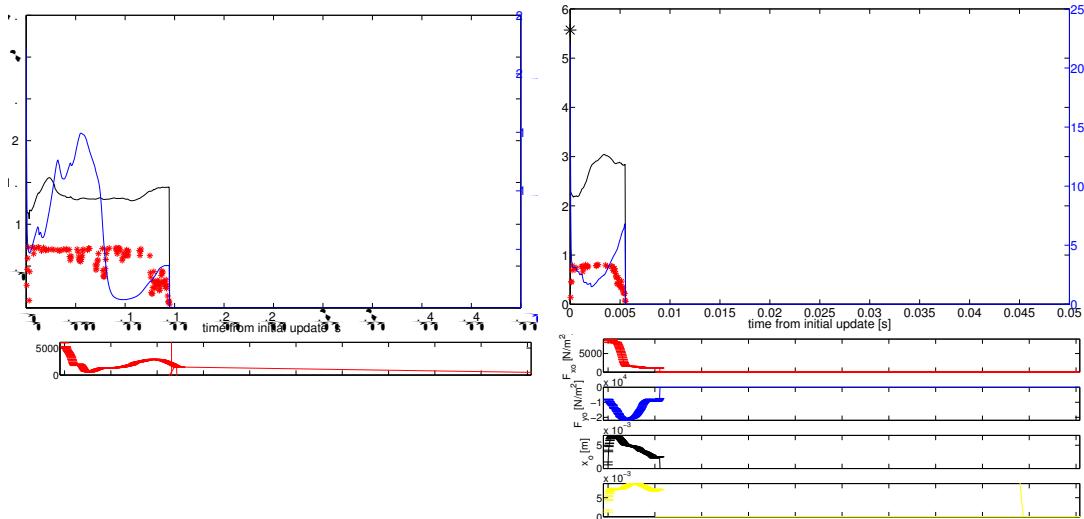
Figure E.8: Force field contours for case 8

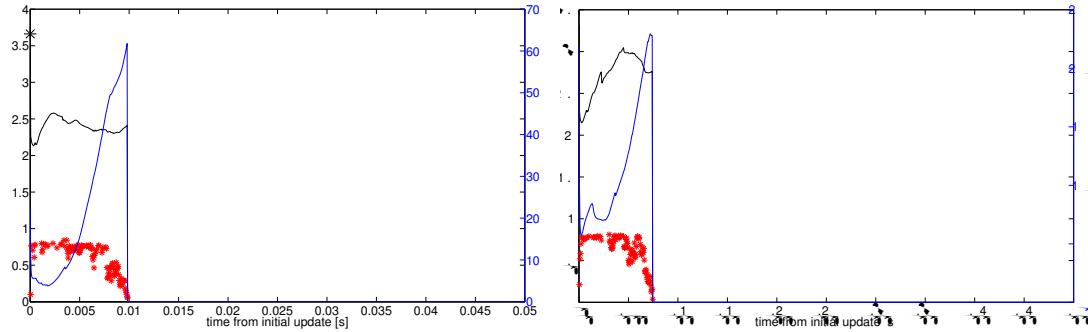
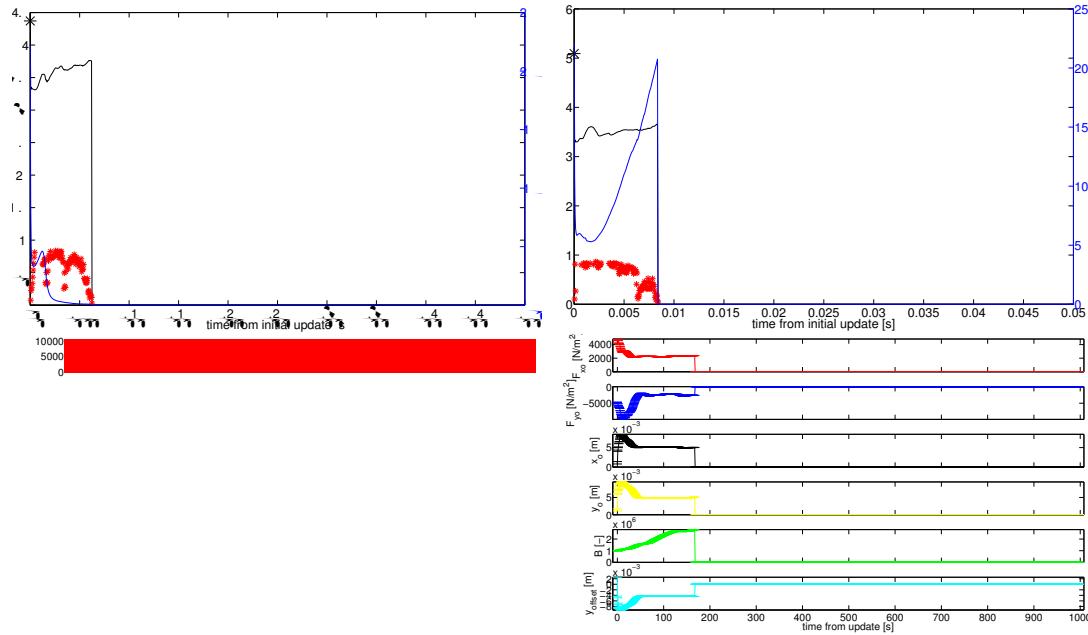
E.3 Parameter estimation development and divergence plots

