

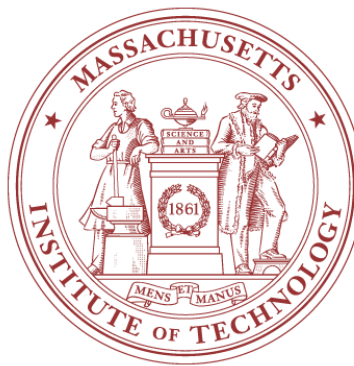
**Multidisciplinary
Simulation, Estimation, and Assimilation Systems
Reports in Ocean Science and Engineering**

MSEAS-10

**Data Assimilation with
Gaussian Mixture Models using the
Dynamically Orthogonal Field Equations**

by

Thomas Sondergaard



**Department of Mechanical Engineering
Massachusetts Institute of Technology**

Cambridge, Massachusetts

September 2011

Data Assimilation with Gaussian Mixture Models using the Dynamically Orthogonal Field Equations

by

Thomas Sondergaard

M.Eng., Imperial College of London (2008)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2011

© Massachusetts Institute of Technology 2011. All rights reserved.

Author
Department of Mechanical Engineering
08/03/2011

Certified by
Pierre F. J. Lermusiaux
Associate Professor of Mechanical Engineering
Thesis Supervisor

Accepted by
David E. Hardt
Chairman, Department Committee on Graduate Theses

Data Assimilation with Gaussian Mixture Models using the Dynamically Orthogonal Field Equations

by

Thomas Sondergaard

Submitted to the Department of Mechanical Engineering
on 08/03/2011, in partial fulfillment of the
requirements for the degree of
Master of Science in Mechanical Engineering

Abstract

Data assimilation, as presented in this thesis, is the statistical merging of sparse observational data with computational models so as to optimally improve the probabilistic description of the field of interest, thereby reducing uncertainties. The centerpiece of this thesis is the introduction of a novel such scheme that overcomes prior shortcomings observed within the community. Adopting techniques prevalent in Machine Learning and Pattern Recognition, and building on the foundations of classical assimilation schemes, we introduce the GMM-DO filter: Data Assimilation with Gaussian mixture models using the Dynamically Orthogonal field equations.

We combine the use of Gaussian mixture models, the EM algorithm and the Bayesian Information Criterion to accurately approximate distributions based on Monte Carlo data in a framework that allows for efficient Bayesian inference. We give detailed descriptions of each of these techniques, supporting their application by recent literature. One novelty of the GMM-DO filter lies in coupling these concepts with an efficient representation of the evolving probabilistic description of the uncertain dynamical field: the Dynamically Orthogonal field equations. By limiting our attention to a dominant evolving stochastic subspace of the total state space, we bridge an important gap previously identified in the literature caused by the dimensionality of the state space.

We successfully apply the GMM-DO filter to two test cases: (1) the Double Well Diffusion Experiment and (2) the Sudden Expansion fluid flow. With the former, we prove the validity of utilizing Gaussian mixture models, the EM algorithm and the Bayesian Information Criterion in a dynamical systems setting. With the application of the GMM-DO filter to the two-dimensional Sudden Expansion fluid flow, we further show its applicability to realistic test cases of non-trivial dimensionality. The GMM-DO filter is shown to consistently capture and retain the far-from-Gaussian statistics that arise, both prior and posterior to the assimilation of data, resulting in its superior performance over contemporary filters.

We present the GMM-DO filter as an efficient, data-driven assimilation scheme, focused on a dominant evolving stochastic subspace of the total state space, that

respects nonlinear dynamics and captures non-Gaussian statistics, obviating the use of heuristic arguments.

Thesis Supervisor: Pierre F. J. Lermusiaux

Title: Associate Professor of Mechanical Engineering

Acknowledgments

I would like to extend my gratitude to my adviser, Pierre Lermusiaux, for having given me the complete freedom to pursue a thesis topic of my own interest. I am further appreciative of the kind understanding he has shown me, particularly in the months leading up to this deadline. His helpful comments, guidance and support have aided in producing a thesis of which I am truly proud.

My most sincere thanks also go to the rest of the MSEAS team, both past and present. Particularly, I wish to thank the members of my office, who have been a family away from home!

I finally wish to acknowledge my family, whose help and support has been invaluable. It is due to them that I have been given the opportunity to pursue a degree at MIT; from them that I have gained my curiosity and desire to learn; and whose values have defined the person that I am.

Contents

1	Introduction	17
1.1	Background	17
1.2	Goals	18
1.3	Thesis Overview	18
2	Data Assimilation	21
2.1	Kalman Filter	22
2.2	Extended Kalman Filter	30
2.3	Ensemble Kalman Filter	34
2.4	Error Subspace Statistical Estimation	37
2.5	Bayes Filter	38
2.6	Particle Filter	40
3	Data Assimilation with Gaussian mixture models using the Dynamically Orthogonal field equations	43
3.1	Gaussian mixture models	44
3.2	The EM algorithm	49
3.2.1	The EM algorithm with Gaussian mixture models	53
3.2.2	Remarks	60
3.3	The Bayesian Information Criterion	61
3.4	The Dynamically Orthogonal field equations	65
3.4.1	Proper Orthogonal Decomposition	66
3.4.2	Polynomial Chaos	66

3.4.3	The Dynamically Orthogonal field equations	67
3.5	The GMM-DO filter	70
3.5.1	Initial Conditions	70
3.5.2	Forecast	72
3.5.3	Observation	73
3.5.4	Update	73
3.5.5	Example	82
3.5.6	Remarks, Modifications and Extensions	88
3.6	Literature Review	91
4	Application 1: Double Well Diffusion Experiment	101
4.1	Introduction	101
4.2	Procedure	104
4.3	Results and Analysis	106
4.4	Conclusion	119
5	Application 2: Sudden Expansion Fluid Flow	121
5.1	Introduction	122
5.2	Procedure	125
5.3	Numerical Method	128
5.4	Results and Analysis	130
5.5	Conclusion	158
6	Conclusion	159
A	Jensen's Inequality and Gibbs' Inequality	161
B	The covariance matrix of a multivariate Gaussian mixture model	165
C	Maximum Entropy Filter	167
C.1	Formulation	167
C.2	Double Well Diffusion Experiment	171

List of Figures

3-1	Gaussian (parametric) distribution, Gaussian mixture model and Gaussian (kernel) density approximation of 20 samples generated from the mixture of uniform distributions: $p_X(x) = \frac{1}{2} \times \mathcal{U}(x; -8, -1) + \frac{1}{2} \times \mathcal{U}(x; 1, 8)$, where $\mathcal{U}(x; a, b) = \frac{1}{b-a}$ denotes the continuous uniform probability density function for random variable X	45
3-2	GMM-DO filter flowchart.	83
3-3	GMM-DO filter update. In column (i), we plot the set of ensemble realizations within the stochastic subspace, $\{\phi\}$; in column (ii), we display the information relevant to the state space. Panel (a) shows the prior state estimate; in panel (b), we show the fitting of Gaussian mixture models of complexity $M = 1$ (PD) and $M = 2$ (GMM), and plot their marginal distributions for each of the stochastic coefficients; in panel (c), we provide the posterior state estimate again in the decomposed form that accords with the D.O. equations.	87
3-4	Schematic representation of advantages of kernel over single Gaussian filter for a low-order model. The background is a projection of a trajectory from the Lorenz-63 model showing the attractor structure. Superimposed is an idealized image of the single Gaussian (outer curve) and kernel (inner curves) prior distributions for a three-member ensemble. (Anderson and Anderson, 1999)	94

4-1	Forcing Function, $f(x)$. At any location (o), x , the ball is forced under pseudo-gravity in the direction indicated by the appropriate vector. The magnitude of the vector corresponds to the strength of the forcing. We note that there exists an unstable node at the origin, and two stable nodes at $x = \pm 1$, corresponding to the minima of the wells.	102
4-2	Climatological distribution and Gaussian mixture approximation for $\kappa = 0.40$. In accordance with intuition, the distributions are bimodal, appropriately centered on the minima of each of the two wells.	103
4-3	Example trajectory of the ball for $\kappa = 0.45$. The horizontal axis denotes time; the vertical axis the location of the ball. Superimposed onto the plot are intermittent measurements, shown in green, with their associated uncertainties.	104
4-4	The true trajectory for the ball is obtained by appropriately stitching together two runs, each initiated at $x = 0$	105
4-5	Legend for the Double Well Diffusion Experiment.	106
4-6	Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.4$; $\sigma_o^2 = 0.025$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.	107
4-7	Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.4$; $\sigma_o^2 = 0.050$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.	108
4-8	Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.4$; $\sigma_o^2 = 0.100$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.	109
4-9	Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.5$; $\sigma_o^2 = 0.025$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.	110
4-10	Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.5$; $\sigma_o^2 = 0.050$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.	111

4-11	Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.5$; $\sigma_o^2 = 0.100$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.	112
4-12	Analysis of the prior and posterior distributions (and particle representations) by each of the three filters (EnKF, GMM-DO and MEF) for the particular case of $N = 1,000$ and $\kappa = 0.5$, centered on the observation immediately prior to the true transition of the ball. . . .	115
4-13	Analysis of the prior and posterior distributions (and particle representations) by each of the three filters (EnKF, GMM-DO and MEF) for the particular case of $N = 1,000$ and $\kappa = 0.5$, centered on the observation immediately following the true transition of the ball. . . .	116
4-14	Analysis of the prior and posterior distributions (and particle representations) by each of the three filters (EnKF, GMM-DO and MEF) for the particular case of $N = 1,000$ and $\kappa = 0.5$, centered on the second observation following the true transition of the ball.	117
5-1	Setup of the sudden expansion test case (Fearn et al., 1990).	122
5-2	Boundaries of symmetric and asymmetric flow as a function of aspect ratio, expansion ratio and Reynolds number (Cherdron et al., 1978). . .	123
5-3	Flow patterns at different Reynolds numbers for an aspect ratio of 8 and an expansion ratio of 2. (a) $Re = 110$. (b) $Re = 150$. (c) $Re = 500$. (Cherdron et al., 1978).	124
5-4	Numerical and experimental velocity plots at $Re = 140$. The numerically calculated profiles are shown as continuous curves. (a) $x/H = 1.25$; (b) $x/H = 2.5$; (c) $x/H = 5$; (d) $x/H = 10$; (e) $x/H = 20$; (f) $x/H = 40$. (Fearn et al., 1990).	124
5-5	Calculated streamlines at $Re = 140$. (Fearn et al., 1990).	124
5-6	Sudden Expansion Test Setup.	125
5-7	Initial mean field of the DO decomposition.	126
5-8	Selection of initial modes of the DO decomposition.	128

5-9	Observation Locations: $(x_{obs}, y_{obs}) = \{(4, -\frac{1}{4}), (4, 0), (4, \frac{1}{4})\}$	129
5-10	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 0$	132
5-11	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 10$	133
5-12	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 20$	134
5-13	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 30$	135
5-14	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 40$	136
5-15	True solution; DO mean field; and first four DO modes at the first assimilation step, time $T = 50$	137
5-16	True solution; DO mean field; and joint and marginal prior distribu- tions, identified by the Gaussian mixture model of complexity 29, and associated ensembles of the first four modes at the first assimilation step, time $T = 50$	138
5-17	True solution; observation and its associated Gaussian distribution; and the prior and posterior distributions at the observation locations at time $T = 50$	139
5-18	True solution; condensed representation of the posterior DO decompo- sition; and root mean square errors at time $T = 50$	140
5-19	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 60$	141
5-20	True solution; DO mean field; and first four DO modes at the second assimilation step, time $T = 70$	142
5-21	True solution; DO mean field; and joint and marginal prior distribu- tions, identified by the Gaussian mixture model of complexity 20, and associated ensembles of the first four modes at the second assimilation step, time $T = 70$	143

5-22	True solution; observation and its associated Gaussian distribution; and the prior and posterior distributions at the observation locations at time $T = 70$	144
5-23	True solution; condensed representation of posterior DO decomposition; and root mean square errors at time $T = 70$	145
5-24	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 80$	146
5-25	True solution; DO mean field; and first four DO modes at the third assimilation step, time $T = 90$	147
5-26	True solution; DO mean field; and joint and marginal prior distributions, identified by the Gaussian mixture model of complexity 14, and associated ensembles of the first four modes at the third assimilation step, time $T = 90$	148
5-27	True solution; observation and its associated Gaussian distribution; and the prior and posterior distributions at the observation locations at time $T = 90$	149
5-28	True solution; condensed representation of posterior DO decomposition; and root mean square errors at time $T = 90$	150
5-29	True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 100$	151
5-30	Time-history of the root means square errors for the GMM-DO filter and DO-ESSE Scheme A.	152
5-31	Bimodal distribution for the most dominant stochastic coefficient, Φ_1 , at $T = 40$. The GMM-DO filter captures and retains this bimodality throughout the simulation of the sudden expansion fluid flow, resulting in its superior performance.	153

5-32	Gaussian mixture model approximation of the ensemble set, $\{\phi\}$, at the time of the first assimilation of observations, $T = 50$. Assuming MATLAB's 'ksdensity' function to represent an appropriate approximation to the true marginal densities, we note the satisfactory approximation of the Gaussian mixture model.	154
5-33	An example of the manner in which the GMM-DO filter captures the true solution through its use of Gaussian mixture models. We equally note the increased weights placed on the mixtures surrounding the true solution following the Bayesian update, depicted by the green curve. .	155
5-34	A second example of the manner in which the GMM-DO filter captures the true solution through its use of Gaussian mixture models. Here, however, the true solution is contained within a mode of small – but finite – probability. Note the increased weights placed on the mixtures surrounding the true solution following the Bayesian update.	156
5-35	We show the part of the true solution orthogonal to the stochastic subspace for the case of 15 and 20 modes at the time of the first assimilation. 'Difference' refers to the difference between the true solution and the mean field; 'error' to the part of the true solution not captured by the GMM-DO filter. We note that as we increase the number of modes, the norm of the error marginally decreases, indicative of convergence.	157

Chapter 1

Introduction

1.1 Background

The need for generating accurate forecasts, whether it be for the atmosphere, weather or ocean, requires little justification and has a long and interesting history (see e.g. Kalnay (2003)). Such forecasts are, needless to say, provided by highly developed and complex computational fluid dynamics codes. An example of such is that due to MSEAS (Haley and Lermusiaux (2010); MSEAS manual) .

Due to the chaotic nature of the weather and ocean, however, classically exemplified by the simple Lorenz-63 model (Lorenz, 1963), any present state estimate – however accurate – is certain to deteriorate with time. This necessitates the assimilation of external observations. Unfortunately, due to the limitation of resources, these are sparse in both space and time. Given further the dimensionality of the state vector associated with the weather and ocean, a crucial research thrust is thus the efficient distribution of the observational information amongst the entire state space.

Data assimilation concerns the statistical melding of computational models with sparse observations for the purposes of improving the current state representation. By arguing that the most complete description of any field of interest is its probability distribution, the ultimate goal of any data assimilation scheme is the ‘Bayes filter’, to be introduced in this thesis. The scheme to be developed in this thesis is no different.

1.2 Goals

The centerpiece of this thesis is the introduction of a novel data assimilation scheme that overcomes prior shortcomings observed within the data assimilation community. Adopting techniques prevalent in Machine Learning and Pattern Recognition, and building on the foundations of the Kalman filter (Kalman) and ESSE (Lermusiaux, 1997), we introduce the GMM-DO filter: Data Assimilation with Gaussian mixture models using the Dynamically Orthogonal field equations. By application of the Dynamically Orthogonal field equations (Sapsis (2010), Sapsis and Lermusiaux (2009)), we focus our attention on a dominant evolving stochastic subspace of the total state space, thereby bridging an important gap previously identified in the literature caused by the dimensionality of the state space. Particularly, with this, we make obsolete ad hoc localization procedures previously adopted – with limited success - by other filters introduced in this thesis. With the GMM-DO filter, we further stray from the redundant operating on ensemble members during the update step; rather, under the assumption that the fitted Gaussian mixture model accurately captures the true prior probability density function, we analytically carry out Bayes’ Law efficiently within the stochastic subspace.

We describe the GMM-DO filter as an efficient, data-driven assimilation scheme that preserves non-Gaussian statistics and respects nonlinear dynamics, obviating the use of heuristic arguments.

1.3 Thesis Overview

In chapter 2, we explore various existing data assimilation schemes, outlining both their strengths and weaknesses. We will mainly limit our attention to methodologies based on the original Kalman filter (Kalman), whose general theory will serve as the foundation for the GMM-DO filter.

In chapter 3, we will introduce the critical components that ultimately combine to produce the GMM-DO filter. After providing the details of the scheme itself, we will

give a simple example of its update step. We conclude the chapter with a literature review, in which we compare and contrast the GMM-DO filter against past and more recent schemes built on similar foundations.

In chapters 4 and 5, we apply the GMM-DO filter to test cases that admit far-from-Gaussian statistics. We specifically evaluate the performance of the GMM-DO filter when applied to the Double Well Diffusion Experiment (Chapter 4) and the Sudden Expansion fluid flow (Chapter 5), comparing its performance against that of contemporary data assimilation schemes. We describe in detail the manner in which the GMM-DO filter efficiently captures the dominant non-Gaussian statistics, ultimately outperforming current state-of-the-art filters.

We give our concluding remarks in chapter 6.

Chapter 2

Data Assimilation

Data assimilation, as presented in this thesis, is the statistical merging of observational data with computational models for the sake of reducing uncertainties and improving the probabilistic description of any field of interest. The concern will particularly be with the aspect of filtering: generating the most complete description at present time, employing all past observations. In a future work, we may further extend this to the case of smoothing.

Today, ocean and weather forecasts are provided by computational models. Inevitably, these models fail to capture the true nature of the field of interest, be it due to discretization errors, approximations of nonlinearities, poor knowledge of parameter values, model simplifications, etc. As a consequence, one often resorts to providing a probabilistic description of the field of interest, introducing stochastic forcing, parameter values and boundary conditions where necessary (Lermusiaux et al., 2006). By further incorporating uncertainties in the initial conditions, data assimilation goes beyond providing a single deterministic estimate of the field of interest; rather, given the inherent uncertainties in both forecast and observations, data assimilation provides a statistical description from which one may quantify states and errors of interest.

The purpose of this chapter is to explore various traditional data assimilation schemes, outlining both their strengths and weaknesses. We will mainly limit our attention to methodologies based on the original Kalman filter, whose general theory will serve as the foundation for the data assimilation scheme to be developed in this

thesis.

The following presentation is largely based on the MSEAS thesis by Eric Heubel (2008), the MIT class notes on Identification, Estimation and Learning by Harry Asada (2011), chapter 22 of Introduction to Geophysical Fluid Dynamics by Cushman-Roisin and Beckers (2007), and the seminal books by Gelb (1974) and Jazwinski (1970).

2.1 Kalman Filter

The Kalman filter (Kalman) merges model predictions with observations based on sound statistical arguments, utilizing quantified uncertainties of both predicted state variables and measurements. It is sequential in nature, essentially consisting of two distinct components performed recursively: a forecast step and an update step. While the structure of the update step will change little as we progress into more evolved filters, the forecast step of the Kalman filter specifically assumes linear dynamics. Particularly, we write for the discrete-time governing equation:

$$\mathbf{X}_{k+1} = \mathbf{A}_k \mathbf{X}_k + \mathbf{G}_k \mathbf{\Gamma}_k, \quad (2.1)$$

where $\mathbf{X} \in \mathbb{R}^n$ is the (random) state vector; $\mathbf{\Gamma}_k \in \mathbb{R}^l$ is a random vector (source of noise); k is a discrete time index; and $\mathbf{A}_k \in \mathbb{R}^{n \times n}$ and $\mathbf{G}_k \in \mathbb{R}^{n \times l}$ are matrices whose physical interpretations require little clarification.

Sparse and noisy measurements of the system, $\mathbf{Y}_k \in \mathbb{R}^p$, are intermittently collected, assumed to be a linear function of the state vector:

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{\Upsilon}_k, \quad (2.2)$$

where the vector $\mathbf{\Upsilon}_k \in \mathbb{R}^p$ represents measurement noise. The observation operator $\mathbf{H}_k \in \mathbb{R}^{p \times n}$ linearly maps elements from the state space to the observation space, thus allowing their statistical comparison.

The Kalman filter makes a number of assumptions on the statistics of the system.

Particularly, the sources of noise are assumed to be unbiased:

$$\begin{aligned}\mathcal{E} [\mathbf{\Gamma}_t] &= \mathbf{0} \\ \mathcal{E} [\mathbf{\Upsilon}_t] &= \mathbf{0},\end{aligned}\tag{2.3}$$

with the following auto- and cross-correlations in space and time:

$$\begin{aligned}\mathcal{E} [\mathbf{\Gamma}_s \mathbf{\Gamma}_t^T] &= \delta_{st} \mathbf{Q}_t \\ \mathcal{E} [\mathbf{\Upsilon}_s \mathbf{\Upsilon}_t^T] &= \delta_{st} \mathbf{R}_t \\ \mathcal{E} [\mathbf{\Gamma}_s \mathbf{\Upsilon}_t^T] &= \mathbf{0},\end{aligned}\tag{2.4}$$

where δ_{ij} denotes the Kronecker delta. With this, we proceed to examine the machinery of the Kalman filter.

Update

At any point in time, the goal of the Kalman filter is to determine the state vector, \mathbf{X}^a , that minimizes the quadratic cost function

$$\mathbf{X}^a = \underset{\mathbf{X}}{\operatorname{argmin}} \mathcal{E} [(\mathbf{X} - \mathbf{x}^t)^T (\mathbf{X} - \mathbf{x}^t) \mid \mathbf{y}^k, \mathbf{X}_0],\tag{2.5}$$

where \mathbf{x}^t is the true state of the system; $\mathbf{y}^k = \{\mathbf{y}_1, \dots, \mathbf{y}_k\}$ represents all measurements collected up to the current time step; and \mathbf{X}_0 is the initial state estimate. We refer to \mathbf{X}^a as the *analysis* vector.

If we define the error $\mathbf{\Delta} = \mathbf{X} - \mathbf{x}^t$, with the assumption that this estimate is unbiased (i.e. $\mathcal{E} [\mathbf{\Delta}] = \mathbf{0}$), we find that (2.5) is equivalent to writing

$$\mathbf{x}^a = \underset{\mathbf{x}}{\operatorname{argmin}} \mathcal{E} [\mathbf{\Delta}^T \mathbf{\Delta} \mid \mathbf{y}^k, \mathbf{X}_0]\tag{2.6}$$

$$= \underset{\mathbf{x}}{\operatorname{argmin}} \operatorname{Tr} (\mathcal{E} [\mathbf{\Delta} \mathbf{\Delta}^T \mid \mathbf{y}^k, \mathbf{X}_0])\tag{2.7}$$

$$\equiv \underset{\mathbf{x}}{\operatorname{argmin}} \operatorname{Tr} (\mathbf{P}),\tag{2.8}$$

where \mathbf{P} is the state covariance matrix conditioned on all available measurements

and $\text{Tr}(\cdot)$ denotes the trace operator. Thus, seen from this perspective, the Kalman filter attempts to find the state that reduces the sum of the variances of the system – a highly reasonable goal.

The idea behind the filter's update step is as follows: at the time of a new measurement, the current state estimate (from hereon *forecast*), denoted \mathbf{X}^f , is *linearly* updated using the observed data, \mathbf{Y} , weighted appropriately by inherent knowledge of the statistical uncertainties. We write:

$$\mathbf{X}^a = \mathbf{X}^f + \mathbf{K} (\mathbf{Y} - \mathbf{H}\mathbf{X}^f), \quad (2.9)$$

for which we wish to evaluate the optimal gain matrix $\mathbf{K} \in \mathbb{R}^{n \times p}$. As before, we define the errors:

$$\begin{aligned} \Delta^f &= \mathbf{X}^f - \mathbf{x}^t \\ \Delta^a &= \mathbf{X}^a - \mathbf{x}^t \\ \Delta^o &= \mathbf{Y} - \mathbf{y}^t \end{aligned} \quad (2.10)$$

with the assumption that these estimates are unbiased, i.e.

$$\mathcal{E} [\Delta^f] = \mathcal{E} [\Delta^a] = \mathcal{E} [\Delta^o] = 0. \quad (2.11)$$

For completeness of notation, we further define the error-covariance matrices:

$$\begin{aligned} \mathbf{R} &= \mathcal{E} [\Delta^o \Delta^{oT}] \\ \mathbf{P}^f &= \mathcal{E} [\Delta^f \Delta^{fT}] \\ \mathbf{P}^a &= \mathcal{E} [\Delta^a \Delta^{aT}]. \end{aligned} \quad (2.12)$$

Using equation (2.10), the analysis step, (2.9), can thus be written

$$\mathbf{x}^t + \Delta^a = \mathbf{x}^t + \Delta^f + \mathbf{K} (\Delta^o - \mathbf{H}\Delta^f) - \mathbf{K} (\mathbf{y}^t - \mathbf{H}\mathbf{x}^t) \quad (2.13)$$

giving

$$\Delta^a = \Delta^f + K (\Delta^o - H \Delta^f). \quad (2.14)$$

With this, we derive an expression for the cost function, denoting this J , in (2.5):

$$J = \mathcal{E} [\Delta^{aT} \Delta^a] \quad (2.15)$$

$$= \mathcal{E} [(\Delta^f + K (\Delta^o - H \Delta^f))^T (\Delta^f + K (\Delta^o - H \Delta^f))] \quad (2.16)$$

$$= \mathcal{E} [(I - KH) \Delta^f + K \Delta^o]^T [(I - KH) \Delta^f + K \Delta^o] \quad (2.17)$$

$$\begin{aligned} &= \mathcal{E} [\Delta^{fT} \Delta^f] + \mathcal{E} [\Delta^{fT} H^T K^T K H \Delta^f] - 2 \mathcal{E} [\Delta^{fT} K H \Delta^f] \\ &\quad + 2 \mathcal{E} [\Delta^{fT} K \Delta^o] - 2 \mathcal{E} [\Delta^{oT} K^T K H \Delta^f] + \mathcal{E} [\Delta^{oT} K^T K \Delta^o] \end{aligned} \quad (2.18)$$

Imposing zero cross-correlations between the state and observation errors,

$$\mathcal{E} [\Delta^o \Delta^{fT}] = \mathbf{0}, \quad (2.19)$$

combined with standard matrix calculus (see e.g. Petersen and Pedersen (2008)), we derive an expression for the gradient of the cost function with respect to the gain matrix:

$$\begin{aligned} \frac{\partial J}{\partial K} &= 2KH \mathcal{E} [\Delta^f \Delta^{fT}] H^T - 2KH \mathcal{E} [\Delta^f \Delta^{oT}] - 2K \mathcal{E} [\Delta^o \Delta^{fT}] H^T \\ &\quad + 2K \mathcal{E} [\Delta^o \Delta^{oT}] + 2 \mathcal{E} [\Delta^f \Delta^{oT}] - 2 \mathcal{E} [\Delta^f \Delta^{fT}] H^T \end{aligned} \quad (2.20)$$

$$= 2KH P^f H^T + 2KR - 2P^f H^T. \quad (2.21)$$

Equating the above with zero leads to the equation for the optimal gain matrix, termed the Kalman gain matrix:

$$K = P^f H^T (H P^f H^T + R)^{-1}. \quad (2.22)$$

With this, we get for the analysis state covariance matrix:

$$\mathbf{P}^a = \mathcal{E} \left[\left(\Delta^f + \mathbf{K} (\Delta^o - \mathbf{H} \Delta^f) \right) \left(\Delta^f + \mathbf{K} (\Delta^o - \mathbf{H} \Delta^f) \right)^T \right] \quad (2.23)$$

$$= \mathcal{E} \left[\left((\mathbf{I} - \mathbf{K} \mathbf{H}) \Delta^f + \mathbf{K} \Delta^o \right) \left((\mathbf{I} - \mathbf{K} \mathbf{H}) \Delta^f + \mathbf{K} \Delta^o \right)^T \right] \quad (2.24)$$

$$= (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}^f (\mathbf{I} - \mathbf{K} \mathbf{H})^T + \mathbf{K} \mathbf{R} \mathbf{K} \quad (2.25)$$

$$= (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}^f. \quad (2.26)$$

We note that neither the Kalman gain matrix nor the analysis state covariance matrix depends on the actual measurements and can thus be calculated off-line. Instead, only the updated state vector accounts for the observations, taking the form:

$$\bar{\mathbf{x}}^a = \mathcal{E} \left[\mathbf{X}^f + \mathbf{K} (\mathbf{Y} - \mathbf{H} \mathbf{X}^f) \right] \quad (2.27)$$

$$= \bar{\mathbf{x}}^f + \mathbf{K} (\mathbf{y} - \mathbf{H} \bar{\mathbf{x}}^f), \quad (2.28)$$

where we have used the over-bar notation, $\bar{\mathbf{x}}$, to denote the mean estimate, thus differentiating this from its associated random vector, \mathbf{X} .