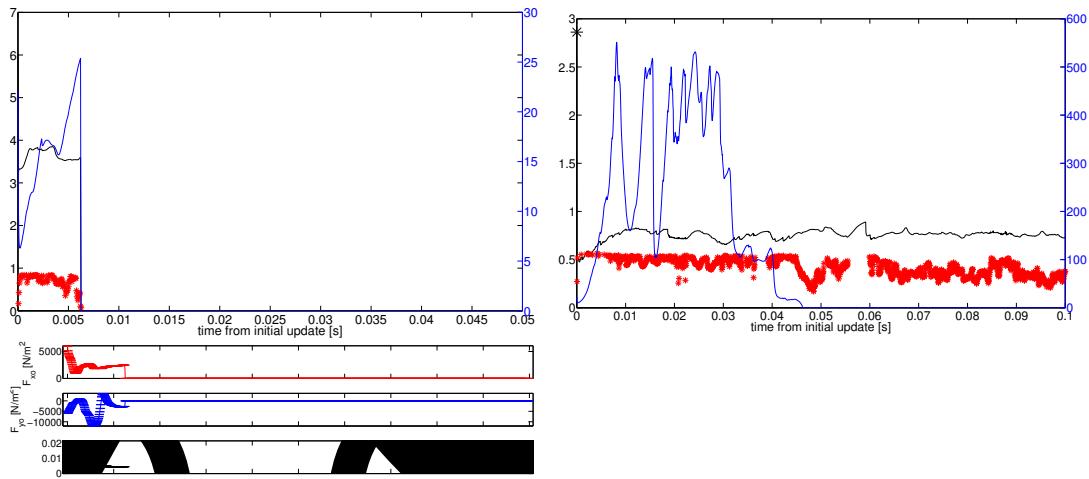
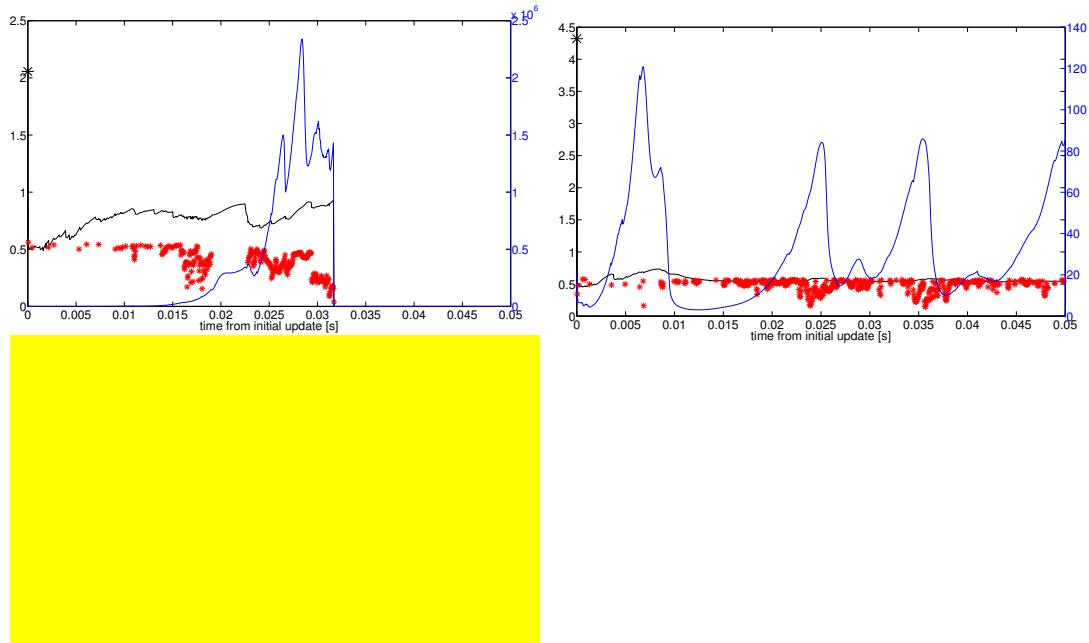
SSSEnKF results; TOP: $RMSE_o * 100[m=s]$ (* black solid line), $\frac{RMSE_f}{100}[N=m^2]$ (x black dashed line), $\frac{\Sigma}{[]} m=s$ (* blue line). BOTTOM: parameter mean and ± 3 error bars. Lines of constant values are equal to true parameter value.

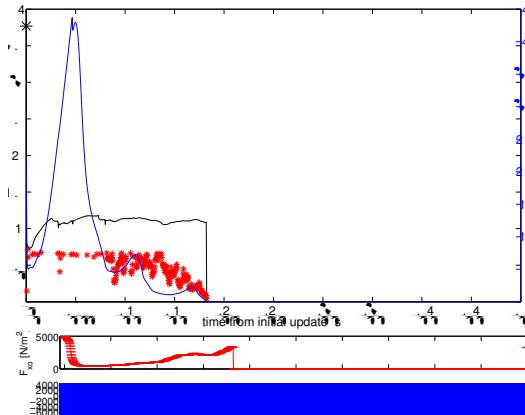
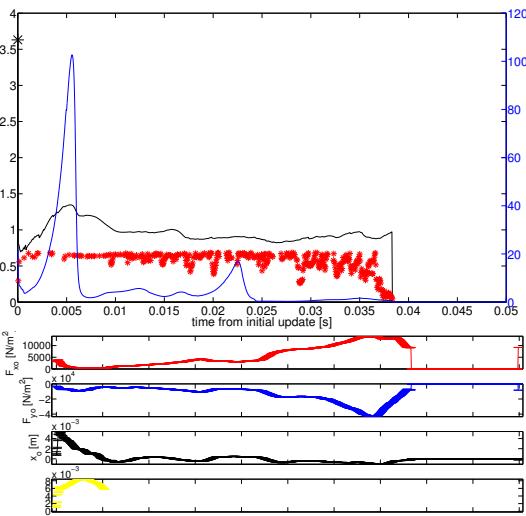
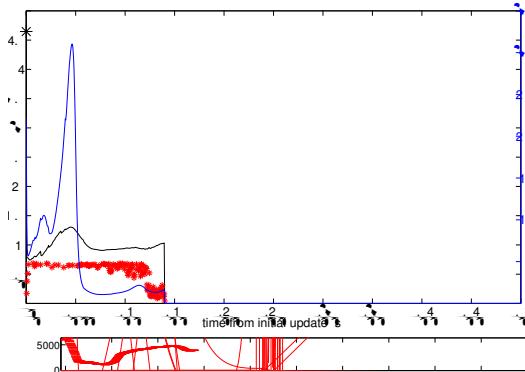
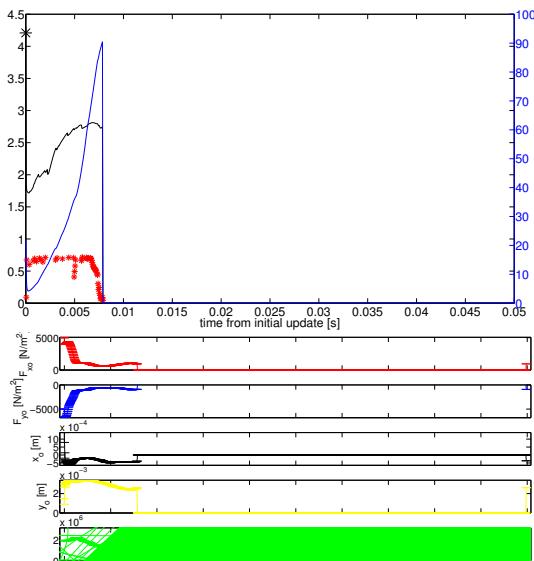
**Figure E.9:** Case 1, prior 2, DEnKF**Figure E.10:** Case 1, prior 3, DEnKF**Figure E.11:** Case 1, prior 4, DEnKF**Figure E.12:** Case 2, prior 1, DEnKF