It is instructive at this point to examine the structure of the Kalman update equations, particularly the importance played by the state covariance matrix \mathbf{P}^f . We do so with a simple example.

Example

Let us assume that $\mathbf{X} \in \mathbb{R}^2$, with the following forecast:

$$\bar{\mathbf{x}}^f = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{and} \quad \mathbf{P}^f = \begin{bmatrix} 16 & -4 \\ -4 & 4 \end{bmatrix}.$$

Let us further assume that we observe only the first state variable, with an observation uncertainty of $\sigma_{obs}^2 = 8$. Therefore,

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{R} = 8.$$

With this, the analysis state vector becomes:

$$\begin{aligned}
\bar{\mathbf{x}}^a &= \bar{\mathbf{x}}^f + \mathbf{P}^f \mathbf{H}^T (\mathbf{H} \mathbf{P}^f \mathbf{H}^T + \mathbf{R})^{-1} (\mathbf{y} - \mathbf{H} \bar{\mathbf{x}}^f) \\
&= \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 16 & -4 \\ -4 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \left(\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 16 & -4 \\ -4 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 8 \right)^{-1} \left(\hat{y} - \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right) \\
&= \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 16 \\ -4 \end{bmatrix} \frac{y - 1}{16 + 8},
\end{aligned}$$

with the associated analysis covariance matrix:

$$\begin{aligned}
\mathbf{P}^a &= (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}^f \\
&= \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{16 + 8} \begin{bmatrix} 16 \\ -4 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \right) \begin{bmatrix} 16 & -4 \\ -4 & 4 \end{bmatrix} \\
&= \frac{1}{3} \begin{bmatrix} 16 & -4 \\ -4 & 10 \end{bmatrix}.
\end{aligned}$$

This simple example serves to illustrate three important roles played by the state covariance matrix, \mathbf{P}^f :

1. In $(\mathbf{H} \mathbf{P}^f \mathbf{H}^T + \mathbf{R})^{-1}$, the matrix \mathbf{P}^f determines the amount of weight that should be allocated to the observations in accordance with the uncertainty in the prior estimate. For this reason, it is crucial that one obtains a good approximation to the true uncertainty in the vicinity of the observations, generally represented by terms on or close to the diagonal of \mathbf{P}^f .
2. The term $\mathbf{P}^f \mathbf{H}^T$ serves to distribute information due to the sparse observations among the entire state space. Thus, while it is crucial to estimate the local variances correctly, it is equally important to correctly estimate the off-diagonal terms of the state covariance matrix. With this, \mathbf{P}^f allows the propagation of information from observation locations to remote, unobserved parts of the system as well as across state variables, taking into account the relative error

of observations and models (Cushman-Roisin and Beckers, 2007).

3. Since the posterior covariance matrix is a function of its prior, any errors initially present will remain following the update and potentially compound when evolving the state estimate forward in time.

From the above, it is evident that the chosen state covariance matrix, \mathbf{P}^f , takes great importance. For a heuristically chosen \mathbf{P}^f , the above analysis refers simply to *Optimal Interpolation*. The Kalman filter uses the given dynamics, however, to update the covariance matrix between time steps. We show this in the following section.

Forecast

Based on the current estimates $\bar{\mathbf{x}}_k^a$ and \mathbf{P}_k^a , we wish to obtain the forecast at time $k + 1$. By taking the expectation of (2.1), we have:

$$\bar{\mathbf{x}}_{k+1}^f = \mathcal{E} [\mathbf{A}_k \mathbf{X}_k^a + \mathbf{G}_k \boldsymbol{\Gamma}_k] \quad (2.29)$$

$$= \mathbf{A}_k \bar{\mathbf{x}}_k^a. \quad (2.30)$$

Furthermore, by using (2.1), (2.2) and (2.4), we can show that $\mathcal{E} [\boldsymbol{\Gamma}_k \boldsymbol{\Delta}_k^{aT}] = \mathbf{0}$. With this, we therefore obtain:

$$\mathbf{P}_{k+1}^f = \mathcal{E} [\boldsymbol{\Delta}_{k+1}^f (\boldsymbol{\Delta}_{k+1}^f)^T] \quad (2.31)$$

$$= \mathcal{E} [(\mathbf{A} \boldsymbol{\Delta}_k^a - \mathbf{G}_k \boldsymbol{\Gamma}_k) (\mathbf{A} \boldsymbol{\Delta}_k^a - \mathbf{G}_k \boldsymbol{\Gamma}_k)^T] \quad (2.32)$$

$$\begin{aligned} &= \mathbf{A}_k \mathcal{E} [\boldsymbol{\Delta}_k^a \boldsymbol{\Delta}_k^{aT}] \mathbf{A}_k^T - \mathbf{G}_k \mathcal{E} [\boldsymbol{\Gamma}_k \boldsymbol{\Delta}_k^{aT}] \mathbf{A}_k^T - \mathbf{A}_k \mathcal{E} [\boldsymbol{\Delta}_k^a \boldsymbol{\Gamma}_k^T] \mathbf{G}_k^T \\ &\quad + \mathbf{G}_k \mathcal{E} [\boldsymbol{\Gamma}_k \boldsymbol{\Gamma}_k^T] \mathbf{G}_k^T \end{aligned} \quad (2.33)$$

$$= \mathbf{A}_k \mathbf{P}_k^a \mathbf{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T. \quad (2.34)$$

This completes the forecast step. In summary, the Kalman filter proceeds as follows:

Definition: **Kalman Filter**

For the discrete time governing equation

$$\mathbf{X}_{k+1} = \mathbf{A}_k \mathbf{X}_k + \mathbf{G}_k \mathbf{\Gamma}_k, \quad (2.35)$$

with observation model

$$\mathbf{Y}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{\Upsilon}_k, \quad (2.36)$$

perform the following two steps recursively:

1. *Forecast:* Given current estimates $\bar{\mathbf{x}}_k^a$ and \mathbf{P}_k^a , obtain estimates at time $k + 1$ using:

$$\bar{\mathbf{x}}_{k+1}^f = \mathbf{A}_k \bar{\mathbf{x}}_k^a \quad (2.37)$$

$$\mathbf{P}_{k+1}^f = \mathbf{A}_k \mathbf{P}_k^a \mathbf{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T. \quad (2.38)$$

2. *Update:* Given estimates $\bar{\mathbf{x}}_k^f$ and \mathbf{P}_k^f , and observation \mathbf{y}_k , with measurement error covariance matrix, \mathbf{R} , update the estimates according to:

$$\bar{\mathbf{x}}_k^a = \bar{\mathbf{x}}_k^f + \mathbf{K} \left(\mathbf{y}_k - \mathbf{H} \bar{\mathbf{x}}_k^f \right) \quad (2.39)$$

$$\mathbf{P}_k^a = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}_k^f, \quad (2.40)$$

where

$$\mathbf{K} = \mathbf{P}_k^f \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_k^f \mathbf{H}^T + \mathbf{R} \right)^{-1}. \quad (2.41)$$

Two remarks may suitably be made on the optimality of the Kalman filter (Asada, 2011):

(1) If we assume that the optimal filter is linear, then the Kalman filter is the state estimator having the smallest posterior error covariance among all *linear* filters.

(2) If we assume that the external noise processes are Gaussian, then the Kalman filter is the optimal minimum variance estimator among all linear *and* nonlinear filters.

For these reasons, the past popularity of the Kalman filter comes as no surprise.

It is firmly grounded in the theory of linearity, however, which greatly restricts its applicability. In the following, we therefore identify filters that have attempted to overcome this limitation.

2.2 Extended Kalman Filter

For the case of nonlinear dynamics

$$\mathbf{X}_{k+1} = \mathbf{a}_k(\mathbf{X}_k) + \mathbf{G}_k \mathbf{\Gamma}_k \quad (2.42)$$

with a nonlinear observation operator

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{X}_k) + \mathbf{\Upsilon}_k, \quad (2.43)$$

we have noted that the theory behind the Kalman filter breaks down. For this, we introduce the Extended Kalman filter, derived by Stanley F. Schmidt (NASA, 2008), in which we resort to linearizing equations (2.42) and (2.43) about the current state estimate, $\bar{\mathbf{x}}_k$, using a Taylor series expansion. As with the regular Kalman filter, we separate the analysis into a forecast step and an update step:

Update

We modify the analysis step of the Kalman filter to include the nonlinear observation operator:

$$\mathbf{X}^a = \mathbf{X}^f + \mathbf{K} (\mathbf{Y} - \mathbf{h}(\mathbf{X}^f)). \quad (2.44)$$

By linearizing the observation operator about the current estimate for the state vector, $\bar{\mathbf{x}}^f$, we may approximate the above as:

$$\mathbf{X}^a \approx \mathbf{X}^f + \mathbf{K} (\mathbf{Y} - \mathbf{h}(\bar{\mathbf{x}}^f) - \mathbf{H}(\mathbf{X}^f - \bar{\mathbf{x}}^f)), \quad (2.45)$$

where

$$\mathcal{H} = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{X}} \right|_{\mathbf{X}=\bar{\mathbf{x}}^f}. \quad (2.46)$$

Using (2.10), and by noting that through our assumption of unbiased estimators we may write $\bar{\mathbf{x}}^f = \mathbf{x}^t$, we have:

$$\mathbf{x}^t + \Delta^a = \mathbf{x}^t + \Delta^f + \mathbf{K} (\mathbf{y}^t + \Delta^o - \mathbf{h}(\bar{\mathbf{x}}^f) - \mathcal{H}(\Delta^f + \mathbf{x}^t - \bar{\mathbf{x}}^f)) \quad (2.47)$$

$$= \mathbf{x}^t + \Delta^f + \mathbf{K} (\mathbf{y}^t + \Delta^o - \mathbf{h}(\mathbf{x}^t) - \mathcal{H}\Delta^f) \quad (2.48)$$

$$= \mathbf{x}^t + \Delta^f + \mathbf{K} (\Delta^o - \mathcal{H}\Delta^f) + \mathbf{K} (\mathbf{y}^t - \mathbf{h}(\mathbf{x}^t)) \quad (2.49)$$

giving

$$\Delta^a = \Delta^f + \mathbf{K} (\Delta^o - \mathcal{H}\Delta^f). \quad (2.50)$$

With this, and repeating the procedure used for the regular Kalman filter, we consequently obtain:

$$\mathbf{P}^a = (\mathbf{I} - \mathbf{K}\mathcal{H}) \mathbf{P}^f \quad (2.51)$$

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^f + \mathbf{K} (\mathbf{y} - \mathbf{h}(\bar{\mathbf{x}}^f)), \quad (2.52)$$

where

$$\mathbf{K} = \mathbf{P}^f \mathcal{H}^T (\mathcal{H} \mathbf{P}^f \mathcal{H}^T + \mathbf{R})^{-1}. \quad (2.53)$$

Forecast

As with the update equation, we linearize the nonlinear operator in the governing equation about the current state estimate,

$$\mathbf{a}_k(\mathbf{X}_k^a) = \mathbf{a}_k(\bar{\mathbf{x}}_k^a) + \mathcal{A}(\mathbf{X}_k^a - \bar{\mathbf{x}}_k^a) \quad (2.54)$$

where

$$\mathcal{A} = \left. \frac{\partial \mathbf{a}_k}{\partial \mathbf{X}} \right|_{\mathbf{X}=\bar{\mathbf{x}}^a}. \quad (2.55)$$

This gives for the governing equation

$$\mathbf{X}_{k+1}^f = \mathbf{a}_k(\bar{\mathbf{x}}_k^a) + \mathcal{A}(\mathbf{X}_k^a - \bar{\mathbf{x}}_k^a) + \mathbf{G}_k \boldsymbol{\Gamma}_k. \quad (2.56)$$

By taking expectations, we have for the forecast at time $k+1$:

$$\bar{\mathbf{x}}_{k+1}^f = \mathcal{E} [\mathbf{X}_{k+1}^f] \quad (2.57)$$

$$= \mathcal{E} [\mathbf{a}_k(\bar{\mathbf{x}}_k^a) + \mathcal{A}(\mathbf{X}_k^a - \bar{\mathbf{x}}_k^a) + \mathbf{G}_k \boldsymbol{\Gamma}_k] \quad (2.58)$$

$$= \mathbf{a}_k(\bar{\mathbf{x}}_k^a). \quad (2.59)$$

By a similar procedure, using equation (2.10) and again writing $\bar{\mathbf{x}}^a = \mathbf{x}^t$, we have:

$$\mathbf{x}_{k+1}^t + \boldsymbol{\Delta}_{k+1}^f = \mathbf{a}_k(\bar{\mathbf{x}}_k^a) + \mathcal{A}(\boldsymbol{\Delta}_k^a + \mathbf{x}_k^t - \bar{\mathbf{x}}_k^a) + \mathbf{G}_k \boldsymbol{\Gamma}_k \quad (2.60)$$

$$= \mathbf{a}_k(\bar{\mathbf{x}}_k^t) + \mathcal{A}\boldsymbol{\Delta}_k^a + \mathbf{G}_k \boldsymbol{\Gamma}_k \quad (2.61)$$

giving

$$\boldsymbol{\Delta}_{k+1}^f = \mathcal{A}\boldsymbol{\Delta}_k^a + \mathbf{G}_k \boldsymbol{\Gamma}_k. \quad (2.62)$$

Therefore

$$\mathbf{P}_{k+1}^f = \mathcal{E} [\boldsymbol{\Delta}_{k+1}^f (\boldsymbol{\Delta}_{k+1}^f)^T] \quad (2.63)$$

$$= \mathcal{E} [(\mathcal{A}_k \boldsymbol{\Delta}_k^a + \mathbf{G}_k \boldsymbol{\Gamma}_k)(\mathcal{A}_k \boldsymbol{\Delta}_k^a + \mathbf{G}_k \boldsymbol{\Gamma}_k)^T] \quad (2.64)$$

$$= \mathcal{A}_k \mathbf{P}_k^a \mathcal{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T. \quad (2.65)$$

Thus, while we evolve the point estimate for the state nonlinearly according to the true governing equation, the covariance matrix is evolved using linearized dynamics.

In summary, the Extended Kalman filter proceeds as follows:

Definition: Extended Kalman Filter

For the discrete time nonlinear governing equation

$$\mathbf{X}_{k+1} = \mathbf{a}_k(\mathbf{X}_k) + \mathbf{G}_k \mathbf{\Gamma}_k \quad (2.66)$$

with nonlinear observation model

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{X}_k) + \mathbf{\Upsilon}_k, \quad (2.67)$$

perform the following two steps recursively:

1. *Forecast: Given current estimates $\bar{\mathbf{x}}_k^a$ and \mathbf{P}_k^a , obtain estimates at time $k + 1$ using:*

$$\bar{\mathbf{x}}_{k+1}^f = \mathbf{a}_k(\bar{\mathbf{x}}_k^a) \quad (2.68)$$

$$\mathbf{P}_{k+1}^f = \mathbf{A}_k \mathbf{P}_k^a \mathbf{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T, \quad (2.69)$$

where

$$\mathbf{A} = \left. \frac{\partial \mathbf{a}_k}{\partial \mathbf{X}} \right|_{\mathbf{X}=\bar{\mathbf{x}}_k^a}. \quad (2.70)$$

2. *Update: Given estimates $\bar{\mathbf{x}}_k^f$ and \mathbf{P}_k^f , and observation \mathbf{y}_k , with measurement error covariance matrix, \mathbf{R} , update the estimates according to:*

$$\bar{\mathbf{x}}_k^a = \bar{\mathbf{x}}_k^f + \mathbf{K} \left(\mathbf{y}_k - \mathbf{h}(\bar{\mathbf{x}}_k^f) \right) \quad (2.71)$$

$$\mathbf{P}_k^a = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}_k^f, \quad (2.72)$$

where

$$\mathbf{K} = \mathbf{P}_k^f \mathbf{H}^T \left(\mathbf{H} \mathbf{P}_k^f \mathbf{H}^T + \mathbf{R} \right)^{-1} \quad (2.73)$$

and

$$\mathcal{H} = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{X}} \right|_{\mathbf{X}=\bar{\mathbf{x}}_k^f}. \quad (2.74)$$

In practice, the Extended Kalman filter has been found to suffer from instability and ultimately divergence of the estimate from the true solution. It is in general only applicable for weakly nonlinear systems in which the timescales that arise due to nonlinearities are on the order of the time between measurements. Furthermore, the calculation of the Jacobian matrices at each time step become prohibitively expensive as we extend the scheme to systems of increasing complexity. For these reasons, we require a fundamentally different approach to approximating the nonlinear filter. With this, we transition to Monte Carlo methods.

2.3 Ensemble Kalman Filter

While the Extended Kalman filter proved popular for a number of decades, experience showed that it was costly to implement, difficult to tune, and only reliable for systems that were almost linear on the time scale of the updates (Julier and Uhlmann, 1997).

To overcome the need for linearization of the governing equation, Monte Carlo methods were adopted. One such scheme was the Unscented Kalman filter introduced by Julier and Uhlmann (1997), in which a set of $N = 2n + 1$ particles, termed ‘sigma points’, were sampled to deterministically capture the first two moments of the state estimate. These would in turn be evolved in time using the nonlinear governing equation to arrive at the state forecast, from which one would proceed with the assimilation of observations.

For complex systems with large state vectors, however, the handling of $N = 2n + 1$ particles would become unfeasible. The Ensemble Kalman filter, introduced by Evensen (1994), circumvents this issue by using only as many particles as is computationally tractable. The Ensemble Kalman filter proceeds as follows:

Forecast

From an initial estimate of uncertainties or using the analysis of a previous assimilation step, we have in our possession an ensemble of sample points, $\{\mathbf{x}_k^a\} = \{\mathbf{x}_{1,k}^a, \dots, \mathbf{x}_{N,k}^a\}$, representative of the state's probability distribution. We propagate each of these particles forward in time using the governing equation,

$$\mathbf{X}_{k+1}^f = \mathbf{a}_k(\mathbf{X}_k^a) + \mathbf{\Gamma}_k, \quad (2.75)$$

to obtain an ensemble representation for the forecast at time $k + 1$:

$$\{\mathbf{x}_{k+1}^f\} = \{\mathbf{x}_{1,k+1}^f, \dots, \mathbf{x}_{N,k+1}^f\} \quad (2.76)$$

$$= \{\mathbf{a}_k(\mathbf{x}_{1,k}^a) + \gamma_1, \dots, \mathbf{a}_k(\mathbf{x}_{N,k}^a) + \gamma_N\}. \quad (2.77)$$

We note that the γ_i refer to realizations of the noise term generated from its appropriate distribution.

Update

At the time of a new measurement, we update the ensemble in a manner that differs little from that of the regular Kalman filter. Here, however, we estimate \mathbf{P}_k^f using the sample covariance matrix:

$$\hat{\mathbf{P}}_k^f = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_{i,k}^f - \bar{\mathbf{x}}_k^f)(\mathbf{x}_{i,k}^f - \bar{\mathbf{x}}_k^f)^T, \quad (2.78)$$

where

$$\bar{\mathbf{x}}_k^f = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_{i,k}^f. \quad (2.79)$$

With this, we proceed with updating each individual particle in turn as if it were the mean of the original Kalman filter:

$$\mathbf{x}_{i,k}^a = \mathbf{x}_{i,k}^f + \hat{\mathbf{K}}(\mathbf{y}_{i,k} - \mathbf{H}\mathbf{x}_{i,k}^f), \quad i = 1, \dots, N. \quad (2.80)$$

We notice that the update equation differs further from that of the regular Kalman filter in that we create an ensemble of observations to which we add noise:

$$\mathbf{y}_{i,k} = \mathbf{y}_k + \mathbf{v}_i, \quad \boldsymbol{\Upsilon} \sim \mathcal{N}(\mathbf{v}; \mathbf{0}, \mathbf{R}). \quad (2.81)$$

With this, we write for the Kalman gain matrix:

$$\hat{\mathbf{K}} = \hat{\mathbf{P}}_k^f \mathbf{H}^T (\mathbf{H} \hat{\mathbf{P}}_k^f \mathbf{H}^T + \hat{\mathbf{R}})^{-1}, \quad (2.82)$$

where

$$\hat{\mathbf{R}} = \frac{1}{N-1} \sum_{i=1}^N (\mathbf{y}_i - \bar{\mathbf{y}})(\mathbf{y}_i - \bar{\mathbf{y}})^T \quad (2.83)$$

with, as for the ensembles, the mean matrix given by:

$$\bar{\mathbf{y}} = \frac{1}{N} \sum_{i=1}^N \mathbf{y}_i. \quad (2.84)$$

Summary

Since its introduction, the Ensemble Kalman filter has been widely applied. It mainly owes its success to the following two points:

1. While the Ensemble Kalman filter retains the linear update equation of the regular Kalman filter, it acts on the individual ensemble members and thus potentially retains some of the non-Gaussian structure that may initially have been present.
2. As opposed to the Unscented Kalman filter, the Ensemble Kalman filter operates only on a user-specified number of particles, usually significantly less than the dimensionality of the system, i.e. $N \ll n$. While, with this, one clearly only spans a subspace of the full state space, it nonetheless importantly makes the filter computationally tractable.

2.4 Error Subspace Statistical Estimation

In his data assimilation via Error Subspace Statistical Estimation (ESSE), Lermusiaux (1997) suggests further condensing the analysis presented by the Ensemble Kalman filter to a mere subspace of the error covariance matrix, thus focusing only on the dominant structures obtained through an appropriate orthonormal decomposition. By limiting his attention to this reduced space, he disregards less pronounced structures and consequently lessens the computational costs involved.

Rather than using the sample covariance matrix as in the Ensemble Kalman filter, identified by the eigenvalue decomposition

$$\hat{\mathbf{P}} = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^T, \quad (2.85)$$

Lermusiaux proposes to retain only the subspace corresponding to its dominant rank- p reduction, identified by use of the Singular Value Decomposition (SVD). Specifically, carrying on the notation used for the Ensemble Kalman filter, and defining

$$\mathbf{M} = \{\mathbf{x}\} - \{\bar{\mathbf{x}}\}, \quad (2.86)$$

he proceeds by taking its SVD,

$$\text{SVD}_p[\mathbf{M}] = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \quad (2.87)$$

to obtain the p most dominant basis vectors, $\mathbf{E}_p = \mathbf{U}$, with associated eigenvalues, $\mathbf{\Lambda}_p = \frac{1}{N-1}\mathbf{\Sigma}^2$. With this, he arrives at an estimate for the error covariance matrix from which he proceeds with the Kalman update equation in the decomposed form.

Benefiting significantly from these efficiencies, the first real-time ensemble data assimilation done at seas was in the Strait of Sicily in 1996 utilizing ESSE (Lermusiaux, 1999).

2.5 Bayes Filter

In the introduction to this thesis we argued that the most complete description of any field of interest is its probability distribution. When placed in the context of filtering, this optimal (nonlinear) filter is coined the Bayes filter. While its implementation in practice is infeasible, it is nonetheless instructive to provide its mathematical description, since ultimately all data assimilation schemes attempt to approximate this filter. Furthermore, it serves to smoothen the transition to particle filters, described in the next section.

Let us, for simplicity, rewrite the (nonlinear) governing equation as

$$\mathbf{x}_{k+1} = \mathbf{a}_k(\mathbf{x}_k) + \mathbf{\Gamma}_k, \quad (2.88)$$

while keeping the observation model as before:

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{\Upsilon}_k. \quad (2.89)$$

In order to proceed, we require the definition of a Markovian system (see e.g. Bertsekas and Tsitsiklis (2008)):

Definition: Markov Process

A system is Markovian if the probability of a future state depends on the past only through the present. Mathematically, we write:

$$p_{\mathbf{X}_{k+1}|\mathbf{X}_k,\dots,\mathbf{X}_1}(\mathbf{x}_{k+1}|\mathbf{x}_k,\dots,\mathbf{x}_1) = p_{\mathbf{X}_{k+1}|\mathbf{X}_k}(\mathbf{x}_{k+1}|\mathbf{x}_k). \quad (2.90)$$

Clearly, by inspection of equation (2.88), our system is Markovian, allowing us to write for the probability distribution of the state forecast at time $k + 1$:

$$p_{\mathbf{X}_{k+1}^f|\mathbf{X}_k^a,\mathbf{X}_k^f,\dots,\mathbf{X}_1^a,\mathbf{X}_1^f}(\mathbf{x}_{k+1}^f|\mathbf{x}_k^a,\mathbf{x}_k^f,\dots,\mathbf{x}_1^a,\mathbf{x}_1^f) = p_{\mathbf{X}_{k+1}^f|\mathbf{X}_k^a}(\mathbf{x}_{k+1}^f|\mathbf{x}_k^a). \quad (2.91)$$

In fact, by conditioning on the previous analysis vector, this Markovian property

further extends to past observations, thus giving

$$p_{\mathbf{X}_{k+1}^f | \mathbf{X}_k^a, \mathbf{Y}_k, \dots, \mathbf{Y}_1}(\mathbf{x}_{k+1}^f | \mathbf{x}_k^a, \mathbf{y}_k, \dots, \mathbf{y}_1) = p_{\mathbf{X}_{k+1}^f | \mathbf{X}_k^a}(\mathbf{x}_{k+1}^f | \mathbf{x}_k^a). \quad (2.92)$$

By the process of marginalization, we therefore arrive at a lossless description of the forecast at time $k + 1$:

$$p_{\mathbf{X}_{k+1}^f}(\mathbf{x}_{k+1}^f) = \int p_{\mathbf{X}_{k+1}^f, \mathbf{X}_k^a}(\mathbf{x}_{k+1}^f, \mathbf{x}_k^a) d\mathbf{x}_k^a \quad (2.93)$$

$$= \int p_{\mathbf{X}_{k+1}^f | \mathbf{X}_k^a}(\mathbf{x}_{k+1}^f | \mathbf{x}_k^a) p_{\mathbf{X}_k^a}(\mathbf{x}_k^a) d\mathbf{x}_k^a \quad (2.94)$$

$$= \int q_{\mathbf{r}}(\mathbf{x}_{k+1}^f - \mathbf{a}(\mathbf{x}_k^a)) p_{\mathbf{x}_k^a}(\mathbf{x}_k^a) d\mathbf{x}_k^a, \quad (2.95)$$

where $q_{\mathbf{r}}(\mathbf{x}_{k+1}^f - \mathbf{a}(\mathbf{x}_k^a))$ is the probability distribution of the noise term in (2.88).

To arrive at the expression for the update equation, we make use of Bayes law:

$$p_{\mathbf{X}_k^a}(\mathbf{x}_k^a) = p_{\mathbf{X}_k^f | \mathbf{Y}_k}(\mathbf{x}_k^f | \mathbf{y}_k) = \frac{p_{\mathbf{Y}_k | \mathbf{X}_k^f}(\mathbf{y}_k | \mathbf{x}_k^f) p_{\mathbf{X}_k^f}(\mathbf{x}_k^f)}{p_{\mathbf{Y}_k}(\mathbf{y}_k)} \quad (2.96)$$

$$\equiv \eta p_{\mathbf{Y}_k | \mathbf{X}_k^f}(\mathbf{y}_k | \mathbf{x}_k^f) p_{\mathbf{X}_k^f}(\mathbf{x}_k^f) \quad (2.97)$$

$$= \eta q_{\mathbf{r}}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k^f)) p_{\mathbf{X}_k^f}(\mathbf{x}_k^f) \quad (2.98)$$

where $q_{\mathbf{r}}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k^f))$ is the probability distribution of the noise term in the measurement model, equation (2.89), and η is a constant to ensure that (2.98) is valid.

We thus see that through the Markovian property of the governing equation, we arrive at the recursive nature of the optimal nonlinear filter, sequentially applying the forecast equation (2.95) and the update equation (2.98). As expected, for linear dynamics and linear observation models with external Gaussian noise sources the Bayes filter reduces to the Kalman filter (Asada, 2011).

2.6 Particle Filter

The Particle filter attempts to approximate the Bayes filter by representing the probability distribution as a weighted sum of dirac functions:

$$p_{\mathbf{X}}(\mathbf{x}) \approx \sum_{i=1}^n w_i \delta(\mathbf{x} - \mathbf{x}_i), \quad (2.99)$$

where w_i are the individual weights assigned to each particle such that $\sum_{i=1}^n w_i = 1$. With this, the forecast step of the Bayes filter, (2.95), becomes:

$$p_{\mathbf{X}_{k+1}^f}(\mathbf{x}_{k+1}^f) = \int q_{\Gamma}(\mathbf{x}_{k+1}^f - \mathbf{a}(\mathbf{x}_k^a)) p_{\mathbf{X}_k^a}(\mathbf{x}_k^a) d\mathbf{x}_k^a \quad (2.100)$$

$$\approx \sum_{i=1}^n w_i \int q_{\Gamma}(\mathbf{x}_{k+1}^f - \mathbf{a}(\mathbf{x}_k^a)) \delta(\mathbf{x}_k^a - \mathbf{x}_{i,k}^a) d\mathbf{x}_k^a \quad (2.101)$$

$$= \sum_{i=1}^n w_i q_{\Gamma}(\mathbf{x}_{k+1}^f - \mathbf{a}(\mathbf{x}_{i,k}^a)) \quad (2.102)$$

$$\equiv \sum_{i=1}^n w_i \delta(\mathbf{x}_{k+1}^f - \mathbf{x}_{i,k+1}^f), \quad (2.103)$$

where the $\mathbf{x}_{i,k+1}^f$ are drawn from the distribution $q_{\Gamma}(\mathbf{x}_{k+1}^f - \mathbf{a}(\mathbf{x}_{i,k}^a))$. We notice that the particle *weights* do not change during the forecast step.

By a similar procedure, at the time of a new measurement, the updated distribution, (2.98), becomes:

$$p_{\mathbf{X}_k^a}(\mathbf{x}_k^a) = \eta q_{\Upsilon}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k^f)) p_{\mathbf{X}_k^f}(\mathbf{x}_k^f) \quad (2.104)$$

$$\approx \sum_{i=1}^n \eta w_i q_{\Upsilon}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k^f)) \delta(\mathbf{x}_k^f - \mathbf{x}_{i,k}^f) \quad (2.105)$$

$$\equiv \sum_{i=1}^n \hat{w}_i \delta(\mathbf{x}_k^a - \mathbf{x}_{i,k}^f), \quad (2.106)$$

where

$$\hat{w}_i = \eta w_i q_{\Upsilon}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_{i,k}^f)). \quad (2.107)$$

Here, we note that the particle *locations* do not change during the analysis step.

To avoid the collapse of weights onto only a few particles, the particle set is often revised. Most commonly, one proceeds by the method of resampling (Doucet et al., 2001): new particles are generated from the posterior distribution (2.106), occasionally dressing the particles with kernels to avoid co-locating multiple particles. See e.g. van Leeuwen (2009) for a comprehensive exposition on this topic.

With this, we complete our introduction to classical data assimilation schemes. In the following chapter, we describe a number of tools that aim to address the shortcomings made explicit in this chapter. Particularly, we introduce our novel data assimilation scheme, the GMM-DO filter, that efficiently preserves non-Gaussian statistics, all the while respecting nonlinear dynamics.

Chapter 3

Data Assimilation with Gaussian mixture models using the Dynamically Orthogonal field equations

In this chapter, we introduce the core components that ultimately combine to produce the proposed data assimilation scheme: the GMM-DO filter. Particularly, we introduce the following concepts:

- Gaussian mixture models;
- the Expectation-Maximization algorithm;
- the Bayesian Information Criterion; and
- the Dynamically Orthogonal field equations.

Rather than merely stating their definitions, we attempt to justify their choice by providing an explanation of their origins and, where possible, placing them in the context of similar ideas. After providing the details of the GMM-DO filter itself, we conclude the section with a literature review, in which we compare and contrast

our data assimilation scheme against past and more recent filters built on similar foundations.

3.1 Gaussian mixture models

Definition: **Gaussian Mixture Model**

The probability density function for a random vector, $\mathbf{X} \in \mathbb{R}^n$, distributed according to a multivariate Gaussian mixture model is given by

$$p_{\mathbf{X}}(\mathbf{x}) = \sum_{j=1}^M \pi_j \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j, \mathbf{P}_j), \quad (3.1)$$

subject to the constraint that

$$\sum_{j=1}^M \pi_j = 1. \quad (3.2)$$

We refer to $M \in \mathbb{N}$ as the mixture complexity; $\pi_j \in [0, 1]$ as the mixture weights; $\bar{\mathbf{x}}_j \in \mathbb{R}^n$ as the mixture mean vectors; and $\mathbf{P}_j \in \mathbb{R}^{n \times n}$ as the mixture covariance matrices. The multivariate Gaussian density function takes the form:

$$\mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}, \mathbf{P}) = \frac{1}{(2\pi)^{n/2} |\mathbf{P}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})^T \mathbf{P}^{-1}(\mathbf{x}-\bar{\mathbf{x}})}. \quad (3.3)$$

Gaussian mixture models (GMMs) provide an attractive *semiparametric* framework in which to approximate unknown distributions based on available data (McLachlan and Peel, 2000). They can be viewed as a flexible compromise between (a) the fully parametric distribution for which $M = 1$ and (b) the kernel density estimator (see e.g. Silverman (1992)) for which $M = N$, the number of data points. The fully parametric distribution, while justified based on maximum entropy arguments (see e.g. Cover and Thomas (2006)), is often found to enforce too much structure onto the data, being particularly incapable of modeling highly skewed or multimodal data. The kernel density estimator, on the other hand, requires one to retain all N data points for the purposes of inference – a computationally burdensome task. Further-

more, due to the granularity associated with fitting a kernel to every data point, they often necessitate the heuristic choosing of the kernel's shape parameter. We will allude to this phenomenon later in this chapter when presenting the literature review on current data assimilation methods. Mixture models enjoy their popularity by efficiently summarizing the data by a parameter vector, while retaining the ability to accurately model complex distributions (see figure 3-1 for a visual depiction of the three approaches). In fact, it can be shown that in the limit of large complexity and small covariance a Gaussian mixture model converges uniformly to any sufficiently smooth distribution (Alspach and Sorenson, 1972).

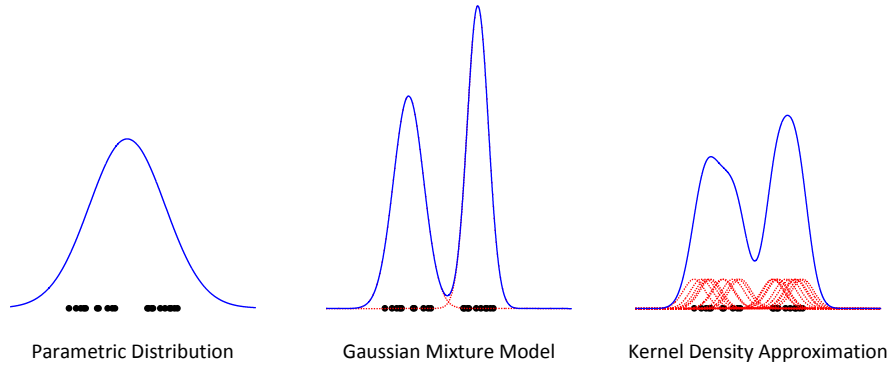


Figure 3-1: Gaussian (parametric) distribution, Gaussian mixture model and Gaussian (kernel) density approximation of 20 samples generated from the mixture of uniform distributions: $p_X(x) = \frac{1}{2} \times \mathcal{U}(x; -8, -1) + \frac{1}{2} \times \mathcal{U}(x; 1, 8)$, where $\mathcal{U}(x; a, b) = \frac{1}{b-a}$ denotes the continuous uniform probability density function for random variable X .

A number of expansions have previously been considered in approximating arbitrary probability distributions, among them the Gram-Charlier expansion, the Edgeworth expansion and Pearson-type density functions (Alspach and Sorenson, 1972). While the former two suffer from being invalid distributions when truncated (namely, that they must integrate to one and be everywhere positive), the latter does not lend itself well to Bayesian inference. In contrast, by equations (3.1) - (3.3), Gaussian mixture models are clearly valid. More importantly, however, for the specific – but popular – case of Gaussian observation models, they make trivial the Bayesian update by invoking the concept of conjugacy, defined as follows (see e.g. Casella and Berger (2001)):

Definition: Conjugate Prior

Let \mathcal{F} denote the class of probability densities $p_{Y|X}(y|x)$. A class \mathcal{G} of prior distributions on X , $p_X(x)$, is a conjugate family for \mathcal{F} if the posterior distribution for X given Y , via Bayes' law, is also in the class \mathcal{G} for all $p_{Y|X}(y|x) \in \mathcal{F}$, all $p_X(x) \in \mathcal{G}$ and all $y \in \mathcal{Y}$.

While the above definition assumes the simple case of univariate random variables living in the same space, the definition trivially extends to that of multivariate random vectors related through a linear operator, i.e. $\mathbf{Y} = \mathbf{H}\mathbf{X}$. For the purposes of simplicity, in the following analysis we restrict our attention to the case of $\mathbf{H} = \mathbf{I}$, the identity operator. When proceeding to introduce the GMM-DO filter, however, we will, for obvious reasons, adopt the more general framework common to ocean and atmospheric applications.

Theorem

A multivariate Gaussian mixture model,

$$p_{\mathbf{X}}(\mathbf{x}) = \sum_{j=1}^M \pi_j^f \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f), \quad (3.4)$$

is a conjugate prior for a multivariate Gaussian observation model,

$$p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{x}, \mathbf{R}). \quad (3.5)$$

Specifically, the posterior distribution equally takes the form of a multivariate Gaussian mixture model,

$$p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y}) = \sum_{j=1}^M \pi_j^a \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^a, \mathbf{P}_j^a), \quad (3.6)$$

with parameters:

$$\begin{aligned}
\pi_j^a &= \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \bar{\mathbf{x}}_j, \mathbf{P}_j^f + \mathbf{R})}{\sum_{i=1}^M \pi_i^f \times \mathcal{N}(\mathbf{y}; \bar{\mathbf{x}}_i, \mathbf{P}_i^f + \mathbf{R})} \\
\bar{\mathbf{x}}_j^a &= \bar{\mathbf{x}}_j^f + \mathbf{P}_j^f (\mathbf{P}_j^f + \mathbf{R})^{-1} (\mathbf{y} - \bar{\mathbf{x}}_j^f) \\
\mathbf{P}_j^a &= (\mathbf{I} - \mathbf{P}_j^f (\mathbf{P}_j^f + \mathbf{R})^{-1}) \mathbf{P}_j^f.
\end{aligned} \tag{3.7}$$

Proof

To avoid cluttering the following analysis, we define the quadratic notation:

$$(\mathbf{a} - \mathbf{b})^T \mathbf{C} (\mathbf{a} - \mathbf{b}) \equiv (\mathbf{a} - \mathbf{b})^T \mathbf{C} (\bullet).$$

We let the prior probability density function take the form of a multivariate Gaussian mixture model,

$$p_{\mathbf{X}}(\mathbf{x}) = \sum_{j=1}^M \pi_j^f \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f),$$

and the observation model that of a multivariate Gaussian distribution,

$$p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{x}, \mathbf{R}).$$

By application of Bayes' Law, we obtain the following posterior distribution:

$$\begin{aligned}
p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y}) &= \frac{p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}) \times p_{\mathbf{X}}(\mathbf{x})}{p_{\mathbf{Y}}(\mathbf{y})} \\
&\propto p_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x}) \times p_{\mathbf{X}}(\mathbf{x}) \\
&= \mathcal{N}(\mathbf{y}; \mathbf{x}, \mathbf{R}) \times \sum_{j=1}^M \pi_j^f \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f) \\
&= \frac{1}{(2\pi)^{n/2} |\mathbf{R}|^{1/2}} e^{-\frac{1}{2}(\mathbf{y}-\mathbf{x})^T \mathbf{R}^{-1} (\bullet)} \times \sum_{j=1}^M \pi_j^f \times \frac{1}{(2\pi)^{n/2} |\mathbf{P}_j^f|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1} (\bullet)} \\
&= \sum_{j=1}^M \frac{\pi_j^f}{(2\pi)^n |\mathbf{R}|^{1/2} |\mathbf{P}_j^f|^{1/2}} e^{-\frac{1}{2}((\mathbf{y}-\mathbf{x})^T \mathbf{R}^{-1} (\bullet) + (\mathbf{x}-\bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1} (\bullet))}.
\end{aligned}$$

By expanding the exponent, we have using the property of symmetric covariance

matrices:

$$\begin{aligned}
& (\mathbf{y} - \mathbf{x})^T \mathbf{R}^{-1}(\bullet) + (\mathbf{x} - \bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1}(\bullet) \\
&= \mathbf{x}^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1}) \mathbf{x} - 2\mathbf{x}^T ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{R}^{-1} \mathbf{y}) + (\bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{y}^T \mathbf{R}^{-1} \mathbf{y}.
\end{aligned}$$

By adding and subtracting $((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j + \mathbf{R}^{-1} \mathbf{y})^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet)$, we get

$$\begin{aligned}
&= \mathbf{x}^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1}) \mathbf{x} - 2\mathbf{x}^T ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{R}^{-1} \mathbf{y}) \\
&\quad + ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j + \mathbf{R}^{-1} \mathbf{y})^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) - ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j + \mathbf{R}^{-1} \mathbf{y})^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) \\
&\quad + (\bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{y}^T \mathbf{R}^{-1} \mathbf{y},
\end{aligned}$$

from which, by completing the square, we obtain

$$\begin{aligned}
&= (\mathbf{x} - ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1} ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{R}^{-1} \mathbf{y}))^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) \\
&\quad - ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{R}^{-1} \mathbf{y})^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) + (\bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{y}^T \mathbf{R}^{-1} \mathbf{y}.
\end{aligned}$$

Finally, by applying the Matrix Inversion Lemma: $(\Sigma_a^{-1} + \Sigma_b^{-1})^{-1} = (\mathbf{I} - \Sigma_a (\Sigma_a + \Sigma_b)^{-1}) \Sigma_a$,

$$\begin{aligned}
&= (\mathbf{x} - (\mathbf{I} - \mathbf{P}_j^f (\mathbf{P}_j^f + \mathbf{R})^{-1}) \mathbf{P}_j^f (\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f - (\mathbf{I} - \mathbf{R} (\mathbf{P}_j^f + \mathbf{R})^{-1}) \mathbf{R} \mathbf{R}^{-1} \mathbf{y})^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) \\
&\quad - ((\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{R}^{-1} \mathbf{y})^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) + (\bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f)^{-1} \bar{\mathbf{x}}_j^f + \mathbf{y}^T \mathbf{R}^{-1} \mathbf{y} \\
&= (\mathbf{x} - (\bar{\mathbf{x}}_j^f + \mathbf{P}_j^f (\mathbf{P}_j^f + \mathbf{R})^{-1} (\mathbf{y} - \bar{\mathbf{x}}_j^f)))^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) \\
&\quad + (\mathbf{y} - \bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f + \mathbf{R})^{-1}(\bullet).
\end{aligned}$$

We therefore have for the posterior distribution:

$$\begin{aligned}
p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y}) &\propto \sum_{j=1}^M \frac{\pi_j^f}{(2\pi)^n |\mathbf{R}|^{1/2} |\mathbf{P}_j^f|^{1/2}} e^{-\frac{1}{2}((\mathbf{x} - \bar{\mathbf{x}}_j^f)^T ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1}(\bullet) + (\mathbf{y} - \bar{\mathbf{x}}_j^f)^T (\mathbf{P}_j^f + \mathbf{R})^{-1}(\bullet))} \\
&= \sum_{j=1}^M \frac{\pi_j^f (2\pi)^n |((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1/2}| |(\mathbf{P}_j^f + \mathbf{R})^{1/2}|}{(2\pi)^n |\mathbf{R}|^{1/2} |\mathbf{P}_j^f|^{1/2}} \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^a, \mathbf{P}_j^a) \times \mathcal{N}(\mathbf{y}; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f + \mathbf{R}) \\
&= \sum_{j=1}^M \pi_j^f \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^a, \mathbf{P}_j^a) \times \mathcal{N}(\mathbf{y}; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f + \mathbf{R})
\end{aligned}$$

which gives

$$p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y}) = \sum_{j=1}^M \pi_j^a \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j^a, \mathbf{P}_j^a),$$

where

$$\begin{aligned} \pi_j^a &= \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f + \mathbf{R})}{\sum_{i=1}^M \pi_i^f \times \mathcal{N}(\mathbf{y}; \bar{\mathbf{x}}_i^f, \mathbf{P}_i^f + \mathbf{R})} \\ \bar{\mathbf{x}}_j^a &= \bar{\mathbf{x}}_j^f + \mathbf{P}_j^f (\mathbf{P}_j^f + \mathbf{R})^{-1} (\mathbf{y} - \bar{\mathbf{x}}_j^f) \\ \mathbf{P}_j^a &= ((\mathbf{P}_j^f)^{-1} + \mathbf{R}^{-1})^{-1} = (\mathbf{I} - \mathbf{P}_j^f (\mathbf{P}_j^f + \mathbf{R})^{-1}) \mathbf{P}_j^f. \quad \square \end{aligned}$$

Consequently, for Gaussian observation models with Gaussian mixture models as priors, the usually intractable Bayesian update reduces to a trivial update of the elements of the parameter set, $\{\pi_1, \dots, \pi_M, \bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_M, \mathbf{P}_1, \dots, \mathbf{P}_M\}$, given by (3.7). Specifically, the individual mixture means and covariance matrices are conveniently updated in accordance with the already familiar Kalman filter, coupled solely through their mixture weights.

Having introduced Gaussian mixture models as an attractive method for approximating distributions for the purposes of Bayesian inference, it remains for us to determine its optimal set of parameters, $\{\pi_1, \dots, \pi_M, \bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_M, \mathbf{P}_1, \dots, \mathbf{P}_M\}_{\text{optimal}}$, based on the data at hand. Particularly, we seek the value for the parameters that maximizes the probability of obtaining the given data; the Maximum Likelihood estimators. For this we make use of the Expectation-Maximization algorithm.

3.2 The EM algorithm

The following exposition on the Expectation-Maximization (EM) algorithm is largely based on the MIT class notes by Jaakkola (2006) and Wornell (2010) as well as the books by McLachlan and Basford (1988), McLachlan and Krishnan (1997) and McLachlan and Peel (2000).

The EM algorithm describes an iterative procedure for estimating the parameters of a target distribution that maximize the probability of obtaining the available data, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, thus arriving at the Maximum Likelihood (ML) estimate for the unknown set of parameters. While ML estimators can be justified on intuition alone, they can further be shown to be both consistent and asymptotically efficient (Bertsekas and Tsitsiklis, 2008) and are thus particularly attractive.

For most realistic cases, obtaining the ML estimate by differentiating the parametric probability distribution, $p_{\{\mathbf{x}\}}(\{\mathbf{x}\}; \theta_1, \dots, \theta_M)$ say, with respect to the parameter of interest and equating with zero,

$$\frac{\partial p_{\{\mathbf{x}\}}(\{\mathbf{x}\}; \theta_1, \dots, \theta_M)}{\partial \theta_i} = 0, \quad i = 1, \dots, M, \quad (3.8)$$

results in a nonlinear equation that lacks a closed form solution (Wornell, 2010). In such cases, one must resort to numerical optimization methods. While various hill-climbing schemes exist, the EM algorithm takes advantage of the particular properties associated with probability distributions from which it ultimately enjoys its simplicity.

In literature, the EM algorithm is commonly introduced in the context of ‘incomplete data’, for which ML parameter estimation by the method of partial differentiation, as described above, can be a difficult task. The primary step is thus to artificially *complete* the data at hand with additional pseudo data (or knowledge about the existing data), thereby giving rise to a simpler structure for which ML estimation is made computationally more tractable. Specifically, the complete data problem typically yields a closed form solution to the estimation problem (McLachlan and Peel, 2000), allowing one to obtain the ML parameters by simple partial differentiation. The data with which to complete the existing data set is chosen by the user and may have little physical relevance; its choice, however, ultimately dictates the efficiency of the algorithm. By conditioning the complete data on the available data, an estimate for the ML parameters may iteratively be obtained. This procedure lies at the heart of the EM algorithm, to be described in detail in what follows.