**Figure E.13:** Case 2, prior 3, DEnKF**Figure E.14:** Case 2, prior 5, DEnKF**Figure E.15:** Case 2, prior 6, DEnKF**Figure E.16:** Case 3, prior 5, DEnKF

**Figure E.17:** Case 3, prior 6, DEnKF**Figure E.18:** Case 3, prior 8, DEnKF**Figure E.19:** Case 3, prior 9, DEnKF**Figure E.20:** Case 4, prior 7, DEnKF

**Figure E.21:** Case 4, prior 8, DEnKF**Figure E.22:** Case 4, prior 9, DEnKF**Figure E.23:** Case 5, prior 7, DEnKF**Figure E.24:** Case 5, prior 8, DEnKF

**Figure E.25:** Case 5, prior 9, DEnKF**Figure E.26:** Case 6, prior 1, DEnKF**Figure E.27:** Case 6, prior 3, DEnKF**Figure E.28:** Case 6, prior 6, DEnKF

**Figure E.29:** Case 7, prior 5, DEnKF**Figure E.30:** Case 7, prior 6, DEnKF**Figure E.31:** Case 7, prior 9, DEnKF**Figure E.32:** Case 8, prior 5, DEnKF

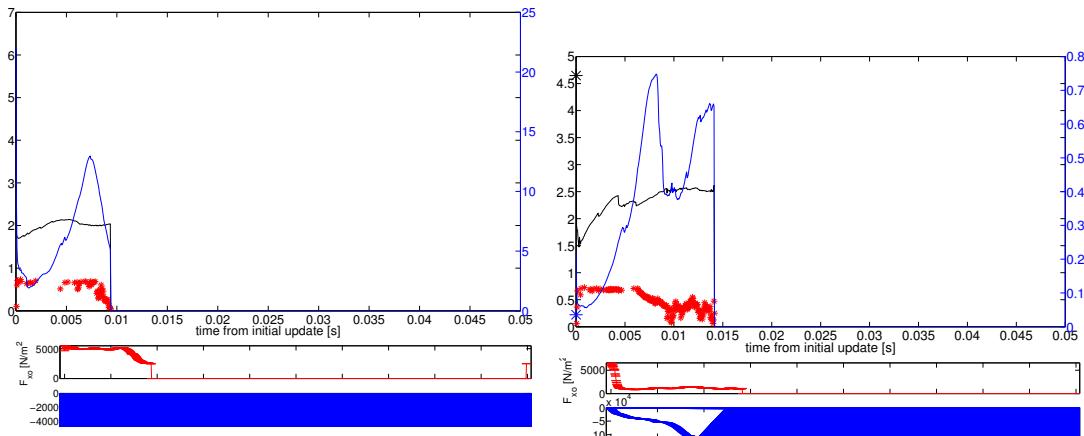


Figure E.33: Case 8, prior 8, DEnKF

Figure E.34: Case 8, prior 9, DEnKF