Following Wornell (2010), we let $\{x\} = \{x_1, \dots, x_N\}^T$ denote the set of available

data, $\{z\}$ the *complete* data vector and $\boldsymbol{\theta} = \{\theta_1, \dots, \theta_M\}^T$ the set of parameters (to be determined) of the *chosen* distributional form, $p_{\{Z\}}(\{z\}; \boldsymbol{\theta})$. We further assume, as is often the case, that the available data is a unique and deterministic function of the complete data, i.e. $\{x\} = g(\{z\})$. (For instance, this may simply be a subset of the complete data.) By the Total Probability Theorem (see e.g. Bertsekas and Tsitsiklis (2008)), we may thus write:

$$p_{\{Z\}}(\{z\}; \boldsymbol{\theta}) = \sum_{\{x\}} p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \boldsymbol{\theta}) \times p_{\{X\}}(\{x\}; \boldsymbol{\theta}) \quad (3.9)$$

$$= p_{\{Z\}|\{X\}}(\{z\}|g(\{z\}); \boldsymbol{\theta}) \times p_{\{X\}}(g(\{z\}); \boldsymbol{\theta}). \quad (3.10)$$

By taking logarithms, we consequently obtain for *any value of $\{z\}$ that satisfies $\{x\} = g(\{z\})$* :

$$\log(p_{\{X\}}(\{x\}; \boldsymbol{\theta})) = \log(p_{\{Z\}}(\{z\}; \boldsymbol{\theta})) - \log(p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \boldsymbol{\theta})). \quad (3.11)$$

By further taking expectations with respect to the complete data, conditioned on the available data and parameterized by an *arbitrary* vector $\tilde{\boldsymbol{\theta}}$ (to be optimized), i.e.

$$\mathcal{E}[(\bullet) | \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] = \int_{\{z\}} (\bullet) p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \tilde{\boldsymbol{\theta}}) d\{z\}, \quad (3.12)$$

the left hand side of equation (3.11) remains unaffected,

$$\mathcal{E}[\log(p_{\{X\}}(\{x\}; \boldsymbol{\theta})) | \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] = \log(p_{\{X\}}(\{x\}; \boldsymbol{\theta})), \quad (3.13)$$

and we thus obtain

$$\begin{aligned} \log(p_{\{X\}}(\{x\}; \boldsymbol{\theta})) &= \mathcal{E}[\log(p_{\{Z\}}(\{z\}; \boldsymbol{\theta})) | \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] \\ &\quad - \mathcal{E}[\log(p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \boldsymbol{\theta})) | \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}]. \end{aligned} \quad (3.14)$$

For the sake of convenience, we denote

$$U(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) = \mathcal{E} [\log (p_{\{Z\}}(\{z\}; \boldsymbol{\theta})) \mid \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] \quad (3.15)$$

$$V(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) = -\mathcal{E} [\log (p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \boldsymbol{\theta})) \mid \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] \quad (3.16)$$

to obtain the simplified expression

$$\log (p_{\{X\}}(\{x\}; \boldsymbol{\theta})) = U(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) + V(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}). \quad (3.17)$$

By application of Gibbs' inequality (see the appendix), we see that

$$V(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) = -\mathcal{E} [\log (p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \boldsymbol{\theta})) \mid \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] \quad (3.18)$$

$$\geq -\mathcal{E} [\log (p_{\{Z\}|\{X\}}(\{z\}|\{x\}; \tilde{\boldsymbol{\theta}})) \mid \{X\} = \{x\}; \tilde{\boldsymbol{\theta}}] \quad (3.19)$$

$$\equiv V(\tilde{\boldsymbol{\theta}}; \tilde{\boldsymbol{\theta}}). \quad (3.20)$$

Therefore, if we denote $\tilde{\boldsymbol{\theta}}$ as our *present* estimate for the parameter vector, by *choosing* $\boldsymbol{\theta} \neq \tilde{\boldsymbol{\theta}}$ such that it further satisfies $U(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) \geq U(\tilde{\boldsymbol{\theta}}; \tilde{\boldsymbol{\theta}})$, we guarantee that

$$\log (p_{\{X\}}(\{x\}; \boldsymbol{\theta})) = U(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) + V(\boldsymbol{\theta}; \tilde{\boldsymbol{\theta}}) \quad (3.21)$$

$$\geq U(\tilde{\boldsymbol{\theta}}; \tilde{\boldsymbol{\theta}}) + V(\tilde{\boldsymbol{\theta}}; \tilde{\boldsymbol{\theta}}) \quad (3.22)$$

$$= \log (p_{\{X\}}(\{x\}; \tilde{\boldsymbol{\theta}})). \quad (3.23)$$

Consequently, upon repeated iterations, our estimate for the parameter vector monotonically increases the log likelihood of generating the data at hand. Assuming further that the likelihood is bounded from above, we are thus guaranteed to converge to a stationary point and as such obtain an estimate for the ML parameter vector (Casella and Berger, 2001).

In summary, the EM algorithm proceeds as follows:

Definition: The EM algorithm

Given the available data, $\{x\} = \{x_1, \dots, x_N\}$, initial parameter estimate $\boldsymbol{\theta}^{(0)}$, pro-

posed complete data vector $\{z\}$ with predetermined, user-specified distribution, $p_{\{Z\}}(\{z\}; \boldsymbol{\theta})$, repeat until convergence:

- Using the present parameter estimate $\boldsymbol{\theta}^{(k)}$, form

$$U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)}) = \mathcal{E} [\log (p_{\{Z\}}(\{z\}; \boldsymbol{\theta})) \mid \{X\} = \{x\}; \boldsymbol{\theta}^{(k)}]. \quad (3.24)$$

- Update the estimate for the parameter vector, $\boldsymbol{\theta}^{(k+1)}$, by maximizing $U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})$:

$$\boldsymbol{\theta}^{(k+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} (U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})). \quad (3.25)$$

For the purposes of this thesis, it is instructive to illustrate the application of the EM algorithm to multivariate Gaussian mixture models. We do so as follows.

3.2.1 The EM algorithm with Gaussian mixture models

We assume that we have the set of data, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, generated by the multivariate Gaussian mixture distribution:

$$p_{\mathbf{X}}(\mathbf{x}) = \sum_{j=1}^M \pi_j \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j, \mathbf{P}_j) \quad (3.26)$$

for which we wish to obtain the maximum likelihood estimate for the parameter vector:

$$\boldsymbol{\theta} = \{\pi_1, \dots, \pi_M, \bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_M, \mathbf{P}_1, \dots, \mathbf{P}_M\}. \quad (3.27)$$

Note the convenient abuse of notation of allowing the parameter vector to contain both non-transposed vectors as well as full matrices. Note, also, that for the moment, we assume the mixture complexity to be fixed and known. (In the subsequent section we will introduce a method that estimates the optimal choice for M based on the data at hand.) We augment the available data set to form the *complete* data set

$$\{z\} = \{c_1, \mathbf{x}_1, \dots, c_N, \mathbf{x}_N\}, \quad (3.28)$$

where \mathbf{c}_i represents an indicator vector such that

$$(\mathbf{c}_i)_j = \begin{cases} 1 & \text{if data point } \mathbf{x}_i \text{ was generated by mixture } j, \\ 0 & \text{otherwise,} \end{cases} \quad (3.29)$$

with $(\mathbf{c}_i)_j$ referring to the j^{th} element of vector \mathbf{c}_i . (We note that, for our purposes, these membership indicators have little physical relevance, and exist merely as a conceptual device within the EM framework.) Conditioned on the additional knowledge of the set $\{\mathbf{c}\} = \{\mathbf{c}_1, \dots, \mathbf{c}_N\}$, we therefore assume we know the origin of each realization, namely the mixture that generated it. This knowledge gives rise to closed form solutions for the Maximum Likelihood estimator of the parameter vector, specifically:

$$\pi_j = \frac{N_j}{N} \quad (3.30)$$

$$\bar{\mathbf{x}}_j = \frac{1}{N_j} \sum_{i=1}^N (\mathbf{c}_i)_j \times \mathbf{x}_i \quad (3.31)$$

$$\mathbf{P}_j = \frac{1}{N_j} \sum_{i=1}^N (\mathbf{c}_i)_j \times (\mathbf{x}_i - \bar{\mathbf{x}}_j)(\mathbf{x}_i - \bar{\mathbf{x}}_j)^T, \quad (3.32)$$

where

$$N_j \equiv \sum_{i=1}^N (\mathbf{c}_i)_j. \quad (3.33)$$

We have thus *completed* the data vector. Rather than hardwiring a realization to a particular mixture as done with the complete data set, however, in the EM algorithm we successively estimate the weights with which a given realization is associated with each of the mixtures based on the present parameter estimates. This is followed by optimizing the parameter vector based on the previously calculated weights. With this, we ultimately arrive at an estimate for the Maximum Likelihood parameter vector based on the available data set, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$.

By the assumed independence of the data, the probability distribution for the

complete data takes the form

$$p_{\{\mathbf{Z}\}}(\{\mathbf{z}\}; \boldsymbol{\theta}) = \prod_{i=1}^N p_{C_i, \mathbf{X}_i}(\mathbf{c}_i, \mathbf{x}_i; \boldsymbol{\theta}) \quad (3.34)$$

$$= \prod_{i=1}^N \prod_{j=1}^M (\pi_j \times \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j, \mathbf{P}_j))^{(\mathbf{c}_i)_j}. \quad (3.35)$$

(Note that in arriving at equation (3.35), we adopted a common trick allowed by the use of categorical random variables. See e.g. Bertsekas and Tsitsiklis (2008)) Upon taking logarithms we obtain

$$\log(p_{\{\mathbf{Z}\}}(\{\mathbf{z}\}; \boldsymbol{\theta})) = \sum_{i=1}^N \sum_{j=1}^M (\mathbf{c}_i)_j (\log \pi_j + \log \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j, \mathbf{P}_j)). \quad (3.36)$$

By further taking the conditional expectation of equation (3.36) with respect to the available data, arbitrarily parameterized by vector $\boldsymbol{\theta}^{(k)}$,

$$\mathcal{E}[(\bullet) \mid \{\mathbf{X}\} = \{\mathbf{x}\}; \boldsymbol{\theta}^{(k)}] = \int_{\{\mathbf{z}\}} (\bullet) p_{\{\mathbf{Z}\}|\{\mathbf{X}\}}(\{\mathbf{z}\}|\{\mathbf{x}\}; \boldsymbol{\theta}^{(k)}) d\{\mathbf{z}\} \quad (3.37)$$

we consequently obtain the expression to be maximized under the EM algorithm:

$$U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)}) = \mathcal{E}[\log(p_{\{\mathbf{Z}\}}(\{\mathbf{z}\}; \boldsymbol{\theta})) \mid \{\mathbf{X}\} = \{\mathbf{x}\}; \boldsymbol{\theta}^{(k)}] \quad (3.38)$$

$$= \mathcal{E}\left[\sum_{i=1}^N \sum_{j=1}^M (\mathbf{c}_i)_j (\log \pi_j + \log \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j, \mathbf{P}_j)) \mid \{\mathbf{X}\} = \{\mathbf{x}\}; \boldsymbol{\theta}^{(k)}\right] \quad (3.39)$$

$$= \sum_{i=1}^N \sum_{j=1}^M \mathcal{E}[(\mathbf{c}_i)_j \mid \{\mathbf{X}\} = \{\mathbf{x}\}; \boldsymbol{\theta}^{(k)}] (\log \pi_j + \log \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j, \mathbf{P}_j)). \quad (3.40)$$

Finally, for convenience of notation, we define

$$\tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \equiv \mathcal{E}[(\mathbf{c}_i)_j \mid \{\mathbf{X}\} = \{\mathbf{x}\}; \boldsymbol{\theta}^{(k)}] \quad (3.41)$$

$$= \mathcal{E}[(\mathbf{c}_i)_j \mid \mathbf{X}_i = \mathbf{x}_i; \boldsymbol{\theta}^{(k)}] \quad (3.42)$$

$$= \Pr((\mathbf{c}_i)_j = 1 \mid \mathbf{X}_i = \mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \quad (3.43)$$

$$= \frac{\pi_j^{(k)} \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j^{(k)}, \mathbf{P}_j^{(k)})}{\sum_{m=1}^M \pi_m^{(k)} \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_m^{(k)}, \mathbf{P}_m^{(k)})}, \quad (3.44)$$

where we made use of the definition for the expectation of a Bernoulli random variable in going from equation (3.42) to (3.43) (see e.g. Bertsekas and Tsitsiklis (2008)). This completes the E-step of the EM algorithm.

We proceed with evaluating $\boldsymbol{\theta}^{(k+1)}$, the parameter vector $\boldsymbol{\theta}$ which maximizes $U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})$. This forms the M-step of the EM algorithm. (The following analysis is rarely given in the literature. In fact, we failed to find a single account of its complete derivation in connection with the EM algorithm. It is given here since it serves to illustrate the simple use of matrix calculus (see e.g. Petersen and Pedersen (2008)) by which the final result is achieved.) By expansion of the multivariate normal distribution, $U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})$ takes the form:

$$U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)}) = \sum_{i=1}^N \sum_{j=1}^M \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\log \pi_j + \log \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j, \mathbf{P}_j)) \quad (3.45)$$

$$= \sum_{j=1}^M \sum_{i=1}^N \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \left(\log \pi_j - \frac{k}{2} \log 2\pi \right. \\ \left. - \frac{1}{2} \log |\mathbf{P}_j| - \frac{1}{2} (\mathbf{x}_i - \bar{\mathbf{x}}_j)^T \mathbf{P}_j^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_j) \right). \quad (3.46)$$

In what follows, we proceed by the method of Lagrange multipliers (see e.g. Cover and Thomas (2006)) for optimizing the terms of the parameter vector in which constraints exist. (Such is for instance the case for π_j where we require $\sum_{j=1}^M \pi_j = 1$. In a later discussion, we will further place a restriction on the mixture means, $\bar{\mathbf{x}}_j$.) For the unrestricted terms of the parameter vector, we determine the optimal parameters by regular differentiation.

To determine $\pi_j^{(k+1)}$, we introduce the auxiliary function with Lagrange multiplier, λ :

$$\begin{aligned} \Lambda = & \sum_{j=1}^M \sum_{i=1}^N \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\log \pi_j - \frac{k}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{P}_j| \\ & - \frac{1}{2} (\mathbf{x}_i - \bar{\mathbf{x}}_j)^T \mathbf{P}_j^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_j)) + \lambda (\sum_{k=1}^M \pi_k - 1). \end{aligned} \quad (3.47)$$

from which, upon equating the gradient with the zero vector,

$$\frac{\partial \Lambda}{\partial \pi_p} = 0 \quad \forall p \in \{1, 2, \dots, M\} \quad \text{and} \quad \frac{\partial \Lambda}{\partial \lambda} = 0, \quad (3.48)$$

we obtain the expression:

$$\left(\frac{1}{\pi_p} \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) + \lambda, \sum_{k=1}^M \pi_k - 1 \right) = (0, 0) \quad \forall p \in \{1, 2, \dots, M\}. \quad (3.49)$$

By observation, since the above holds for all p , and λ is unique, we thus have the condition:

$$\frac{1}{\pi_p} \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) = \frac{1}{\pi_q} \sum_{i=1}^N \tau_q(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \quad \forall p, q \in \{1, 2, \dots, M\} \quad (3.50)$$

or equivalently

$$\pi_q \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) = \pi_p \sum_{i=1}^N \tau_q(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \quad \forall p, q \in \{1, 2, \dots, M\}. \quad (3.51)$$

Upon summing over q , we obtain for the left hand side of (3.51):

$$\sum_{q=1}^M \pi_q \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) = \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \sum_{q=1}^M \pi_q \quad (3.52)$$

$$= \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \quad (3.53)$$

and similarly for the right hand side of (3.51):

$$\sum_{q=1}^M \pi_p^{(k+1)} \sum_{i=1}^N \tau_q(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) = \pi_p^{(k+1)} \sum_{i=1}^N \sum_{q=1}^M \tau_q(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \quad (3.54)$$

$$= \pi_p^{(k+1)} \sum_{i=1}^N 1 \quad (3.55)$$

$$= N \times \pi_p^{(k+1)} \quad (3.56)$$

and therefore arrive at the final expression for the updated mixture weights:

$$\pi_p^{(k+1)} = \frac{\sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)})}{N} \equiv \frac{N_p^{(k)}}{N}, \quad (3.57)$$

where $N_p^{(k)}$ is the sum total of particles associated with a given mixture, p , under the current parameter vector, $\boldsymbol{\theta}^{(k)}$.

With this, we proceed to determine the unconstrained parameters $\bar{\mathbf{x}}_p^{(k+1)}$ and $\mathbf{P}_p^{(k+1)}$ simply by taking the appropriate partial derivatives and equating with zero. Specifically, to obtain $\bar{\mathbf{x}}_p^{(k+1)}$, we have:

$$\begin{aligned} \frac{\partial U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})}{\partial \bar{\mathbf{x}}_p} &= \frac{\partial}{\partial \bar{\mathbf{x}}_p} \left(\sum_{j=1}^M \sum_{i=1}^N \left(\tau_i(\mathbf{x}_j; \boldsymbol{\theta}^{(k)}) \left(\log \pi_j - \frac{k}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{P}_j| \right. \right. \right. \\ &\quad \left. \left. \left. - \frac{1}{2} (\mathbf{x}_i - \bar{\mathbf{x}}_j)^T \mathbf{P}_j^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_j) \right) \right) \right) \\ &= -\frac{1}{2} \sum_{i=1}^N \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \frac{\partial}{\partial \bar{\mathbf{x}}_p} \left((\mathbf{x}_i - \bar{\mathbf{x}}_p)^T \mathbf{P}_p^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_p) \right) \right) \\ &= \sum_{i=1}^N \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\mathbf{x}_i - \bar{\mathbf{x}}_p)^T \mathbf{P}_p^{-1} \right) \\ &= \mathbf{0} \end{aligned} \quad (3.58)$$

giving

$$\bar{\mathbf{x}}_p^{(k+1)} = \frac{1}{N_p^{(k)}} \sum_{i=1}^N \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \mathbf{x}_i \right), \quad (3.59)$$

and, similarly, to obtain $\mathbf{P}_p^{(k+1)}$, we have (with knowledge of $\bar{\mathbf{x}}_p^{(k+1)}$):

$$\begin{aligned}
\frac{\partial U(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})}{\partial \mathbf{P}_p} &= \frac{\partial}{\partial \mathbf{P}_p} \left(\sum_{j=1}^M \sum_{i=1}^N \left(\tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \left(\log \pi_j - \frac{k}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{P}_j| \right. \right. \right. \\
&\quad \left. \left. \left. - \frac{1}{2} (\mathbf{x}_i - \bar{\mathbf{x}}_j^{(k+1)})^T \mathbf{P}_j^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_j^{(k+1)}) \right) \right) \right) \\
&= -\frac{1}{2} \sum_{i=1}^N \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \frac{\partial}{\partial \mathbf{P}_p} \left(\log |\mathbf{P}_p| + (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)})^T \mathbf{P}_p^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)}) \right) \right) \\
&= -\frac{1}{2} \sum_{i=1}^N \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\mathbf{P}_p^{-T} + \mathbf{P}_p^{-T} (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)}) (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)})^T \mathbf{P}_p^{-T}) \right) \\
&= -\frac{1}{2} \sum_{i=1}^N \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\mathbf{P}_p^{-1} + \mathbf{P}_p^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)}) (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)})^T \mathbf{P}_p^{-1}) \right) \\
&= -\frac{1}{2} \sum_{i=1}^N \mathbf{P}_p^{-1} \left(\tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\mathbf{P}_p + (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)}) (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)})^T) \right) \mathbf{P}_p^{-1} \\
&= \mathbf{0}
\end{aligned} \tag{3.60}$$

giving

$$\mathbf{P}_p^{(k+1)} = \frac{1}{N_p^{(k)}} \sum_{i=1}^N \tau_p(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)}) (\mathbf{x}_i - \bar{\mathbf{x}}_p^{(k+1)})^T. \tag{3.61}$$

In summary, the EM algorithm for Gaussian mixture models proceeds as follows:

Definition: The EM algorithm for Gaussian mixture models

Given the available data, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, and initial parameter estimate,

$$\boldsymbol{\theta}^{(0)} = \{\pi_1^{(0)}, \dots, \pi_M^{(0)}, \bar{\mathbf{x}}_1^{(0)}, \dots, \bar{\mathbf{x}}_M^{(0)}, \mathbf{P}_1^{(0)}, \dots, \mathbf{P}_M^{(0)}\}, \tag{3.62}$$

repeat until convergence:

- For all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$, using the present parameter estimate $\boldsymbol{\theta}^{(k)}$, form

$$\tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) = \frac{\pi_j^{(k)} \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_j^{(k)}, \mathbf{P}_j^{(k)})}{\sum_{m=1}^M \pi_m^{(k)} \mathcal{N}(\mathbf{x}_i; \bar{\mathbf{x}}_m^{(k)}, \mathbf{P}_m^{(k)})}. \tag{3.63}$$

- For all $j \in \{1, 2, \dots, M\}$, update the parameter estimate $\boldsymbol{\theta}^{(k+1)}$ according to

$$\pi_j^{(k+1)} = \frac{N_j^{(k)}}{N} \quad (3.64)$$

$$\bar{\mathbf{x}}_j^{(k+1)} = \frac{1}{N_j^{(k)}} \sum_{i=1}^N \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \mathbf{x}_i \quad (3.65)$$

$$\mathbf{P}_j^{(k+1)} = \frac{1}{N_j^{(k)}} \sum_{i=1}^N \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) (\mathbf{x}_i - \bar{\mathbf{x}}_j^{(k+1)}) (\mathbf{x}_i - \bar{\mathbf{x}}_j^{(k+1)})^T \quad (3.66)$$

where

$$N_j^{(k)} = \sum_{i=1}^N \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}). \quad (3.67)$$

3.2.2 Remarks

Before concluding this section, we wish to address two issues regarding the EM algorithm, namely the choice of starting parameters and the issue of convergence.

The goal of the EM algorithm is to obtain the Maximum Likelihood estimate for the parameter vector, namely the parameter vector that globally maximizes the probability of obtaining the data. We've already noted, however, that the EM algorithm is guaranteed to converge only to a stationary point. Thus, for a likelihood with several stationary points, convergence to either a local or global maximum depends on the initial estimate for the parameter vector. While this starting point may be chosen deterministically through a crude but efficient clustering algorithm (e.g. *Affinity Propagation* due to Frey and Dueck (2007)), such methods fail to explore the parameter space and therefore risk converging to a local maximum. Instead, one commonly runs the EM algorithm from a number of starting points chosen stochastically, but based somehow on the data spread. The optimal parameter vector is then chosen as the one that maximizes the probability of the data over all runs.

An issue arises here, however. The global maximum, corresponding to the ML estimate, often arises in the interior of the parameter space. For Gaussian mixture models, however, the likelihood is unbounded on the edge of the parameter space in

which a Gaussian with zero covariance is fitted to a data point, thus giving rise to a singularity however small the mixture weight (McLachlan and Peel, 2000). Therefore consideration has to be given to large local maxima that occur as a consequence of small (but nonzero) covariance matrices. Such components correspond to mixtures containing few data points close together or almost residing in a lower dimensional subspace. A condition for avoiding such occurrences may be given by:

$$\frac{|\Sigma_i|}{|\Sigma_j|} \geq C > 0, \quad \forall i, j \in \{1, \dots, M\}$$

with C appropriately chosen. McLachlan and Peel (2000) note that a more informative approach is to examine the actual eigenvalues of the covariance matrices as these offer a better reflection of the shapes of the mixtures. Particularly, in this way one may differentiate between small compact clusters and long thin clusters, potentially indicating viable and less viable mixtures.

3.3 The Bayesian Information Criterion

Determining the optimal complexity of a Gaussian mixture model can be a complicated task, particularly given limited *a priori* knowledge, and is often guided by empirical evidence, namely the available data. Such a task is formally referred to as model selection, and while numerous schemes exist (see e.g. McLachlan and Peel (2000)), for the purposes of this thesis, we will restrict our attention to the metric defined by the popular Bayesian Information Criterion (BIC). Much of this exposition is due to Wornell (2010) and McLachlan and Peel (2000).

As the name suggests, the Bayesian Information Criterion is most easily introduced in a Bayesian framework. Specifically, for the time being, we let the parameter vector, θ , be random. We will see that under the assumption of sufficiently large data sets, however, the arbitrarily assigned prior distribution will have little effect. As such, we ultimately remain within the framework of classical statistics, and for this reason we let the mixture complexity, M , be constant but unknown.

We introduce the following notation: $p_{\Theta}(\boldsymbol{\theta}; M)$ defines the prior distribution over the parameter vector for a given mixture complexity, and $p_{\mathbf{X}|\Theta}(\mathbf{x}|\boldsymbol{\theta}; M)$ is the probability distribution for the available data conditioned on a parameter vector at a given complexity. In this thesis, the latter obviously takes the form of a Gaussian mixture model, however we will leave the following analysis in its generic form. To guide the reader, we stress that both of the aforementioned distributions are known up to the value of the parameter vector, $\boldsymbol{\theta}$.

We wish to select the model complexity, M , that maximizes the likelihood of obtaining the available data, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$. In other words, by the assumed independence of the data points, we seek M for which

$$p_{\{\mathbf{X}\}}(\{\mathbf{x}\}; M) = \prod_{i=1}^N p_{\mathbf{X}_i}(\mathbf{x}_i; M) \quad (3.68)$$

is a maximum. Continuing the analysis for a single data point, \mathbf{x}_i , we have by Bayes' Law that

$$p_{\Theta|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M) = \frac{p_{\mathbf{X}_i|\Theta}(\mathbf{x}_i|\boldsymbol{\theta}; M)p_{\Theta}(\boldsymbol{\theta}; M)}{p_{\mathbf{X}_i}(\mathbf{x}_i; M)} \quad (3.69)$$

where

$$p_{\mathbf{X}_i}(\mathbf{x}_i; M) = \int p_{\mathbf{X}_i|\Theta}(\mathbf{x}_i|\boldsymbol{\theta}; M)p_{\Theta}(\boldsymbol{\theta}; M)d\boldsymbol{\theta}. \quad (3.70)$$

Using Laplace's approximation (see e.g. Wornell (2010)), we expand the logarithm of the left hand side of (3.69) by a Taylor series about an arbitrary parameter value, $\hat{\boldsymbol{\theta}}$:

$$\begin{aligned} \ln(p_{\Theta|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M)) &= \ln(p_{\Theta|\mathbf{X}_i}(\hat{\boldsymbol{\theta}}|\mathbf{x}_i; M)) + \frac{\partial}{\partial \boldsymbol{\theta}} \ln(p_{\Theta|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M)) \Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) \\ &+ \frac{1}{2} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T \frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln(p_{\Theta|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M)) \Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) + \dots \end{aligned} \quad (3.71)$$

By application of the EM algorithm introduced in the previous section, however, we may choose $\hat{\boldsymbol{\theta}}$ to be the ML estimate for the parameter vector, $\hat{\boldsymbol{\theta}}_{ML}$, such that the following holds:

$$\frac{\partial}{\partial \boldsymbol{\theta}} \ln(p_{\Theta|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M)) \Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{ML}} = 0. \quad (3.72)$$

Defining further the Fisher information in the data \mathbf{x}_i about the parameter vector $\boldsymbol{\theta}$,

$$\mathbf{J}_{\mathbf{X}_i=\mathbf{x}_i}(\hat{\boldsymbol{\theta}}_{ML}) = -\frac{\partial^2}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T} \ln(p_{\boldsymbol{\Theta}|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M)) \Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}_{ML}}, \quad (3.73)$$

using (3.71) we thus arrive at the approximation

$$p_{\boldsymbol{\Theta}|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M) \approx p_{\boldsymbol{\Theta}|\mathbf{X}_i}(\hat{\boldsymbol{\theta}}_{ML}|\mathbf{x}_i; M) e^{-\frac{1}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{ML})^T \mathbf{J}_{\mathbf{X}_i=\mathbf{x}_i}(\hat{\boldsymbol{\theta}}_{ML})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{ML})} \quad (3.74)$$

$$= \frac{p_{\mathbf{X}_i|\boldsymbol{\Theta}}(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_{ML}; M) p_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\theta}}_{ML}; M)}{p_{\mathbf{X}_i}(\mathbf{x}_i; M)} e^{-\frac{1}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{ML})^T \mathbf{J}_{\mathbf{X}_i=\mathbf{x}_i}(\hat{\boldsymbol{\theta}}_{ML})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{ML})} \quad (3.75)$$

by applying (3.69) evaluated at $\hat{\boldsymbol{\theta}}_{ML}$. By integrating over $\boldsymbol{\theta}$, we obtain the relations

$$\int p_{\boldsymbol{\Theta}|\mathbf{X}_i}(\boldsymbol{\theta}|\mathbf{x}_i; M) d\boldsymbol{\theta} = 1, \quad \text{and} \quad (3.76)$$

$$\int e^{-\frac{1}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{ML})^T \mathbf{J}_{\mathbf{X}_i=\mathbf{x}_i}(\hat{\boldsymbol{\theta}}_{ML})(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}_{ML})} d\boldsymbol{\theta} = (2\pi)^{\frac{K_m}{2}} |\mathbf{J}_{\mathbf{X}_i=\mathbf{x}_i}(\hat{\boldsymbol{\theta}}_{ML})|^{-\frac{1}{2}} \quad (3.77)$$

from which, upon rearranging the terms in equation (3.75), we have that

$$p_{\mathbf{X}_i}(\mathbf{x}_i; M) \approx p_{\mathbf{X}_i|\boldsymbol{\Theta}}(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_{ML}; M) p_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\theta}}_{ML}; M) (2\pi)^{\frac{K_m}{2}} |\mathbf{J}_{\mathbf{X}_i=\mathbf{x}_i}(\hat{\boldsymbol{\theta}}_{ML})|^{-\frac{1}{2}}. \quad (3.78)$$

Thus for N independent realizations, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, we have

$$\prod_{i=1}^N p_{\mathbf{X}_i}(\mathbf{x}_i; M) = p_{\{\mathbf{X}\}}(\{\mathbf{x}\}; M) \quad (3.79)$$

$$\approx p_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\theta}}_{ML}; M) (2\pi)^{\frac{K_m}{2}} p_{\{\mathbf{X}\}|\boldsymbol{\Theta}}(\{\mathbf{x}\}|\hat{\boldsymbol{\theta}}_{ML}; M) |\mathbf{J}_{\{\mathbf{X}\}=\{\mathbf{x}\}}(\hat{\boldsymbol{\theta}}_{ML})|^{-\frac{1}{2}} \quad (3.80)$$

$$\approx p_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\theta}}_{ML}; M) (2\pi)^{\frac{K_m}{2}} p_{\{\mathbf{X}\}|\boldsymbol{\Theta}}(\{\mathbf{x}\}|\hat{\boldsymbol{\theta}}_{ML}; M) \left| \frac{1}{N} \mathbf{J}_{\bar{\mathbf{x}}_i}(\hat{\boldsymbol{\theta}}_{ML}) \right|^{-\frac{1}{2}} \quad (3.81)$$

$$= p_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\theta}}_{ML}; M) \left(\frac{2\pi}{N} \right)^{\frac{K_m}{2}} |\mathbf{J}_{\bar{\mathbf{x}}_i}(\hat{\boldsymbol{\theta}}_{ML})|^{-\frac{1}{2}} \prod_{i=1}^N p_{\mathbf{X}_i|\boldsymbol{\Theta}}(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_{ML}; M), \quad (3.82)$$

where we used the asymptotic property

$$\lim_{N \rightarrow \infty} \mathbf{J}_{\{\mathbf{x}\}=\{\mathbf{x}\}}(\hat{\boldsymbol{\theta}}_{ML}) = \frac{1}{N} \mathbf{J}_{\bar{\mathbf{x}}_i}(\hat{\boldsymbol{\theta}}_{ML}), \quad (3.83)$$

the latter being the *expected* Fisher information in any single observation, \mathbf{x}_i , about $\boldsymbol{\theta}$ (Wornell, 2010), in arriving at equation (3.81), and where K_m is the length of the parameter vector. Taking logarithms and normalizing by N we get

$$\begin{aligned} \frac{1}{N} L_{\mathbf{x}}^N(M) &= \frac{1}{N} L_{\mathbf{x}}^N(\hat{\boldsymbol{\theta}}_{ML}, M) + \frac{1}{N} \log p_{\boldsymbol{\Theta}}(\hat{\boldsymbol{\theta}}_{ML}; M) + \frac{K_m}{2N} \log 2\pi \\ &\quad - \frac{K_m}{2N} \log N - \frac{1}{N} \log |\mathbf{J}_{\bar{\mathbf{x}}_i}(\hat{\boldsymbol{\theta}}_{ML})|, \end{aligned} \quad (3.84)$$

where we have defined the notation:

$$L_{\mathbf{x}}^N(M) = \sum_{i=1}^N \log p_{\mathbf{x}}(\mathbf{x}_i; M) \quad (3.85)$$

$$L_{\mathbf{x}}^N(\hat{\boldsymbol{\theta}}_{ML}, M) = \sum_{i=1}^N \log p_{\mathbf{x}|\boldsymbol{\Theta}}(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_{ML}; M). \quad (3.86)$$

The above two equalities define the log-likelihood of the data integrated across all possible parameter vectors and the log-likelihood of the data evaluated at the ML estimate for the parameter vector, respectively, both for a mixture complexity of M .

Finally, for sufficiently large N , we keep only the order one terms of equation (3.84) to arrive at the Bayesian Information Criterion:

Definition: Bayesian Information Criterion

$$-2L_{\mathbf{x}}^N(M) = BIC \approx K_m \log N - 2L_{\mathbf{x}}^N(\hat{\boldsymbol{\theta}}_{ML}, M), \quad (3.87)$$

where N is the number of realizations; M is the mixture complexity; $L_{\mathbf{x}}^N(M)$ is the log-likelihood of the ensemble set integrated across all possible parameter values; $L_{\mathbf{x}}^N(\hat{\boldsymbol{\theta}}_{ML}, M)$ is the log-likelihood of the ensemble set evaluated at the ML estimate for the parameter vector; and K_m is the number of parameters.

Based on the set of ensemble realizations, $\{\mathbf{x}\} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$, we choose the

mixture complexity, M , for which the BIC is a minimum. We may conveniently consider the BIC to be the quantitative equivalent of the popular ‘Occam’s Razor’ (see e.g. MacKay (2003)), namely that one should favor the simplest hypothesis consistent with the ensemble set. Here, we wish to strike a balance between underfitting - and thus imposing too much structure onto the data - and overfitting, for which we limit our predictive capacity beyond the ensemble set. We do so by penalizing the fit of the realizations, quantified by twice the log-likelihood of the ensemble set evaluated at the ML parameter vector, $2L_{\mathbf{x}}^N(\hat{\boldsymbol{\theta}}_{ML}, M)$, with a term proportional to the mixture complexity, $K_m \log N$.

At this point, what remains for us is to introduce an efficient method for evolving the probabilistic description of the state in time. For this, we make use of the Dynamically Orthogonal field equations.

3.4 The Dynamically Orthogonal field equations

The Dynamically Orthogonal (DO) field equations, introduced by Sapsis and Lermusiaux (2009), are a closed set of evolution equations for general stochastic continuous fields, $X(\mathbf{r}, t; \omega)$, described by a stochastic partial differential equation (SPDE):

$$\frac{\partial X(\mathbf{r}, t; \omega)}{\partial t} = \mathcal{L}[X(\mathbf{r}, t; \omega); \omega], \quad (3.88)$$

where \mathbf{r} denotes the position in space; t is time; ω a random process source of stochasticity; and, $\mathcal{L}[\cdot]$ a general, potentially nonlinear, differential operator. By hypothesizing a generalized, time-dependent Karhunen-Loeve decomposition of the stochastic continuous field (Lermusiaux (2006), Sapsis and Lermusiaux (2009)),

$$X(\mathbf{r}, t; \omega) = \bar{x}(\mathbf{r}, t) + \sum_{i=1}^s \tilde{x}_i(\mathbf{r}, t) \Phi_i(t; \omega), \quad (3.89)$$

where $\bar{x}(\mathbf{r}, t)$ is the mean field; $\tilde{x}_i(\mathbf{r}, t)$ are orthonormal modes describing a basis for the stochastic subspace; and $\Phi_i(t; \omega)$ are zero-mean, stochastic processes, they

provide equations, based on the full SPDE, that govern the evolution of each of the aforementioned components.

Reduced-order statistical models, in which the full equations are projected onto a dominant, stochastic subspace, have previously been applied with great success, two such being the Proper Orthogonal Decomposition and Polynomial Chaos. Before describing the Dynamically Orthogonal field equations, we will give a brief introduction to these two approaches.

The following work is based on the MIT class notes on Numerical Methods for SPDEs (Marzouk and Wang, 2010), the Ph.D. thesis by Sapsis (2010) as well as the papers by Sapsis and Lermusiaux (2009, 2010).

3.4.1 Proper Orthogonal Decomposition

For Proper Orthogonal Decomposition (POD), the state of the dynamical system is assumed to take the form:

$$X(\mathbf{r}, t; \omega) = \sum_{i=1}^s x_i(\mathbf{r}) \Phi_i(t; \omega), \quad (3.90)$$

where the $\Phi_i(t; \omega)$ are stochastic processes and the $x_i(\mathbf{r})$ denote time-*independent*, precomputed fields based on *a priori* knowledge. Specifically, the set of $x_i(\mathbf{r})$ are orthonormal, providing an optimal modal decomposition in the sense that, when truncating the infinite expansion of modes, these capture the dominant physics.

A Galerkin projection of the original governing equation onto the low-dimensional subspace described by the basis functions, $x_i(\mathbf{r})$, is used to provide the reduced-order ordinary differential equations governing the evolution of the unknown stochastic coefficients, $\Phi_i(t; \omega)$.

3.4.2 Polynomial Chaos

For Polynomial Chaos (PC), instead of fixing the orthonormal fields, $x_i(\mathbf{r})$, as for POD, the stochastic processes, $X_i(t; \omega)$, are spectrally represented in terms of

fixed multi-dimensional hypergeometric polynomials based, for instance, on the Askey scheme,

$$X(\mathbf{r}, t; \omega) = \sum_{i=1}^s x_i(\mathbf{r}, t) \Phi_i(\eta(\omega)), \quad (3.91)$$

where the $\Phi_i(\omega)$ are orthogonal polynomials of random variables $\eta(\omega)$.

A Galerkin projection of the governing equation onto the low-dimensional subspace defined by the Φ_i transforms the original SPDE into a set of coupled deterministic PDEs for the unknown fields $x_i(\mathbf{r}, t)$. The *a priori* choice of random variable $\eta(\omega)$ dictates the convergence of the expansion. This choice is often highly problem dependent and, given their time-independence, their suitable choice can be a difficult task.

3.4.3 The Dynamically Orthogonal field equations

Both the Proper Orthogonal Decomposition and Polynomial Chaos suffer from fixing in time parts of their expansion, thus failing to adapt to changing dynamics. This deficiency was the main motivation for developing the Dynamically Orthogonal field equations.

As mentioned previously, for the D.O. equations the solution field is decomposed into a mean and stochastic dynamical component,

$$X(\mathbf{r}, t; \omega) = \bar{x}(\mathbf{r}, t) + \sum_{i=1}^s \tilde{x}_i(\mathbf{r}, t) \Phi_i(t; \omega), \quad (3.92)$$

where $\tilde{x}_i(\mathbf{r}, t)$ are the modes describing the stochastic subspace of size s , and $\Phi_i(t; \omega)$ are zero-mean, stochastic processes. By imposing nothing more than a condition on the time-evolution of the subspace, namely that it be orthogonal to itself,

$$\left\langle \frac{\partial \tilde{x}_i(\cdot, t)}{\partial t}, \tilde{x}_j(\cdot, t) \right\rangle = 0 \quad \forall i, j \in \{1, \dots, s\}, \quad (3.93)$$

the original SPDE is transformed into: (1) a partial differential equation (PDE) for the mean field; (2) a family of PDEs for the orthonormal bases describing the

stochastic subspace; and (3) a system of stochastic differential equations that define how the stochasticity evolves within the time-varying stochastic subspace. Using this approach, the stochastic subspace is dynamically evolved and therefore need not be chosen *a priori*; rather, it adapts to the stochasticity introduced by the stochastic initial and boundary conditions, and evolves according to the SPDE governing $X(\mathbf{r}, t; \omega)$. The stochastic coefficients, $\Phi_i(t; \omega)$, equally evolve according to dynamical equations derived directly from the original SPDE, allowing the use of the numerical scheme of choice for their solution (e.g. Monte Carlo methods). For details of an efficient numerical implementation of the D.O. equations, see Ueckermann et al. (2011). We also note that, while in the previous analysis the dimensionality of the stochastic subspace, s , has been assumed known and given, it can be evolved based on the dynamics and observations of the system (Sapsis and Lermusiaux, 2010), similarly to the ESSE scheme (Lermusiaux, 1999).

For a governing equation of the generic form:

$$\frac{\partial X(\mathbf{r}, t; \omega)}{\partial t} = \mathcal{L}[X(\mathbf{r}, t; \omega); \omega] \quad (3.94)$$

with initial conditions

$$X(\mathbf{r}, t_0; \omega) = X_0(\mathbf{r}; \omega) \quad (3.95)$$

and boundary conditions

$$\mathcal{B}[X(\mathbf{r}, t; \omega)]|_{\mathbf{r}=\boldsymbol{\xi}} = h(\boldsymbol{\xi}, t; \omega), \quad (3.96)$$

where \mathcal{B} is a linear differential operator and $\boldsymbol{\xi}$ the spatial coordinate denoting the boundary, we introduce the generalized, time-dependent Karhunen-Loeve decomposition:

$$X(\mathbf{r}, t; \omega) = \bar{x}(\mathbf{r}, t) + \sum_{i=1}^s \tilde{x}_i(\mathbf{r}, t) \Phi_i(t; \omega), \quad (3.97)$$

where $\bar{x}(\mathbf{r}, t)$ is the mean field; $\tilde{x}_i(\mathbf{r}, t)$ are orthonormal modes describing a basis for the stochastic subspace; and $\Phi_i(t; \omega)$ are zero-mean, stochastic processes. The D.O.

evolution equations are then defined as follows (where Einstein summation is adopted, i.e. $\sum_i a_i b_i \equiv a_i b_i$):

$$\frac{d\Phi_i(t; \omega)}{dt} = \langle \mathcal{L}[X(\cdot, t; \omega); \omega] - \mathcal{E}[\mathcal{L}[X(\cdot, t; \omega); \omega]], \tilde{x}_i(\cdot, t) \rangle, \quad (3.98)$$

$$\frac{\partial \bar{x}(\mathbf{r}, t)}{\partial t} = \mathcal{E}[\mathcal{L}[X(\mathbf{r}, t; \omega); \omega]], \quad (3.99)$$

$$\frac{\partial \tilde{x}_i(\mathbf{r}, t)}{\partial t} = \Pi_{\mathcal{X}}^{\perp}(\mathcal{E}[\mathcal{L}[X(\mathbf{r}, t; \omega); \omega] \phi_j(t; \omega)]) C_{\Phi_i(t) \Phi_j(t)}^{-1}, \quad (3.100)$$

where

$$\Pi_{\mathcal{X}}^{\perp}(F(\mathbf{r})) \equiv F(\mathbf{r}) - \langle F(\cdot), \tilde{x}_k(\cdot, t) \rangle \tilde{x}_k(\mathbf{r}, t) \quad (3.101)$$

is the projection of $F(\mathbf{r})$ onto the null space of the stochastic subspace and

$$C_{\Phi_i(t) \Phi_j(t)} \equiv \mathcal{E}[\Phi_i(t; \omega) \Phi_j(t; \omega)] \quad (3.102)$$

is the correlation between random variables $\Phi_i(t; \omega)$ and $\Phi_j(t; \omega)$. The associated boundary conditions take the form

$$\mathcal{B}[\bar{x}(\mathbf{r}, t)]|_{\mathbf{r}=\boldsymbol{\xi}} = \mathcal{E}[h(\boldsymbol{\xi}, t; \omega)] \quad (3.103)$$

$$\mathcal{B}[\tilde{x}_i(\mathbf{r}, t)]|_{\mathbf{r}=\boldsymbol{\xi}} = \mathcal{E}[h(\boldsymbol{\xi}, t; \omega) \Phi_j(t; \omega)] C_{\Phi_i(t) \Phi_j(t)}^{-1} \quad (3.104)$$

and the initial conditions are given by

$$\Phi_i(t_0; \omega) = \langle X_0(\cdot; \omega) - \bar{x}_0(\cdot), \tilde{x}_{i0}(\cdot) \rangle \quad (3.105)$$

$$\bar{x}(\mathbf{r}, t_0) = \bar{x}_0(\mathbf{r}) = \mathcal{E}[X_0(\mathbf{r}; \omega)] \quad (3.106)$$

$$\tilde{x}_i(\mathbf{r}, t_0) = \tilde{x}_{i0}(\mathbf{r}) \quad (3.107)$$

for all $i = 1, \dots, s$, where $\tilde{x}_{i0}(\mathbf{r})$ are the orthonormal modes describing a basis for the stochastic subspace at time zero. This completes the definition of the Dynamically Orthogonal field equations.

On an aside, if suitable assumptions are made, either on the form of the fields

$\bar{x}(\mathbf{r}, t)$ and $\tilde{x}_i(\mathbf{r}, t)$, or on that of $\Phi_i(t; \omega)$, the Dynamically Orthogonal field equations may be shown to reproduce the reduced-order equations obtained by application of the Proper Orthogonal Decomposition or Polynomial Chaos, respectively (Sapsis and Lermusiaux, 2009).

3.5 The GMM-DO filter

Combining the concepts introduced in the previous section, and building on the foundations of classical assimilation schemes, we introduce the GMM-DO filter: data assimilation with Gaussian mixture models using the Dynamically Orthogonal field equations. We view the GMM-DO filter as an efficient, data-driven assimilation scheme that preserves non-Gaussian statistics and respects nonlinear dynamics. With the GMM-DO filter we solely focus on the task of filtering; the derivation of a smoothing algorithm is to be addressed in a future work by the MSEAS group.

Consistent with Bayes' filter, our scheme is composed of two distinct components: a forecast step and an update step. In what follows, we proceed to describe each of these in detail. We refer the reader to table 3.1 for clarification of notation specific to the GMM-DO filter.

3.5.1 Initial Conditions

We initialize the state vector at discrete time $k = 0$ in a decomposed form that accords with the Dynamically Orthogonal field equations:

$$\mathbf{X}_0 = \bar{\mathbf{x}}_0 + \sum_{i=1}^{s_0} \tilde{\mathbf{x}}_{i,0} \Phi_{i,0}(\omega). \quad (3.108)$$

We choose the state mean, $\bar{\mathbf{x}}$, the orthonormal modes, $\tilde{\mathbf{x}}_i$, and the stochastic coefficients, $\Phi_i(\omega)$, such as to best represent our current knowledge of the state. While various possible representations for the stochastic coefficients, $\Phi_i(\omega)$, exist, we adopt a Monte Carlo approach in accordance with the scheme presented by Ueckermann et al. (2011). Specifically, we draw N realizations from the multivariate random

Table 3.1: Notation relevant to the GMM-DO filter.

<i>Descriptors</i>		
$(\cdot)^f$		forecast
$(\cdot)^a$		analysis
<i>Scalars</i>		
i	$\in \mathbb{N}$	stochastic subspace index
j	$\in \mathbb{N}$	mixture index
k	$\in \mathbb{N}$	discrete time index
n	$\in \mathbb{N}$	dimension of state vector
p	$\in \mathbb{N}$	dimension of observation vector
r	$\in \mathbb{N}$	realization index
s	$\in \mathbb{N}$	dimension of stochastic subspace
M	$\in \mathbb{N}$	complexity of Gaussian Mixture Model
N	$\in \mathbb{N}$	number of Monte Carlo members
Φ_i	$\in \mathbb{R}$	random variable describing probability density function for orthonormal mode $\tilde{\mathbf{x}}_i$
<i>Vectors</i>		
\mathbf{X}	$\in \mathbb{R}^n$	state (random) vector
$\tilde{\mathbf{x}}_i$	$\in \mathbb{R}^n$	modes describing an orthonormal basis for the stochastic subspace
$\bar{\mathbf{x}}$	$\in \mathbb{R}^n$	mean state vector
\mathbf{x}_r	$\in \mathbb{R}^n$	state realization
\mathbf{Y}	$\in \mathbb{R}^m$	observation (random) vector
\mathbf{y}	$\in \mathbb{R}^m$	observation realization
$\bar{\mathbf{x}}_j$	$\in \mathbb{R}^n$	mean vector of mixture j in state space
$\boldsymbol{\mu}_j$	$\in \mathbb{R}^s$	mean vector of mixture j in stochastic subspace
$\boldsymbol{\phi}_r$	$\in \mathbb{R}^s$	realization residing in stochastic subspace
$\boldsymbol{\Upsilon}$	$\in \mathbb{R}^m$	observation noise (random) vector
<i>Matrices</i>		
\mathbf{P}	$\in \mathbb{R}^{n \times n}$	covariance matrix in state space
$\boldsymbol{\Sigma}_j$	$\in \mathbb{R}^{s \times s}$	covariance matrix of mixture j in stochastic subspace
\mathbf{P}_j	$\in \mathbb{R}^{n \times n}$	covariance matrix of mixture j in state space
\mathbf{R}	$\in \mathbb{R}^{m \times m}$	observation covariance matrix
\mathbf{H}	$\in \mathbb{R}^{m \times n}$	(linear) observation model
$\boldsymbol{\mathcal{X}}$	$\in \mathbb{R}^{n \times s}$	matrix of orthonormal modes, $[\tilde{\mathbf{x}}_1 \tilde{\mathbf{x}}_2 \dots \tilde{\mathbf{x}}_s]$
$\{\boldsymbol{\phi}\}$	$\in \mathbb{R}^{s \times N}$	set of subspace ensemble realizations, $\{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_N\}$

vector, $\{\Phi_1(\omega), \Phi_2(\omega), \dots, \Phi_s(\omega)\}$, to arrive at its Monte Carlo representation,

$$\{\boldsymbol{\phi}\} = \{\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_N\}. \quad (3.109)$$

We emphasize that the $\boldsymbol{\phi}_r \in \mathbb{R}^s$ represent realizations residing in the stochastic subspace of dimension s . With this, we rewrite equation (3.108) in its Monte Carlo form,

$$\mathbf{x}_{r,0} = \bar{\mathbf{x}}_0 + \boldsymbol{\mathcal{X}}_0 \boldsymbol{\phi}_{r,0}, \quad r = \{1, \dots, N\}, \quad (3.110)$$

where, as noted in table 3.1, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n \times s}$ defines the matrix of modes describing an orthonormal basis for the stochastic subspace.

3.5.2 Forecast

Using either the initial D.O. conditions or the posterior state description following the assimilation of data at time $k - 1$,

$$\mathbf{x}_{r,k-1}^a = \bar{\mathbf{x}}_{k-1}^a + \boldsymbol{\mathcal{X}}_{k-1} \boldsymbol{\phi}_{r,k-1}^a, \quad r = \{1, \dots, N\}, \quad (3.111)$$

we use the D.O. equations, (3.98) - (3.100), to efficiently evolve the probabilistic description of the state vector in time, arriving at a forecast for observation time k :

$$\mathbf{x}_{r,k}^f = \bar{\mathbf{x}}_k^f + \boldsymbol{\mathcal{X}}_k \boldsymbol{\phi}_{r,k}^f, \quad r = \{1, \dots, N\}. \quad (3.112)$$

We again refer the reader to the paper by Ueckermann et al. (2011) for an efficient implementation of the forecast step.

We note that we neglect the notation of $(\cdot)^f$ and $(\cdot)^a$ on the stochastic subspace, $\boldsymbol{\mathcal{X}}$, as this is independent of the assimilated observations. In some error subspace schemes (Lermusiaux, 1999), observation updates and posterior misfits are used to learn and update the subspace following the assimilation step, resulting in prior and posterior subspaces that may not be identical.

3.5.3 Observation

Common to ocean and atmospheric applications, we impose an observation model in accordance with the classical representation,

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{\Upsilon}, \quad \mathbf{\Upsilon} \sim \mathcal{N}(\mathbf{v}; \mathbf{0}, \mathbf{R}). \quad (3.113)$$

We denote the realized observation by $\mathbf{y} \in \mathbb{R}^p$.

3.5.4 Update

In what follows, to avoid clutter of the analysis, we omit the notation of time with the understanding that the update occurs at discrete time k .

Based on our state forecast,

$$\mathbf{x}_r^f = \bar{\mathbf{x}}^f + \mathbf{X}\boldsymbol{\phi}_r^f, \quad r = \{1, \dots, N\},$$

our goal is to update the mean vector, $\bar{\mathbf{x}}^f$, as well as the set of ensemble realizations, $\{\boldsymbol{\phi}^f\} = \{\boldsymbol{\phi}_1^f, \dots, \boldsymbol{\phi}_N^f\}$, in accordance with measurement \mathbf{y} . With this, we arrive at the posterior state estimate:

$$\mathbf{x}_r^a = \bar{\mathbf{x}}^a + \mathbf{X}\boldsymbol{\phi}_r^a, \quad r = \{1, \dots, N\}.$$

We do so by optimally fitting a Gaussian mixture model to the set of ensemble realizations from which we may proceed with updating our state estimate in accordance with Bayes' Law. Under the assumption that the Gaussian mixture model provides an accurate representation of the true probability density function we thus arrive at an equally accurate description of the posterior state of the system following the assimilation of the measurement.

In what follows, we describe the update algorithm in detail.

(i) GMM representation of prior set of ensemble realizations

At the time of a new set of measurements, \mathbf{y} , we use the EM algorithm to determine the Gaussian mixture model that best represents the set of ensemble realizations *within the stochastic subspace*, $\{\boldsymbol{\phi}^f\} = \{\boldsymbol{\phi}_1^f, \dots, \boldsymbol{\phi}_N^f\}$. We denote the parameters of the Gaussian mixture model by

$$\pi_j^f, \boldsymbol{\mu}_j^f, \boldsymbol{\Sigma}_j^f, \quad j = 1, \dots, M,$$

where $\pi_j^f \in [0, 1]$, $\boldsymbol{\mu}_j^f \in \mathbb{R}^s$ and $\boldsymbol{\Sigma}_j^f \in \mathbb{R}^{s \times s}$. We again stress that the Gaussian mixture model efficiently resides in an s -dimensional subspace of the n -dimensional state space, with $s \ll n$, thus making the prior estimation procedure computationally feasible.

We determine the optimal mixture complexity by application of the Bayesian Information Criterion, (3.87), successively fitting Gaussian mixture models of increasing complexity (i.e. $M = 1, 2, 3, \dots$) until a minimum of the BIC is met. The final result is a Gaussian mixture model optimally fit to the ensemble realizations *in the stochastic subspace* whose probability density function we write as

$$p_{\Phi^f}(\boldsymbol{\phi}^f) = \sum_{j=1}^M \pi_j^f \times \mathcal{N}(\boldsymbol{\phi}^f; \boldsymbol{\mu}_j^f, \boldsymbol{\Sigma}_j^f). \quad (3.114)$$

Due to the affine transformation linking the stochastic subspace with the state space we may expand the previously determined Gaussian mixture model into the state space according to the equations:

$$\bar{\mathbf{x}}_j^f = \bar{\mathbf{x}}^f + \boldsymbol{\chi} \boldsymbol{\mu}_j^f \quad (3.115)$$

$$\mathbf{P}_j^f = \boldsymbol{\chi} \boldsymbol{\Sigma}_j^f \boldsymbol{\chi}^T. \quad (3.116)$$

The mixture weights, π_j^f , naturally remain unchanged. We note that $\bar{\mathbf{x}}_j^f$ and \mathbf{P}_j^f now refer to the mean vector and covariance matrix, respectively, for mixture j in the state space. We thus arrive at the prior distribution for the state vector in state

space, taking the form of the following Gaussian mixture model:

$$p_{\mathbf{X}^f}(\mathbf{x}^f) = \sum_{j=1}^M \pi_j^f \times \mathcal{N}(\mathbf{x}^f; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f). \quad (3.117)$$

We emphasize that, due to the affine transformation linking the stochastic subspace with the state space, this distribution would equally have been obtained had we performed the prior fitting of the Gaussian mixture model directly in the state space based on the set of realizations $\{\mathbf{x}^f\} = \{\mathbf{x}_1^f, \dots, \mathbf{x}_N^f\}$.

(ii) Bayesian update

Using update equations (3.4) - (3.7) based on measurement \mathbf{y} , but extending it to the case of a linear observation operator \mathbf{H} , equation (3.113), we obtain for the posterior distribution for the state vector in state space (see also Alspach and Sorenson (1972))

$$p_{\mathbf{X}^a}(\mathbf{x}^a) = \sum_{j=1}^M \pi_j^a \times \mathcal{N}(\mathbf{x}^a; \bar{\mathbf{x}}_j^a, \mathbf{P}_j^a) \quad (3.118)$$

with

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_j^f, \mathbf{H}\mathbf{P}_j^f\mathbf{H}^T + \mathbf{R})}{\sum_{m=1}^M \pi_m^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_m^f, \mathbf{H}\mathbf{P}_m^f\mathbf{H}^T + \mathbf{R})} \quad (3.119)$$

$$\bar{\mathbf{x}}_j^a = \bar{\mathbf{x}}_j^f + \mathbf{K}_j(\mathbf{y} - \mathbf{H}\bar{\mathbf{x}}_j^f) \quad (3.120)$$

$$\mathbf{P}_j^a = (\mathbf{I} - \mathbf{K}_j\mathbf{H})\mathbf{P}_j^f \quad (3.121)$$

where

$$\mathbf{K}_j = \mathbf{P}_j^f\mathbf{H}^T(\mathbf{H}\mathbf{P}_j^f\mathbf{H}^T + \mathbf{R})^{-1} \quad (3.122)$$

is the Kalman gain matrix associated with mixture j .

With this, we may obtain the expression for the posterior mean field in the state

space:

$$\bar{\mathbf{x}}^a = \sum_{j=1}^M \pi_j^a \times \bar{\mathbf{x}}_j^a. \quad (3.123)$$

Although unnecessary, we may equally determine the posterior (full) covariance matrix in the state space using the Law of Total Variance (see the appendix):

$$\mathbf{P}^a = \sum_{j=1}^M \pi_j^a \mathbf{P}_j^a + \sum_{j=1}^M \pi_j^a (\bar{\mathbf{x}}_j^a - \bar{\mathbf{x}}^a) (\bar{\mathbf{x}}_j^a - \bar{\mathbf{x}}^a)^T. \quad (3.124)$$

(iii) GMM representation of posterior set of ensemble realizations

Ultimately, we wish to project the updated GMM parameters, $\bar{\mathbf{x}}_j^a$ and \mathbf{P}_j^a , back into the stochastic subspace, obtaining values for $\boldsymbol{\mu}_j^a$ and $\boldsymbol{\Sigma}_j^a$. In doing so, we again make use of the affine transformation linking the stochastic subspace with the state space. We re-emphasize that the stochastic subspace itself has remained unchanged during assimilation of the observations and is thus still described by matrix $\boldsymbol{\mathcal{X}}$.

To determine the updated mixture means, $\boldsymbol{\mu}_j^a$, similar to (3.115) we first write:

$$\bar{\mathbf{x}}_j^a = \bar{\mathbf{x}}^a + \boldsymbol{\mathcal{X}} \boldsymbol{\mu}_j^a. \quad (3.125)$$

By subtraction of $\bar{\mathbf{x}}^a$ and left multiplication by $\boldsymbol{\mathcal{X}}^T$, we then obtain:

$$\boldsymbol{\mu}_j^a = (\boldsymbol{\mathcal{X}}^T \boldsymbol{\mathcal{X}})^{-1} \boldsymbol{\mathcal{X}}^T (\bar{\mathbf{x}}_j^a - \bar{\mathbf{x}}^a) \quad (3.126)$$

$$= \boldsymbol{\mathcal{X}}^T (\bar{\mathbf{x}}_j^a - \bar{\mathbf{x}}^a), \quad (3.127)$$

where (3.127) results from the orthonormality of the modes, i.e. $\boldsymbol{\mathcal{X}}^T \boldsymbol{\mathcal{X}} = \mathbf{I}$.

To determine the updated mixture covariance matrices, $\boldsymbol{\Sigma}_j^a$, we proceed in a similar manner. Using the decomposed structure of equation (3.116), as well as equation (3.121), repeated here for convenience,

$$\begin{aligned} \mathbf{P}_j^a &= \boldsymbol{\mathcal{X}} \boldsymbol{\Sigma}_j^a \boldsymbol{\mathcal{X}}^T \\ &= (\mathbf{I} - \mathbf{K}_j \mathbf{H}) \mathbf{P}_j^f, \end{aligned}$$

we left multiply by \mathcal{X}^T and right multiply by \mathcal{X} to obtain:

$$\Sigma_j^a = (\mathcal{X}^T \mathcal{X})^{-1} \mathcal{X}^T (I - K_j H) (\mathcal{X}^T \mathcal{X})^{-1} \quad (3.128)$$

$$= \mathcal{X}^T (I - K_j H) P_j^f \mathcal{X}, \quad (3.129)$$

where, as before, equation (3.129) follows from the orthonormality of the modes.

At this point, we have arrived at expressions for the posterior mean vector, $\bar{\mathbf{x}}^a$, as well as the posterior GMM parameters in the stochastic subspace, π_j^a , $\boldsymbol{\mu}_j^a$ and Σ_j^a , repeated here for clarity:

$$\bar{\mathbf{x}}^a = \sum_{j=1}^M \pi_j^a \times \bar{\mathbf{x}}_j^a \quad (3.130)$$

$$= \sum_{j=1}^M \pi_j^a \times (\bar{\mathbf{x}}_j^f + K_j (\mathbf{y} - H \bar{\mathbf{x}}_j^f)) \quad (3.131)$$

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; H \bar{\mathbf{x}}_j^f, H P_j^f H^T + \mathbf{R})}{\sum_{m=1}^M \pi_m^f \times \mathcal{N}(\mathbf{y}; H \bar{\mathbf{x}}_m^f, H P_m^f H^T + \mathbf{R})} \quad (3.132)$$

$$\boldsymbol{\mu}_j^a = \mathcal{X}^T (\bar{\mathbf{x}}_j^a - \bar{\mathbf{x}}^a) \quad (3.133)$$

$$\Sigma_j^a = \mathcal{X}^T (I - K_j H) P_j^f \mathcal{X}. \quad (3.134)$$

We use the latter three to generate a posterior set of ensemble realizations within the stochastic subspace, $\{\boldsymbol{\phi}^a\} = \{\boldsymbol{\phi}_1^a, \dots, \boldsymbol{\phi}_N^a\}$, thus arriving at our Monte Carlo form for the posterior state description at discrete time k ,

$$\mathbf{x}_{r,k}^a = \bar{\mathbf{x}}_k^a + \mathcal{X}_k \boldsymbol{\phi}_{r,k}^a, \quad r = \{1, \dots, N\}. \quad (3.135)$$

Before proceeding to do so, however, we remark on an efficient implementation of the previously described algorithm, significantly lessening the computational burden.

Remark:

The previous update equations were deliberately performed in the state space to provide an ease of understanding. Considering that uncertainty of the state is restricted to the stochastic subspace, however, we may conveniently perform the Bayesian update

therein. (Anything in the null space of the stochastic subspace remains deterministic and unknown). This, of course, provides significant computational savings due to its reduced dimensionality. In what follows, it will be convenient to define the notation:

$$\tilde{\mathbf{H}} \equiv \mathbf{H}\mathbf{X} \quad (3.136)$$

$$\tilde{\mathbf{y}} \equiv \mathbf{y} - \mathbf{H}\bar{\mathbf{x}}^f \quad (3.137)$$

$$\tilde{\mathbf{K}}_j \equiv \Sigma_j^f \tilde{\mathbf{H}}^T (\tilde{\mathbf{H}} \Sigma_j^f \tilde{\mathbf{H}}^T + \mathbf{R})^{-1} \quad (3.138)$$

$$\begin{aligned} &= \Sigma_j^f \mathbf{X}^T \mathbf{H}^T (\mathbf{H} \mathbf{X} \Sigma_j^f \mathbf{X}^T \mathbf{H}^T + \mathbf{R})^{-1} \\ &= \mathbf{X}^T \mathbf{P}_j^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_j^f \mathbf{H}^T + \mathbf{R})^{-1} \\ &= \mathbf{X}^T \mathbf{K}_j, \end{aligned} \quad (3.139)$$

where, in arriving at equation (3.139), we made use of the identity $\mathbf{P}^f = \mathbf{X} \Sigma_j^f \mathbf{X}^T$, the orthonormality of the modes as well as definition (3.136).

We will show – through simple manipulation of terms – that the update equations for parameters $\bar{\mathbf{x}}^a$, π_j^a , $\boldsymbol{\mu}_j^a$ and Σ_j^a , previously performed in the state space, may equivalently be expressed in notation specific to the stochastic subspace. What results is an efficient implementation of the prior results.

Starting with the update equation for the mixture weights, equation (3.132), we have:

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_j^f, \mathbf{H}\mathbf{P}_j^f \mathbf{H}^T + \mathbf{R})}{\sum_{l=1}^M \pi_l^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_l^f, \mathbf{H}\mathbf{P}_l^f \mathbf{H}^T + \mathbf{R})} \quad (3.140)$$

$$= \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}(\bar{\mathbf{x}}^f + \mathbf{X}\boldsymbol{\mu}_j^f), \mathbf{H}\mathbf{X}\Sigma_j^f \mathbf{X}^T \mathbf{H}^T + \mathbf{R})}{\sum_{l=1}^M \pi_l^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}(\bar{\mathbf{x}}^f + \mathbf{X}\boldsymbol{\mu}_l^f), \mathbf{H}\mathbf{X}\Sigma_l^f \mathbf{X}^T \mathbf{H}^T + \mathbf{R})}, \quad (3.141)$$

by using equations (3.115) and (3.116),

$$= \frac{\pi_j^f \times \mathcal{N}(\mathbf{y} - \mathbf{H}\bar{\mathbf{x}}^f; \mathbf{H}\mathbf{X}\boldsymbol{\mu}_j^f, \mathbf{H}\mathbf{X}\Sigma_j^f \mathbf{X}^T \mathbf{H}^T + \mathbf{R})}{\sum_{l=1}^M \pi_l^f \times \mathcal{N}(\mathbf{y} - \mathbf{H}\bar{\mathbf{x}}^f; \mathbf{H}\mathbf{X}\boldsymbol{\mu}_l^f, \mathbf{H}\mathbf{X}\Sigma_l^f \mathbf{X}^T \mathbf{H}^T + \mathbf{R})}, \quad (3.142)$$

by simple rearranging of terms,

$$= \frac{\pi_j^f \times \mathcal{N}(\tilde{\mathbf{y}}; \tilde{\mathbf{H}}\boldsymbol{\mu}_j^f, \tilde{\mathbf{H}}\boldsymbol{\Sigma}_j^f\tilde{\mathbf{H}}^T + \mathbf{R})}{\sum_{l=1}^M \pi_l^f \times \mathcal{N}(\tilde{\mathbf{y}}; \tilde{\mathbf{H}}\boldsymbol{\mu}_l^f, \tilde{\mathbf{H}}\boldsymbol{\Sigma}_l^f\tilde{\mathbf{H}}^T + \mathbf{R})}, \quad (3.143)$$

by application of definitions (3.136) and (3.137). With this, we've expressed the update equation for the mixture weights in notation specific to the stochastic subspace, all the while retaining the familiar structure of equation (3.132).

In a similar manner, starting with the equation for the updated mean vector, equation (3.131), we have:

$$\bar{\mathbf{x}}^a = \sum_{j=1}^M \pi_j^a \times (\bar{\mathbf{x}}_j^f + \mathbf{K}_j(\mathbf{y} - \mathbf{H}\bar{\mathbf{x}}_j^f)) \quad (3.144)$$

$$= \sum_{j=1}^M \pi_j^a \times (\bar{\mathbf{x}}^f + \boldsymbol{\chi}\boldsymbol{\mu}_j^f + \boldsymbol{\chi}\tilde{\mathbf{K}}_j(\mathbf{y} - \mathbf{H}(\bar{\mathbf{x}}^f + \boldsymbol{\chi}\boldsymbol{\mu}_j^f))) \quad (3.145)$$

by using equation (3.115) and applying definition (3.139),

$$= \bar{\mathbf{x}}^f + \boldsymbol{\chi} \sum_{j=1}^M \pi_j^a \times (\boldsymbol{\mu}_j^f + \tilde{\mathbf{K}}_j(\tilde{\mathbf{y}} - \tilde{\mathbf{H}}\boldsymbol{\mu}_j^f)) \quad (3.146)$$

by using $\sum_{j=1}^M \pi_j^a \times \bar{\mathbf{x}}^f = \bar{\mathbf{x}}^f$ and applying definitions (3.136) and (3.137),

$$= \bar{\mathbf{x}}^f + \boldsymbol{\chi} \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a, \quad (3.147)$$

where we have defined the 'intermediate' mean vector in the stochastic subspace, $\hat{\boldsymbol{\mu}}_j^a = \boldsymbol{\mu}_j^f + \tilde{\mathbf{K}}_j(\tilde{\mathbf{y}} - \tilde{\mathbf{H}}\boldsymbol{\mu}_j^f)$. Its intermediate nature results from the fact that we require the parametric distribution describing the stochastic subspace to be of mean zero, i.e. $\sum_{j=1}^M \pi_j^a \times \boldsymbol{\mu}_j^a = 0$. Presumably, we fail to satisfy this condition by adopting the intermediate mean vectors, $\hat{\boldsymbol{\mu}}_j^a$. This is clearly and simply circumvented by imposing

the map:

$$\hat{\boldsymbol{\mu}}_j^a \mapsto \hat{\boldsymbol{\mu}}_j^a - \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a. \quad (3.148)$$

Rather than merely stating this as a matter of fact, however, we may equally arrive at this result by manipulating the appropriate equation in the state space. Specifically, starting with equation (3.133), we have:

$$\boldsymbol{\mu}_j^a = \boldsymbol{\mathcal{X}}^T (\bar{\mathbf{x}}_j^a - \bar{\mathbf{x}}^a) \quad (3.149)$$

$$= \boldsymbol{\mathcal{X}}^T (\bar{\mathbf{x}}_j^f + \mathbf{K}_j(\mathbf{y} - \mathbf{H}\bar{\mathbf{x}}_j^f) - \bar{\mathbf{x}}^f - \boldsymbol{\mathcal{X}} \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a), \quad (3.150)$$

by using equations (3.120) and (3.147),

$$= \boldsymbol{\mathcal{X}}^T (\bar{\mathbf{x}}^f + \boldsymbol{\mathcal{X}}\boldsymbol{\mu}_j^f + \boldsymbol{\mathcal{X}}\tilde{\mathbf{K}}_j(\mathbf{y} - \mathbf{H}\bar{\mathbf{x}}_j^f) - \bar{\mathbf{x}}^f - \boldsymbol{\mathcal{X}} \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a), \quad (3.151)$$

by using equation (3.115) and definition (3.139),

$$= \boldsymbol{\mu}_j^f + \tilde{\mathbf{K}}_j(\tilde{\mathbf{y}} - \tilde{\mathbf{H}}\bar{\mathbf{x}}_j^f) - \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a, \quad (3.152)$$

by orthonormality of the modes,

$$= \hat{\boldsymbol{\mu}}_j^a - \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a, \quad (3.153)$$

as required.

Finally, starting with the update equation for the mixture covariance matrices, equation (3.134), we have:

$$\boldsymbol{\Sigma}_j^a = \boldsymbol{\mathcal{X}}^T (\mathbf{I} - \mathbf{K}_j \mathbf{H}) \mathbf{P}_j^f \boldsymbol{\mathcal{X}} \quad (3.154)$$

$$= \boldsymbol{\mathcal{X}}^T (\mathbf{I} - \boldsymbol{\mathcal{X}}\tilde{\mathbf{K}}_j \mathbf{H}) \boldsymbol{\mathcal{X}} \boldsymbol{\Sigma}_j^f \boldsymbol{\mathcal{X}}^T \boldsymbol{\mathcal{X}}, \quad (3.155)$$

by using equation (3.116),

$$= (\mathbf{I} - \tilde{\mathbf{K}}_j \tilde{\mathbf{H}}) \Sigma_j^f, \quad (3.156)$$

by orthonormality of the modes and using definitions (3.136) and (3.139).

With this analysis, we have efficiently arrived at the updated parameters in a framework associated with the stochastic subspace:

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^f + \boldsymbol{\chi} \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a \quad (3.157)$$

$$= \bar{\mathbf{x}}^f + \boldsymbol{\chi} \sum_{j=1}^M \pi_j^a \times (\boldsymbol{\mu}_j^f + \tilde{\mathbf{K}}_j (\tilde{\mathbf{y}} - \tilde{\mathbf{H}} \boldsymbol{\mu}_j^f)) \quad (3.158)$$

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\tilde{\mathbf{y}}; \tilde{\mathbf{H}} \boldsymbol{\mu}_j^f, \tilde{\mathbf{H}} \Sigma_j^f \tilde{\mathbf{H}}^T + \mathbf{R})}{\sum_{l=1}^M \pi_l^f \times \mathcal{N}(\tilde{\mathbf{y}}; \tilde{\mathbf{H}} \boldsymbol{\mu}_l^f, \tilde{\mathbf{H}} \Sigma_l^f \tilde{\mathbf{H}}^T + \mathbf{R})} \quad (3.159)$$

$$\boldsymbol{\mu}_j^a = \hat{\boldsymbol{\mu}}_j^a - \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a \quad (3.160)$$

$$\Sigma_j^a = (\mathbf{I} - \tilde{\mathbf{K}}_j \tilde{\mathbf{H}}) \Sigma_j^f. \quad (3.161)$$

We have proved their equivalence with the update equations in the state space by direct manipulation of terms.

(iv) Generation of posterior ensemble realizations

We complete the update step, as with ESSE scheme A (Lermusiaux, 1997), by generating a new set of realizations *within the stochastic subspace*, $\{\boldsymbol{\phi}^a\} = \{\boldsymbol{\phi}_1^a, \dots, \boldsymbol{\phi}_N^a\}$, according to the multivariate Gaussian mixture model with parameters

$$\pi_j^a, \boldsymbol{\mu}_j^a, \Sigma_j^a, \quad j = 1, \dots, M.$$

With this, we have arrived at an updated D.O. representation for the state vector

based on the assimilation of observations at time k ,

$$\mathbf{x}_{r,k}^a = \bar{\mathbf{x}}_k^a + \mathcal{X}_k \phi_{r,k}^a, \quad r = \{1, \dots, N\}. \quad (3.162)$$

From here, we proceed with the next forecast from time step k to $k+1$. This concludes the GMM-DO filter. We summarize the procedure using the flowchart displayed in figure 3-2.

In what follows, we illustrate the update procedure of the GMM-DO filter by way of a simple example.

3.5.5 Example

Assume we are provided with the following (arbitrarily chosen) forecast for the D.O. decomposed representation of the state:

$$\bar{\mathbf{x}}^f = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad \text{and} \quad \mathcal{X} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix},$$

with one hundred subspace realizations, $\{\phi^f\} = \{\phi_1^f, \dots, \phi_{100}^f\}$, generated from a Gaussian mixture model of complexity two:

$$p_{\Phi}(\phi) = \sum_{j=1}^2 \pi_j \times \mathcal{N}(\phi; \boldsymbol{\mu}_j^f, \boldsymbol{\Sigma}_j^f).$$

Let us further arbitrarily choose the following forecast parameters:

$$\begin{aligned} \pi_1^f &= 0.5, & \boldsymbol{\mu}_1^f &= \begin{bmatrix} -10 \\ -1 \end{bmatrix}, & \boldsymbol{\Sigma}_1^f &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ \pi_2^f &= 0.5, & \boldsymbol{\mu}_2^f &= \begin{bmatrix} 10 \\ 1 \end{bmatrix}, & \boldsymbol{\Sigma}_2^f &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \end{aligned}$$

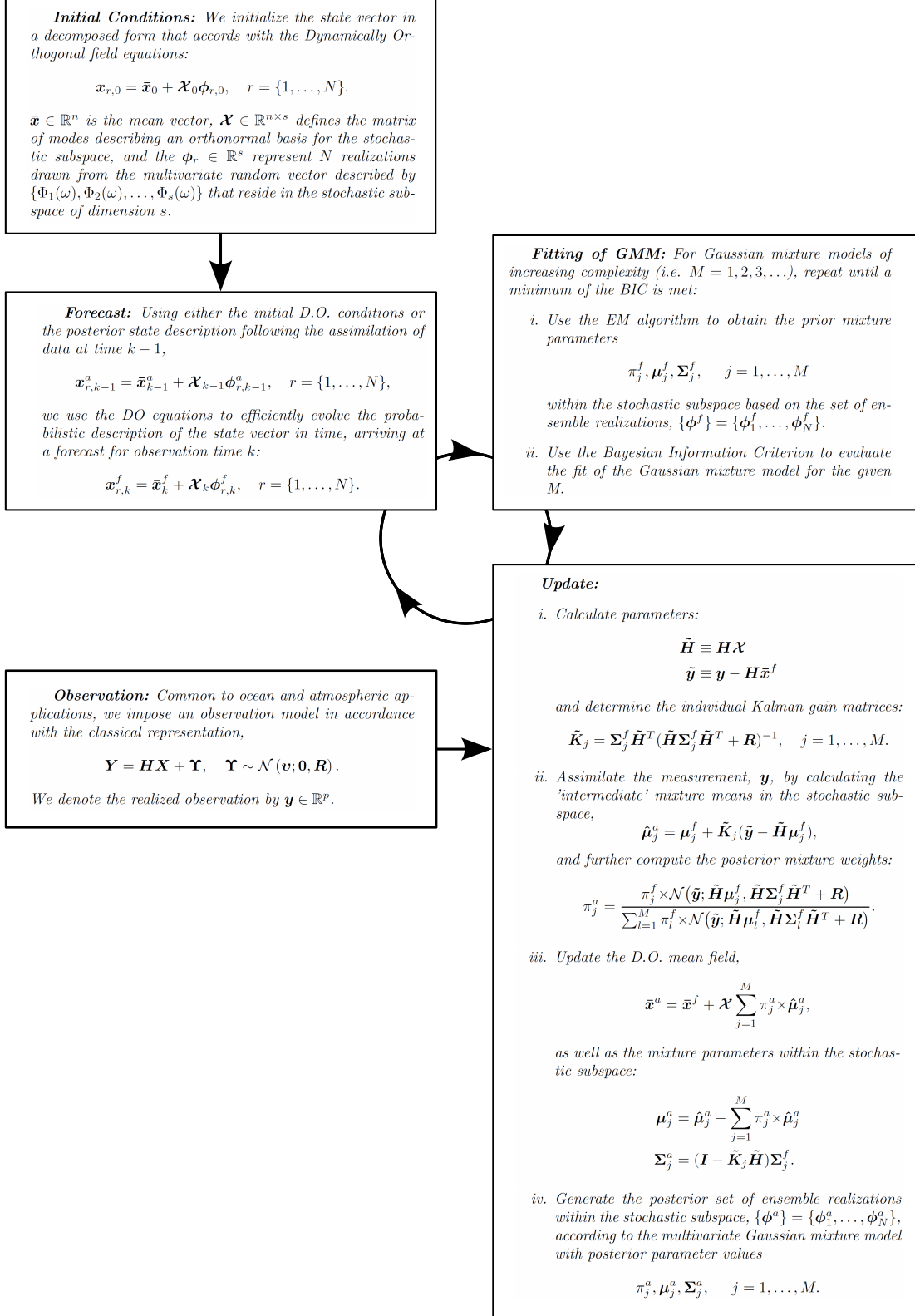


Figure 3-2: GMM-DO filter flowchart.

For simplicity, we'll take the true field to coincide with one of the realizations, i.e.

$$\mathbf{x}^t = \bar{\mathbf{x}}^f + \mathbf{X}\phi_1.$$

We make noisy measurements of the first and third elements of the state vector, i.e.

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

normally distributed with an error covariance matrix given by

$$\mathbf{R} = \sigma_{obs}^2 \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

where $\sigma_{obs} = 5$. We illustrate all of the above in panel (a) of figure 3-3.

With this, we proceed with the update step, using the GMM-DO flowchart, figure 3-2, as reference. In what follows, for the purposes of illustration, we bypass the application of the Bayesian Information Criterion and rather present results directly for fitted Gaussian mixture models of complexity, M , one and two. We note that the latter would – with high probability – be obtained using the BIC criterion.

Fitting of GMM

1. Use the EM algorithm to obtain the prior mixture parameters

$$\pi_j^f, \boldsymbol{\mu}_j^f, \boldsymbol{\Sigma}_j^f, \quad j = 1, \dots, M$$

within the stochastic subspace based on the set of ensemble realizations, $\{\boldsymbol{\phi}^f\} = \{\boldsymbol{\phi}_1^f, \dots, \boldsymbol{\phi}_{100}^f\}$. The identified mixtures (of complexities one and two), along with their marginal distributions, are displayed in panel (b-i) of figure 3-3.

Update

1. Calculate parameters:

$$\begin{aligned}\tilde{\mathbf{H}} &\equiv \mathbf{H}\mathcal{X} \\ \tilde{\mathbf{y}} &\equiv \mathbf{y} - \mathbf{H}\bar{\mathbf{x}}^f\end{aligned}$$

and determine the mixture Kalman gain matrices:

$$\tilde{\mathbf{K}}_j = \Sigma_j^f \tilde{\mathbf{H}}^T (\tilde{\mathbf{H}} \Sigma_j^f \tilde{\mathbf{H}}^T + \mathbf{R})^{-1}.$$

2. Assimilate the measurements, \mathbf{y} , by calculating the 'intermediate' mixture means in the stochastic subspace,

$$\hat{\boldsymbol{\mu}}_j^a = \boldsymbol{\mu}_j^f + \tilde{\mathbf{K}}_j(\tilde{\mathbf{y}} - \tilde{\mathbf{H}}\boldsymbol{\mu}_j^f),$$

and further compute the posterior mixture weights:

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\tilde{\mathbf{y}}; \tilde{\mathbf{H}}\boldsymbol{\mu}_j^f, \tilde{\mathbf{H}}\Sigma_j^f\tilde{\mathbf{H}}^T + \mathbf{R})}{\sum_{l=1}^M \pi_l^f \times \mathcal{N}(\tilde{\mathbf{y}}; \tilde{\mathbf{H}}\boldsymbol{\mu}_l^f, \tilde{\mathbf{H}}\Sigma_l^f\tilde{\mathbf{H}}^T + \mathbf{R})}.$$

3. Update the D.O. mean field (displayed in panel (c), column (ii) of figure 3-3),

$$\bar{\mathbf{x}}^a = \bar{\mathbf{x}}^f + \mathcal{X} \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a,$$

as well as the mixture parameters within the stochastic subspace:

$$\begin{aligned}\boldsymbol{\mu}_j^a &= \hat{\boldsymbol{\mu}}_j^a - \sum_{j=1}^M \pi_j^a \times \hat{\boldsymbol{\mu}}_j^a \\ \Sigma_j^a &= (\mathbf{I} - \tilde{\mathbf{K}}_j \tilde{\mathbf{H}}) \Sigma_j^f.\end{aligned}$$

4. Generate the posterior set of ensemble realizations within the stochastic sub-

space, $\{\phi^a\} = \{\phi_1^a, \dots, \phi_{100}^a\}$, according to the multivariate Gaussian mixture model with posterior parameter values

$$\pi_j^a, \mu_j^a, \Sigma_j^a, \quad j = 1, \dots, M.$$

We display the posterior set of realizations in panel (c-i) of figure 3-3.

By way of this simple example, we may draw two conclusions on the benefits of adopting the update procedure of the GMM-DO filter. Firstly, given the initial non-Gaussian statistics, the Gaussian mixture model (GMM) – of mixture complexity two – was found to provide a superior posterior estimate for the true solution when compared with the Gaussian parametric distribution (PD) (as evidenced by their posterior means displayed in panel (c-ii) of figure 3-3). In particular, due to the PD’s conservative estimate for the covariance matrix of the true probability density function (see panel (b-i) of figure 3-3), the noisy measurements were favored over the prior mean estimate, essentially resulting in an ‘overshoot’ of its posterior estimate for the mean. Given the GMM’s accurate representation of the prior statistics, on the other hand, the prior information was accurately balanced with that due to the measurements, resulting in a successful update of the mean state. While this was to be expected given the initial bimodal structure, previous arguments suggest that this holds for arbitrary distributions as long as the fitting of Gaussian mixture models based on the EM algorithm and the Bayesian Information Criterion provides a good approximation of the true probability density function.

The second conclusion refers to the posterior statistics, represented by the subspace realizations, $\{\phi^a\} = \{\phi_1^a, \dots, \phi_{100}^a\}$, in panel (c-i) of figure 3-3. In addition to the GMM’s accurate approximation of the true solution, the compactness of the posterior set of realizations emphasizes its added belief in this estimate. The accuracy of the posterior representation of the true statistics clearly affects future assimilations. We hypothesize that the GMM-DO filter outperforms simpler schemes in this respect. In chapters 4 and 5, we support this hypothesis by applying the GMM-DO filter in a dynamical systems setting.

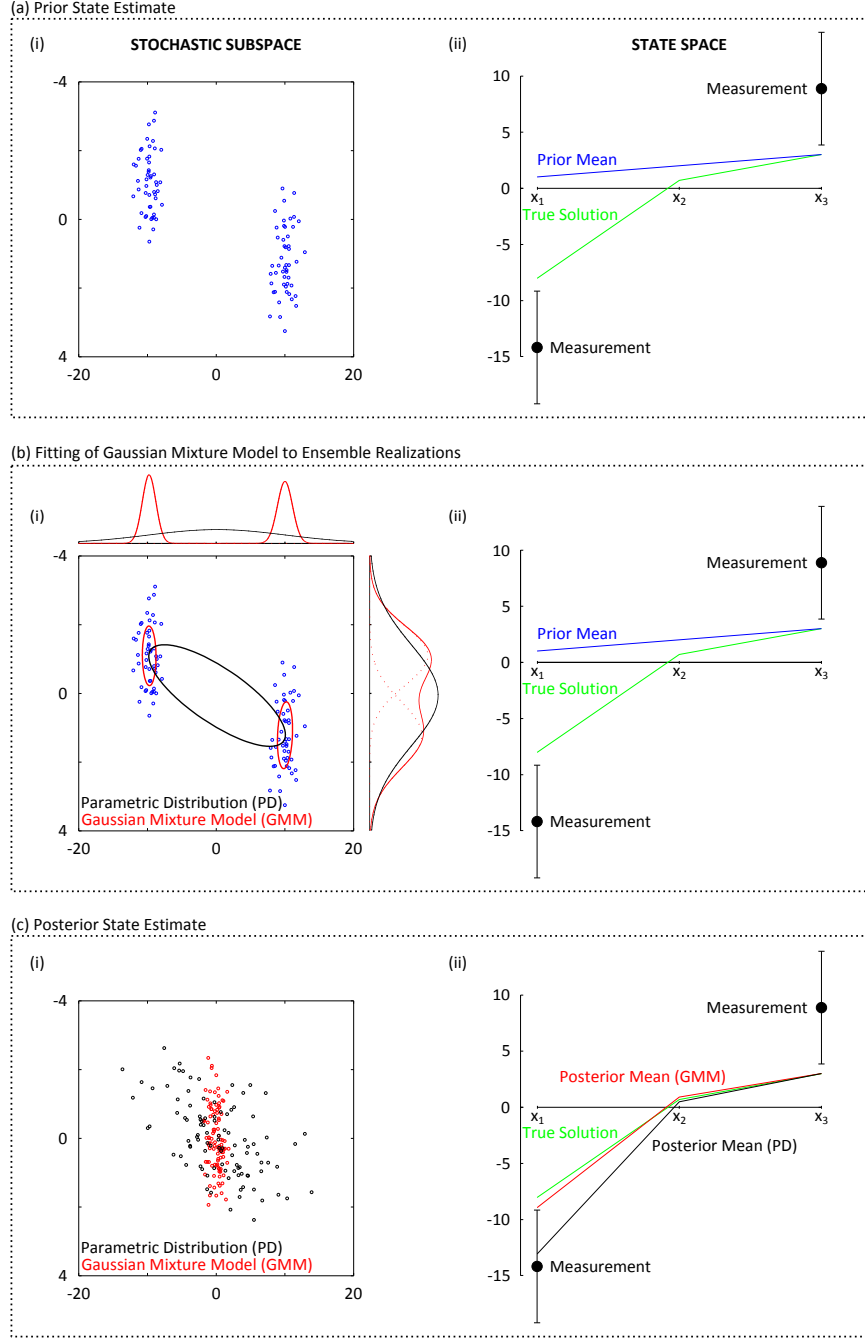


Figure 3-3: GMM-DO filter update. In column (i), we plot the set of ensemble realizations within the stochastic subspace, $\{\phi\}$; in column (ii), we display the information relevant to the state space. Panel (a) shows the prior state estimate; in panel (b), we show the fitting of Gaussian mixture models of complexity $M = 1$ (PD) and $M = 2$ (GMM), and plot their marginal distributions for each of the stochastic coefficients; in panel (c), we provide the posterior state estimate again in the decomposed form that accords with the D.O. equations.

3.5.6 Remarks, Modifications and Extensions

In what follows, we develop a few remarks, modifications and extensions to the GMM-DO filter:

EM algorithm in p -dominant space of stochastic subspace

Estimating and manipulating non-trivial probability density functions in high-dimensional spaces can be a difficult task (Bengtsson et al., 2003). Heuristic arguments, for instance, suggest that the number of realizations required to accurately represent multivariate probability density functions grows exponentially with the dimension of the space (Silverman, 1992). For this reason, it is worthwhile investigating modifications to the current procedure for fitting Gaussian mixture models to realizations in which the dimension of the stochastic subspace may pose a difficulty.

As in the main body of this paper, we let the dimension of the stochastic subspace be s , i.e. $\mathbf{X} \in \mathbb{R}^{n \times s}$. When deemed necessary on the grounds of tractability, we can limit our estimation of mixtures to the stochastic coefficients associated with the space defined by the p most dominant modes, denoting this $\mathbf{X}^p \in \mathbb{R}^{n \times p}$. We in turn approximate the stochastic coefficients of the remaining $s - p$ modes, $\{\Phi_{p+1}, \dots, \Phi_s\}$, as zero mean Gaussian with (co)variances based on the sample covariance matrix. For our purposes, an obvious and appropriate measure of dominance is the variance of each of the stochastic coefficients.

In what follows, we describe the modified EM algorithm for Gaussian mixture models in a p -dominant space of the stochastic subspace:

Definition: EM algorithm in p -dominant space of stochastic subspace

Given the set of ensemble realizations, $\{\phi\} \in \mathbb{R}^{s \times N}$, associated with the stochastic subspace, $\mathbf{X} \in \mathbb{R}^{n \times s}$, we limit our attention to the ensemble set, $\{\phi^p\} \in \mathbb{R}^{p \times N}$, associated with the p -dominant reduced space, $\mathbf{X}^p \in \mathbb{R}^{n \times p}$, of the stochastic subspace (i.e. $p \leq s$). We define p such that the following holds:

$$1 \geq \frac{\sum_{i=1}^p \text{var}(\Phi_i)}{\sum_{j=1}^s \text{var}(\Phi_j)} \geq C \geq 0, \quad (3.163)$$

where C denotes a user-specified constant chosen such that the majority of the energy in the stochastic subspace is captured. (Note, we assume that the stochastic coefficients, Φ_i , are ordered by decreasing variance, i.e. $\text{var}(\Phi_1) \geq \text{var}(\Phi_2) \geq \dots \geq \text{var}(\Phi_s)$.)

Therefore, based on the reduced ensemble set, $\{\boldsymbol{\phi}^p\} = \{\boldsymbol{\phi}_1^p, \dots, \boldsymbol{\phi}_N^p\}$, and an initial parameter estimate,

$$\boldsymbol{\theta}^{p,(0)} = \{\pi_1^{p,(0)}, \dots, \pi_M^{p,(0)}, \bar{\mathbf{x}}_1^{p,(0)}, \dots, \bar{\mathbf{x}}_M^{p,(0)}, \mathbf{P}_1^{p,(0)}, \dots, \mathbf{P}_M^{p,(0)}\},$$

appropriately sized for the reduced EM estimation procedure, we repeat until convergence:

- For all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$, use the present parameter estimate, $\boldsymbol{\theta}^{p,(k)}$, to form

$$\tau_j(\boldsymbol{\phi}_i^p; \boldsymbol{\theta}^{p,(k)}) = \frac{\pi_j^{p,(k)} \mathcal{N}(\boldsymbol{\phi}_i^p; \boldsymbol{\mu}_j^{p,(k)}, \boldsymbol{\Sigma}_j^{p,(k)})}{\sum_{m=1}^M \pi_m^{p,(k)} \mathcal{N}(\boldsymbol{\phi}_i^p; \boldsymbol{\mu}_m^{p,(k)}, \boldsymbol{\Sigma}_m^{p,(k)})}. \quad (3.164)$$

- For all $j \in \{1, 2, \dots, M\}$, update the parameter estimate, $\boldsymbol{\theta}^{p,(k+1)}$, according to

$$\pi_j^{p,(k+1)} = \frac{N_j^{p,(k)}}{N} \quad (3.165)$$

$$\boldsymbol{\mu}_j^{p,(k+1)} = \frac{1}{N_j^{p,(k)}} \sum_{i=1}^N \tau_j(\boldsymbol{\phi}_i^p; \boldsymbol{\theta}^{p,(k)}) \boldsymbol{\phi}_i^p \quad (3.166)$$

$$\boldsymbol{\Sigma}_j^{p,(k+1)} = \frac{1}{N_j^{p,(k)}} \sum_{i=1}^N \tau_j(\boldsymbol{\phi}_i^p; \boldsymbol{\theta}^{p,(k)}) (\boldsymbol{\phi}_i^p - \boldsymbol{\mu}_j^{p,(k+1)}) (\boldsymbol{\phi}_i^p - \boldsymbol{\mu}_j^{p,(k+1)})^T \quad (3.167)$$

where

$$N_j^{p,(k)} = \sum_{i=1}^N \tau_j(\boldsymbol{\phi}_i^p; \boldsymbol{\theta}^{p,(k)}). \quad (3.168)$$

Once converged, we obtain the Gaussian mixture model associated with the stochastic subspace, $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n \times s}$, by embedding the above p -dominant vectors and matrices into

their appropriately sized equivalent as follows:

$$\boldsymbol{\mu}_j = \begin{bmatrix} \boldsymbol{\mu}_j^p \\ \mathbf{0} \end{bmatrix}, \quad \mathbf{0} \in \mathbb{R}^{s-p} \quad (3.169)$$

and

$$\boldsymbol{\Sigma}_j = \begin{bmatrix} \boldsymbol{\Sigma}_j^p & \boldsymbol{\Sigma}_{1:p,(p+1):s} \\ \boldsymbol{\Sigma}_{(p+1):s,1:p} & \boldsymbol{\Sigma}_{(p+1):s,(p+1):s} \end{bmatrix}, \quad (3.170)$$

where $\boldsymbol{\Sigma} \in \mathbb{R}^{s \times s}$ is the sample covariance matrix,

$$\boldsymbol{\Sigma} = \frac{1}{N-1} \sum_{i=1}^N \boldsymbol{\phi} \boldsymbol{\phi}^T, \quad (3.171)$$

and $\boldsymbol{\Sigma}_{a:b,c:d}$ denotes the sub-matrix of $\boldsymbol{\Sigma}$ defined by rows a - b and columns c - d . We arrive at equations (3.169) and (3.170) by application of the Law of Iterated Expectations and the Law of Total Variance, respectively (see e.g. Bertsekas and Tsitsiklis (2008)), ensuring that the stochastic coefficients, $\{\Phi_{p+1}, \dots, \Phi_s\}$, are approximated as zero mean Gaussian with variances based on the sample covariance matrix.

Constraining the mean of the Gaussian Mixture Model

In the D.O. formulation, equation (3.108), we impose a zero-mean constraint on the random vector, $\boldsymbol{\Phi}(\omega)$, represented by the ensemble set, $\{\boldsymbol{\phi}\}$. Since the EM algorithm is an unconstrained optimization procedure in this regard, however, the EM fit of the Gaussian mixture model may not necessarily itself be of zero mean, i.e.

$$\sum_{j=1}^M \pi_j \boldsymbol{\mu}_j \neq \mathbf{0}. \quad (3.172)$$

While the test cases presented in part II of this two-part paper give evidence to suggest that this is little cause for concern (namely that this mean offset is negligible), we nonetheless propose two possible remedies:

1. When forming the auxiliary function in equation (3.47), one may add the con-

straint that the Gaussian mixture model be of zero mean, i.e.

$$\sum_{j=1}^M \pi_j \boldsymbol{\mu}_j = 0, \quad (3.173)$$

thus obtaining for the auxiliary function:

$$\begin{aligned} \Lambda = & \sum_{j=1}^M \sum_{i=1}^N \tau_j(\mathbf{x}_i; \boldsymbol{\theta}^{(k)}) \left(\log \pi_j - \frac{k}{2} \log 2\pi - \frac{1}{2} \log |\mathbf{P}_j| \right. \\ & \left. - \frac{1}{2} (\mathbf{x}_i - \bar{\mathbf{x}}_j)^T \mathbf{P}_j^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_j) \right) + \lambda_1 \left(\sum_{k=1}^M \pi_k - 1 \right) + \lambda_2 \sum_{l=1}^M \pi_l \boldsymbol{\mu}_l. \end{aligned} \quad (3.174)$$

While this clearly provides a viable solution, a closer inspection reveals that such a constraint destroys the simplicity of the EM algorithm. Particularly, with this added constraint, closed form equations for the updated mixture parameters, equations (3.64) - (3.66), no longer arise. Rather, the parameters to be optimized become coupled.

2. A complementary approach proceeds by obtaining an estimate for the parameter vector by means of the regular EM algorithm for Gaussian mixture models. This estimate may in turn be fed as an initial guess to the coupled set of equations in (i), for which an optimization procedure of choice may be utilized. With this, it is estimated that few iterations are needed to arrive at the optimal set of parameter values.

We leave further investigations into each of these approaches for a future work.

3.6 Literature Review

Gaussian mixture models are by no means a new phenomenon within the data assimilation community. In this section, we therefore provide a review of appropriate literature that places the GMM-DO filter in the context of past and recent schemes that have approached filtering in a similar manner. We wish to show the evolution of

such methods over the past few decades, ultimately outlining the shortcomings and limitations overcome by the GMM-DO filter.

Alspach and Sorenson (1972)

Gaussian mixture models were essentially first addressed in the context of filtering theory in the seminal paper by Alspach and Sorenson (1972). Here, the authors were particularly motivated by the inappropriate use of Gaussian distributions, stating that

"the Gaussian [parametric] approximation greatly reduces the amount of information that is contained in the true density, particularly when it is multimodal".

They emphasized the ability of Gaussian mixture models to approximate arbitrary densities, all the while retaining the familiar computational tractability when placed in the context of Bayesian inference.

Based on an approximation of the known, initial (non-Gaussian) distribution by a Gaussian mixture model of complexity M , their scheme would essentially run M extended Kalman filters in parallel – one for each mixture – coupled solely through the mixture weights. Their update equation would thus take a form structurally similar to that of the GMM-DO filter, set aside the latter's efficient use of the stochastic subspace. While the authors freed themselves of the Gaussian, parametric constraint, their scheme remained grounded in linear theory, however, having been inspired by the moderate success of the Extended Kalman filter.

In their paper, the authors made no mention of the appropriate mixture complexity, nor the manner in which the initial mixture parameters were obtained. Moreover, while they alluded to the potential necessity for having to intermittently restart the distribution – either due to the poor mismatch of forecast distribution with observations, or the collapse of weights onto a single mixture – no appropriate remedies were proposed.

Anderson and Anderson (1999)

Anderson and Anderson (1999), most likely inspired by the recent advances of ensemble methods within the data assimilation community (e.g. Evensen (1994) and Lermusiaux (1997)), extended the work of Alspach and Sorenson (1972) by resorting to a Monte Carlo approach for evolving the state estimate. By arguing that

”one of the fundamental advantages of a Monte Carlo approach [is its] ability to represent non-Gaussian probability distributions”,

they chose to approximate the density in question based on a kernel approach,

$$p_{\mathbf{X}^f}(\mathbf{x}^f) = \frac{1}{N} \sum_{i=1}^N \mathcal{N}(\mathbf{x}^f; \mathbf{x}_i^f, \alpha \Sigma^f), \quad (3.175)$$

with \mathbf{x}_i representing the particle locations; Σ the ensemble covariance matrix; and α an heuristically chosen scaling parameter.

Upon assimilating data, \mathbf{y} , from a Gaussian observation model, their posterior distribution for the state vector would thus take the familiar form

$$p_{\mathbf{X}^a}(\mathbf{x}^a) = p_{\mathbf{X}^f|\mathbf{Y}}(\mathbf{x}^f|\mathbf{y}) = \sum_{i=1}^N \pi_i^a \mathcal{N}(\mathbf{x}^a; \mathbf{x}_i^a, \alpha \Sigma^a), \quad (3.176)$$

from which they would draw N new particles.

The authors justifiably argued for its advantages over filters that would invoke the regular parametric Gaussian distribution, giving as example their respective performances when applied to the Lorenz-63 model (Lorenz, 1963). While the kernel filter would essentially represent states solely in accordance with the model dynamics, simpler filters would potentially assign finite probability to regions of state space never visited. Such is the case depicted in figure 3-4.

The main drawback of the filter lay in the vague arguments for choosing the scaling parameter, α . Specifically, the authors stated that while

”a number of methods for computing the constant covariance reduction factor, α , have been developed, ... the value of α is often subsumed into a

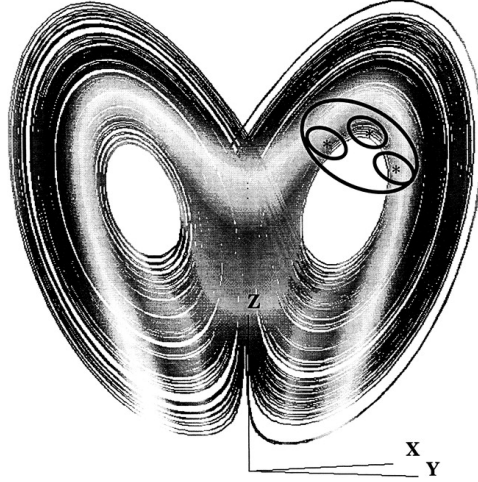


Figure 3-4: Schematic representation of advantages of kernel over single Gaussian filter for a low-order model. The background is a projection of a trajectory from the Lorenz-63 model showing the attractor structure. Superimposed is an idealized image of the single Gaussian (outer curve) and kernel (inner curves) prior distributions for a three-member ensemble. (Anderson and Anderson, 1999)

tuning constant and so does not need to be calculated explicitly. ... Tuning a filter for a real system is complicated ... [and] must be chosen with care."

Hoteit et al. (2007) would later extend the filter by allowing the particles to carry uneven weights, drawing on concepts familiar to the particle filter. To avoid the collapse of weights onto only a few particles, they proposed a number of interesting methods for resampling. While effective, these ideas will not be pursued further here.

Bengtsson et al. (2003)

Bengtsson et al. (2003) expressed a concern over Anderson and Anderson's use of kernel density methods for approximating distributions, arguing that the use of

"scaled versions of the full ensemble covariance around each center in the mixture ... cannot adapt as easily to local structure in the forecast distribution".

(We greatly share this opinion and equally express it in this thesis.) Instead, they proposed to approximate the set of ensembles by a Gaussian mixture model (of complexity less than the number of ensemble members), in which the mixture parameters

were estimated using knowledge of the ensemble distribution. They stated that such an approach would provide a more accurate local approximation to the true probability distribution, as is the point of view taken in this thesis.

Their scheme essentially proceeded as follows: M ensemble members would *arbitrarily* be chosen to act as means for the proposed Gaussian mixtures, from which the N_n nearest neighbors to each center would be used to approximate their respective mixture covariance matrices. From here, one would proceed with the Bayesian update, inspired in part by the particle operations of the Ensemble Kalman filter.

As with Alspach and Sorensen, the authors left unanswered methods for determining both the mixture complexity, M , as well as the appropriate choice of N_n , the number of nearest neighbors. Furthermore, the choice of mixture centers, based on the *arbitrary* sampling of ensemble members, would certainly invite for sampling noise.

In their paper, the authors further expressed difficulties associated with manipulating probability density functions in high dimensional spaces. As a consequence, they introduced a hierarchy of adaptations to the aforementioned filter, in which they invoked varying degrees of localization approximations, all again based on heuristic arguments. In this thesis, we overcome all such approximations by adopting the D.O. framework.

Smith (2007)

Indirectly extending the work by Bengtsson et al., Smith (2007) proposed to use the EM algorithm to uncover the underlying structure of the particle distribution, thus replacing the former heuristic arguments. In his paper, he modified the ensemble Kalman filter to allow for a Gaussian mixture representation of the prior distribution, using Akaike’s Information Criterion (AIC) as the method for selecting the appropriate mixture complexity. (As a side note, McLachlan and Peel (2000) found BIC to outperform AIC when fitting Gaussian mixtures to data; specifically, the latter would have the tendency to overestimate the mixture complexity.) Similar to the scheme proposed by Bengtsson et al., Smith retained the concept of operating

on individual ensemble members, invoking only the approximation that the posterior distribution be normally distributed. His Cluster Ensemble Kalman filter proceeded as follows (adopting notation previously applied in this document):

1. Determine the mixture complexity, M , using Akaike's Information Criterion.
2. Apply the EM algorithm to the ensemble of states *in the full state space*, \mathbf{x}_i^f , to obtain the ML estimate for the Gaussian mixture parameter vector,

$$\boldsymbol{\theta} = \{\pi_1^f, \dots, \pi_M^f, \bar{\mathbf{x}}_1^f, \dots, \bar{\mathbf{x}}_M^f, \mathbf{P}_1^f, \dots, \mathbf{P}_M^f\}, \quad (3.177)$$

as well as the weights

$$w_{i,j} = \tau_j(\mathbf{x}_i^f; \boldsymbol{\theta}) = \frac{\pi_j^f \mathcal{N}(\mathbf{x}_i^f; \bar{\mathbf{x}}_j^f, \mathbf{P}_j^f)}{\sum_{q=1}^M \pi_q^f \mathcal{N}(\mathbf{x}_i^f; \bar{\mathbf{x}}_q^f, \mathbf{P}_q^f)}. \quad (3.178)$$

3. For each component distribution, j , compute:

$$\mathbf{P}_j^f \mathbf{H}^T = \sum_{i=1}^m w_{i,j} (\mathbf{x}_i^f - \bar{\mathbf{x}}_j) (\mathbf{H} \mathbf{x}_i^f - \mathbf{H} \bar{\mathbf{x}}_j)^T \quad (3.179)$$

$$\mathbf{H} \mathbf{P}_j^f \mathbf{H}^T = \sum_{i=1}^m w_{i,j} (\mathbf{H} \mathbf{x}_i^f - \mathbf{H} \bar{\mathbf{x}}_j) (\mathbf{H} \mathbf{x}_i^f - \mathbf{H} \bar{\mathbf{x}}_j)^T \quad (3.180)$$

with

$$\mathbf{K}_j = \mathbf{P}_j^f \mathbf{H}^T (\mathbf{H} \mathbf{P}_j^f \mathbf{H}^T + \mathbf{R})^{-1}. \quad (3.181)$$

4. Compute the Kalman update for each ensemble member, \mathbf{x}_i , under each component distribution.

$$\mathbf{x}_i^{a,j} = \mathbf{x}_i^f + \mathbf{K}_j (\mathbf{y} - \mathbf{H} \mathbf{x}_i^f - \mathbf{e}_i), \quad (3.182)$$

with

$$\mathbf{e}_i \sim \mathcal{N}(\mathbf{e}; \mathbf{0}, \mathbf{R}). \quad (3.183)$$

5. Update the mixture weights based on the observed data, \mathbf{y} ,

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_j^f, \mathbf{H}\mathbf{P}_j^f\mathbf{H}^T + \mathbf{R})}{\sum_{q=1}^M \pi_q^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_q^f, \mathbf{H}\mathbf{P}_q^f\mathbf{H}^T + \mathbf{R})}. \quad (3.184)$$

6. Create the remapped analysis ensemble:

$$\mathbf{x}_i^a = \sum_{j=1}^M \pi_j^a \left(\bar{\mathbf{x}}_j^a + \sqrt{\mathbf{P}_j^a} \left(\sum_{k=1}^M w_{i,k} (\sqrt{\mathbf{P}_j^a})^{-1} (\mathbf{x}_i^{a,k} - \bar{\mathbf{x}}_k^a) \right) \right). \quad (3.185)$$

(Smith justified this latter procedure by wanting to approximate the posterior distribution by a parametric Gaussian distribution with $\bar{\mathbf{x}}^a = \sum_{j=1}^M \pi_j^a \bar{\mathbf{x}}_j^a$ and $\mathbf{P}^a = \sum_{j=1}^M \pi_j^a \mathbf{P}_j^a$. By the Law of Total Variance, we note that the latter approximation is incorrect (see the Appendix).)

In his paper, he applied his Cluster Ensemble Kalman Filter to a simple two-dimensional phytoplankton-zooplankton biological model. While successful for such simple models, he emphasized the difficulties of extending his scheme to test cases of larger dimensions, making, however, the useful comment that

“the state space could be projected onto a lower dimensional space depicting some relevant phenomenon, and the full covariance matrix in this state space could be used.”

By adopting the D.O. equations, we exactly allow for this.

Dovera and Rossa (2010)

Dovera and Rossa (2010) modified the approach of Smith by attempting to overcome the constraint that the posterior distribution be Gaussian. Their scheme proceeded as follows:

1. For each component $j \in \{1, \dots, M\}$, compute the updated mixture weights based on the observed data, \mathbf{y} ,

$$\pi_j^a = \frac{\pi_j^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_j^f, \mathbf{H}\mathbf{P}_j^f\mathbf{H}^T + \mathbf{R})}{\sum_{q=1}^M \pi_q^f \times \mathcal{N}(\mathbf{y}; \mathbf{H}\bar{\mathbf{x}}_q^f, \mathbf{H}\mathbf{P}_q^f\mathbf{H}^T + \mathbf{R})}. \quad (3.186)$$

2. Loop on ensemble members. For each \mathbf{x}_i^f :

- set k as the known component of the member \mathbf{x}_i^f ;
- generate a random index of new component $l \in \{1, \dots, M\}$ according to the discrete distribution given by $\{\pi_1^a, \dots, \pi_M^a\}$;
- compute the auxiliary vector $\mathbf{x}_i^{f'}$ according to

$$\mathbf{x}_i^{f'} = \bar{\mathbf{x}}_l^f + \sqrt{\mathbf{P}_l^f}(\sqrt{\mathbf{P}_j^f})^{-1}(\mathbf{x}_i^f - \bar{\mathbf{x}}_j^f) \quad (3.187)$$

- compute the updated vector \mathbf{x}_i^a using the updating equation for component l on the auxiliary vector $\mathbf{x}_i^{f'}$:

$$\mathbf{x}_i^a = \mathbf{x}_i^{f'} + \mathbf{K}_l(\mathbf{y} - \mathbf{H}\mathbf{x}_i^{f'}). \quad (3.188)$$

The authors successfully applied their scheme to both the Lorenz-63 model as well as a two-dimensional reservoir model, as expected outperforming the regular ensemble Kalman filter. As with previous schemes, however, they noted the problems caused by systems of high dimensionality, stating that

"this restriction poses two obstacles in the numerical implementation of the proposed method for large scale applications. The first problem is the application of the EM algorithm to the forecasted ensemble ... The second problem is due to the covariance matrices factorization by Cholesky decomposition ... that cannot be addressed directly in a high dimensional space."

As previously done, the authors adopted a number of localization arguments to overcome the aforementioned burdens. They specifically hypothesized that the correlation matrices be local, therefore retaining only the model states in the vicinity of the observations. By adopting the D.O. framework within the GMM-DO filter, we address – and ultimately eliminate – all such approximations.

Summary

Past literature has identified the advantages of adopting Gaussian mixture models when assimilating data, allowing the update step to capture and retain potential non-Gaussian structures. Its success has been shown using a number of simplified test cases, including the classic Lorenz-63 model. Later publications, specifically those due to Smith (2007) and Dovera and Rossa (2010), have further made use of both the EM algorithm and model selection criteria for arriving at appropriate mixture parameters, resulting in a better resolution of the probability density function. All of this is equally utilized in the GMM-DO filter.

The novelty of the GMM-DO filter lies in having identified the necessity to couple the previous concepts with an efficient reduced order model, specifically the Dynamically Orthogonal field equations due to Sapsis and Lermusiaux (2009). With this, we address prior limitations caused by the size of the state space. Particularly, we make obsolete ad hoc localization procedures previously adopted – with limited success – by filters introduced in this section. We further stray from operating on individual ensemble members; rather, we efficiently manipulate directly the determined Gaussian mixture model exactly within the stochastic subspace under Bayes’ Law by the assumption that the aforementioned Gaussian mixture model captures the true non-Gaussian structures.

In conclusion, we present the GMM-DO filter as an efficient, data-driven assimilation scheme that respects nonlinear dynamics and captures non-Gaussian statistics, obviating the use of heuristic arguments. By limiting our attention to a dominant stochastic subspace of the total state space, we specifically bridge an important gap previously identified in the literature. In the following chapters, we apply the GMM-

DO filter to a number of test cases with the intention of evaluating its performance when compared against popular filters currently in use.

Chapter 4

Application 1: Double Well Diffusion Experiment

The Double Well Diffusion Experiment has served as a test case for a number of data assimilation schemes (e.g. Miller et al. (1994)), recently among them the Maximum Entropy filter (MEF) introduced by Eyink and Kim (2006) and outlined in the appendix of this thesis. Due to its bimodal climatological distribution, the experiment lends itself well to filters that aim to extract non-Gaussian structures.

Given the experiment's low dimensionality, the Dynamically Orthogonal field equations, introduced as an integral part of the GMM-DO filter, will here be of little use and thus excluded. Instead, the purpose of this test case will be to validate the use of the EM algorithm with Gaussian mixture models in a dynamical setting.

After introducing the physics of the experiment, we will evaluate the performance of the GMM-DO filter against the Ensemble Kalman filter (EnKF) and the Maximum Entropy filter, the latter of which is particularly well-suited to the given test case.

4.1 Introduction

In the Double Well Diffusion Experiment, our goal is to track the location of a ball, located in one of two wells. The ball is forced under pseudo-gravity and externally excited by white noise. Specifically, the location of the ball evolves according to the

following stochastic differential equation (Miller et al., 1994):

$$dx = f(x)dt + \kappa d\Gamma(t), \quad \Gamma \sim \mathcal{N}(\gamma; 0, 1), \quad (4.1)$$

with

$$f(x) = 4x - 4x^3 \quad (4.2)$$

essentially acting as the gravitational force (see figure 4-1 for a graphical depiction). We understand κ as a diffusion coefficient that tunes the strength of the stochastic forcing. We also note that $x \in \mathbb{R}$.

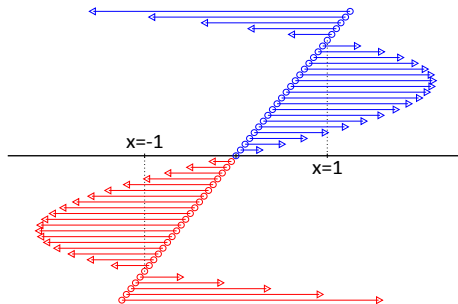


Figure 4-1: Forcing Function, $f(x)$. At any location (o), x , the ball is forced under pseudo-gravity in the direction indicated by the appropriate vector. The magnitude of the vector corresponds to the strength of the forcing. We note that there exists an unstable node at the origin, and two stable nodes at $x = \pm 1$, corresponding to the minima of the wells.

We occasionally get access to direct, but noisy, measurements of the current ball location, modeled as:

$$p_{Y|X}(y|x) \sim \mathcal{N}(y; x, \sigma_o^2). \quad (4.3)$$

From these measurements, we wish to infer the current location of the ball. We are thus faced with a filtering task.

The Double Well Diffusion Experiment is an ergodic Markov Chain (see e.g. Cover and Thomas (2006)) and therefore possesses a stationary distribution (from hereon *climatological* distribution). It can be shown that this distribution satisfies (Eyink and Kim, 2006):

$$q_X(x) \propto e^{-\frac{2x^4 - 4x^2}{\kappa^2}}, \quad (4.4)$$

which may adequately be approximated by a Gaussian mixture model of complexity two. Specifically, we write for the approximation of the climatological distribution:

$$q_X(x) = \sum_{m=1}^2 w_m \mathcal{N}(x; \mu_m, \sigma_m^2), \quad (4.5)$$

with, by arguments of symmetry, the following properties:

$$w_1 = w_2 = 0.5 \quad (4.6)$$

$$-\mu_1 = \mu_2 = \mu \quad (4.7)$$

$$\sigma_1^2 = \sigma_2^2 = \sigma^2. \quad (4.8)$$

For the particular case of $\kappa = 0.40$, Eyink and Kim (2006) determined (by an unspecified procedure and metric) the mean and variance of the Gaussian mixture model to be $\mu = 0.98$ and $\sigma^2 = 0.011$, respectively. This is plotted against the exact distribution in figure 4-2:

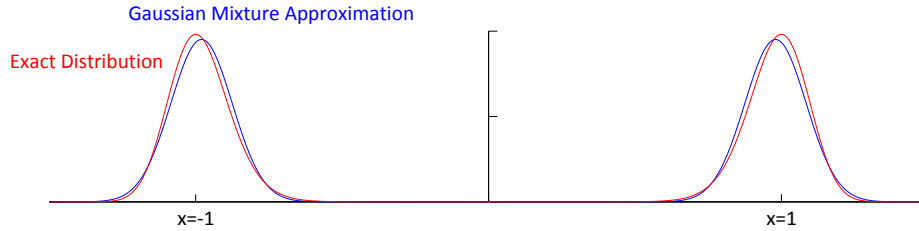


Figure 4-2: Climatological distribution and Gaussian mixture approximation for $\kappa = 0.40$. In accordance with intuition, the distributions are bimodal, appropriately centered on the minima of each of the two wells.

The choice of κ determines the average residence time of the ball spent in any one well. For instance, according to Eyink and Kim (2006), for the case of $\kappa = 0.40$, this residence time is $\tilde{\tau}_{res} \approx 10^5$ with transitions from one well to the other taking only $\tilde{\tau}_{trans} \approx 10^1$. For small values of κ , we are thus faced with a phenomenon perhaps most accurately characterized as a noisy switch.

For the sake of illustration, we plot in figure 4-3 a viable, arbitrarily generated, trajectory for the ball under the governing stochastic differential equation (4.1) for the case of $\kappa = 0.45$. We have purposely centered the plot about a transition of the

ball from one well to the other, as this event will be of central interest to us. We have framed the transition within a suitable time window that will allow for appropriate analysis. Superimposed onto the trajectory of the ball are noisy measurements with

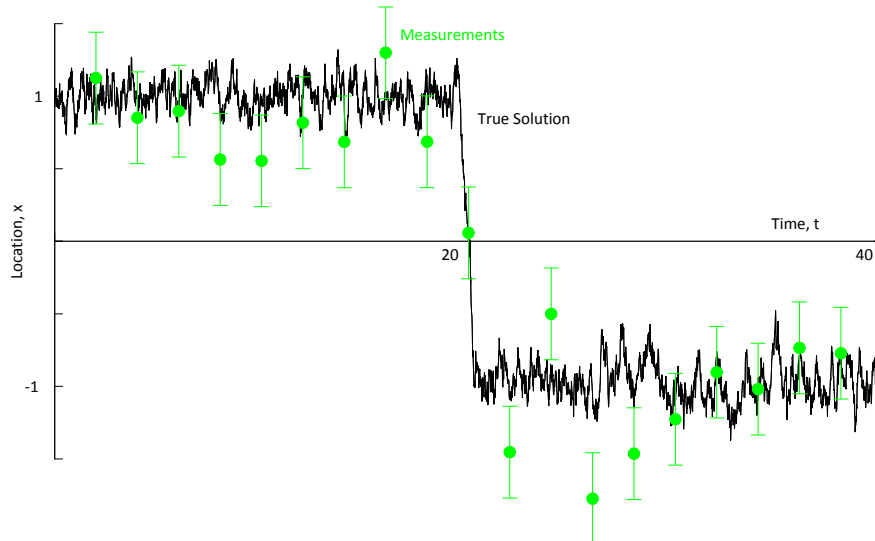


Figure 4-3: Example trajectory of the ball for $\kappa = 0.45$. The horizontal axis denotes time; the vertical axis the location of the ball. Superimposed onto the plot are intermittent measurements, shown in green, with their associated uncertainties.

their standard deviation indicated by the length of the error bar.

4.2 Procedure

We wish to evaluate the performance of the GMM-DO filter against the Ensemble Kalman filter and the Maximum Entropy filter in its ability to track the ball. We will do so by varying the following parameters: (1) the observation error, σ_o^2 ; (2) the diffusion coefficient, κ ; and, (3) the number of particles, N .

For each choice of κ , we will generate the true trajectory for the ball by appropriately stitching together two runs. For each run, the ball is allowed to propagate under the stochastic differential equation (4.1) from an initial position of zero (see figure 4-4). We justify this procedure by noting that, when switching from one well to the other, the ball must cross the zero line. (Alternatively, we could generate the trajectory by allowing the ball to diffuse naturally from one well to the other. This,

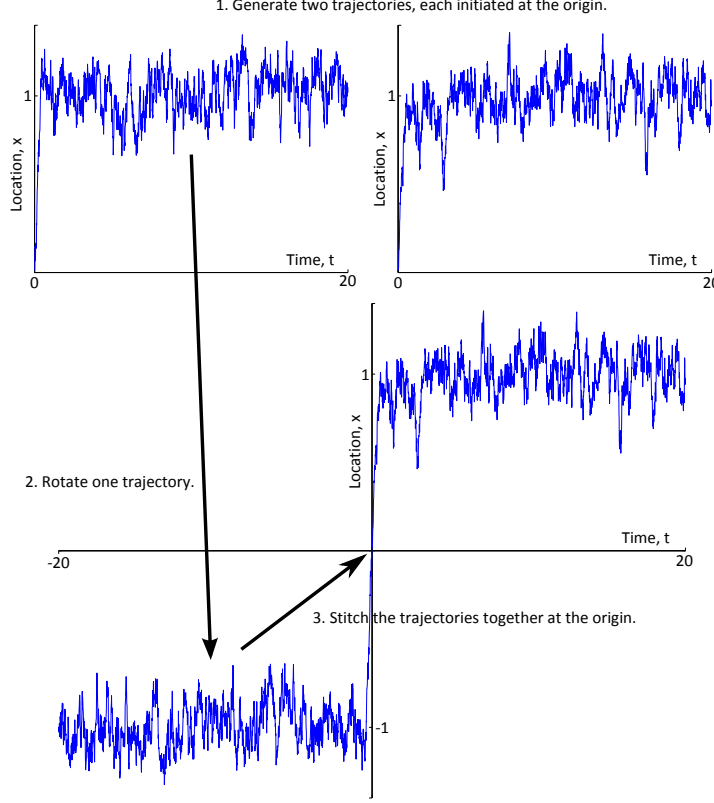


Figure 4-4: The true trajectory for the ball is obtained by appropriately stitching together two runs, each initiated at $x = 0$.

however, would certainly be time consuming and has therefore been avoided). This simulated true trajectory is held constant as we vary the other parameters.

We implement the governing stochastic differential equation (4.1) computationally using the Euler-Maruyama scheme (see e.g. Higham (2001)):

$$x_{k+1} = x_k + f(x_k) \Delta t + \kappa \gamma \sqrt{\Delta t}, \quad (4.9)$$

where γ is drawn from a normal distribution with zero mean, unit standard deviation and white in time.

For each choice of κ , we computationally derive the optimal parameters of the Gaussian mixture approximation for the climatological distribution. We do so by initially placing 500,000 particles at $x = -1$ and 500,000 at $x = 1$, allowing these to run 10,000 time steps at $\Delta t = 0.01$, at which point we use the EM algorithm to approximate the appropriate parameters.

In order to allow for a fair comparison, all filters are initiated with the same particles, generated from the Gaussian mixture approximation for the climatological distribution. Furthermore, the stochastic forcing is held constant across the three filters. For each observation error, σ_o^2 , we further hold the observations constant as we vary the number of particles, N .

We will provide results for the following range of parameters:

- $\kappa = \{0.4, 0.5\}$
- $\sigma_o^2 = \{0.025, 0.050, 0.100\}$
- $N = \{100, 1000, 10000\}$

4.3 Results and Analysis

The following plot, figure 4-5, serves as legend for each of the resulting figures, 4-6 - 4-11. Superimposed onto the true solution we show the temporal mean and standard deviation envelope for each of three filters, as well as the obtained measurements. In figures 4-6 - 4-8, we investigate the results for a diffusion coefficient, κ , of 0.40; in figures 4-9 - 4-11, the diffusion coefficient is increased to a value of $\kappa = 0.50$. We do so by varying the measurement uncertainty, σ_o^2 , as well as the number of particles, N , as described above.

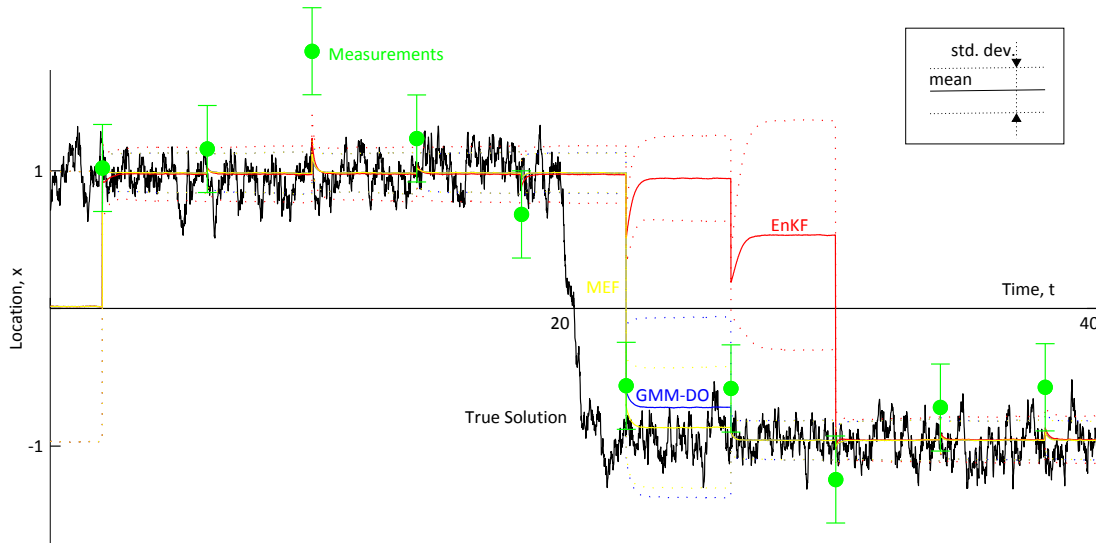


Figure 4-5: Legend for the Double Well Diffusion Experiment.

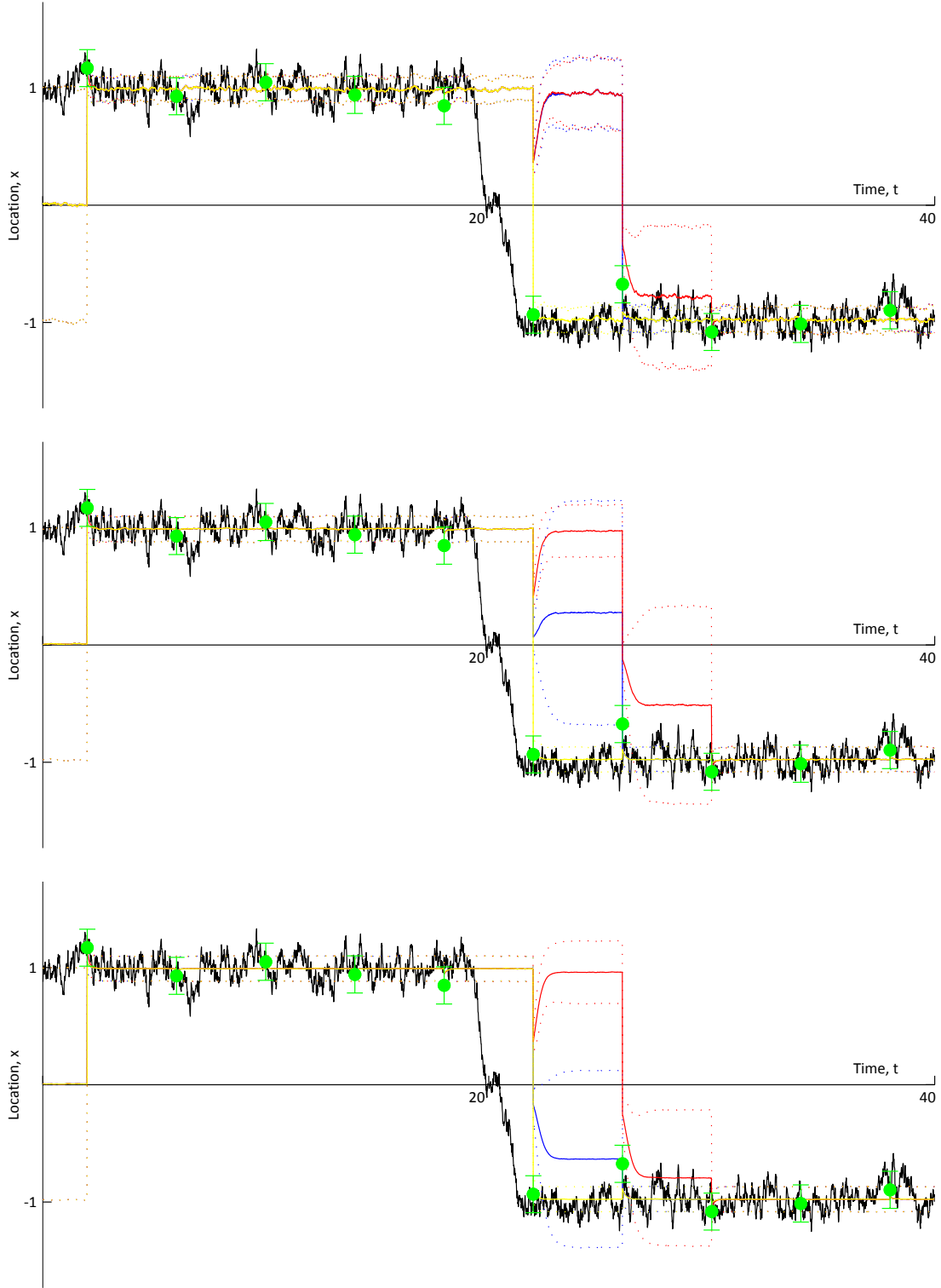


Figure 4-6: Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.4$; $\sigma_o^2 = 0.025$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.

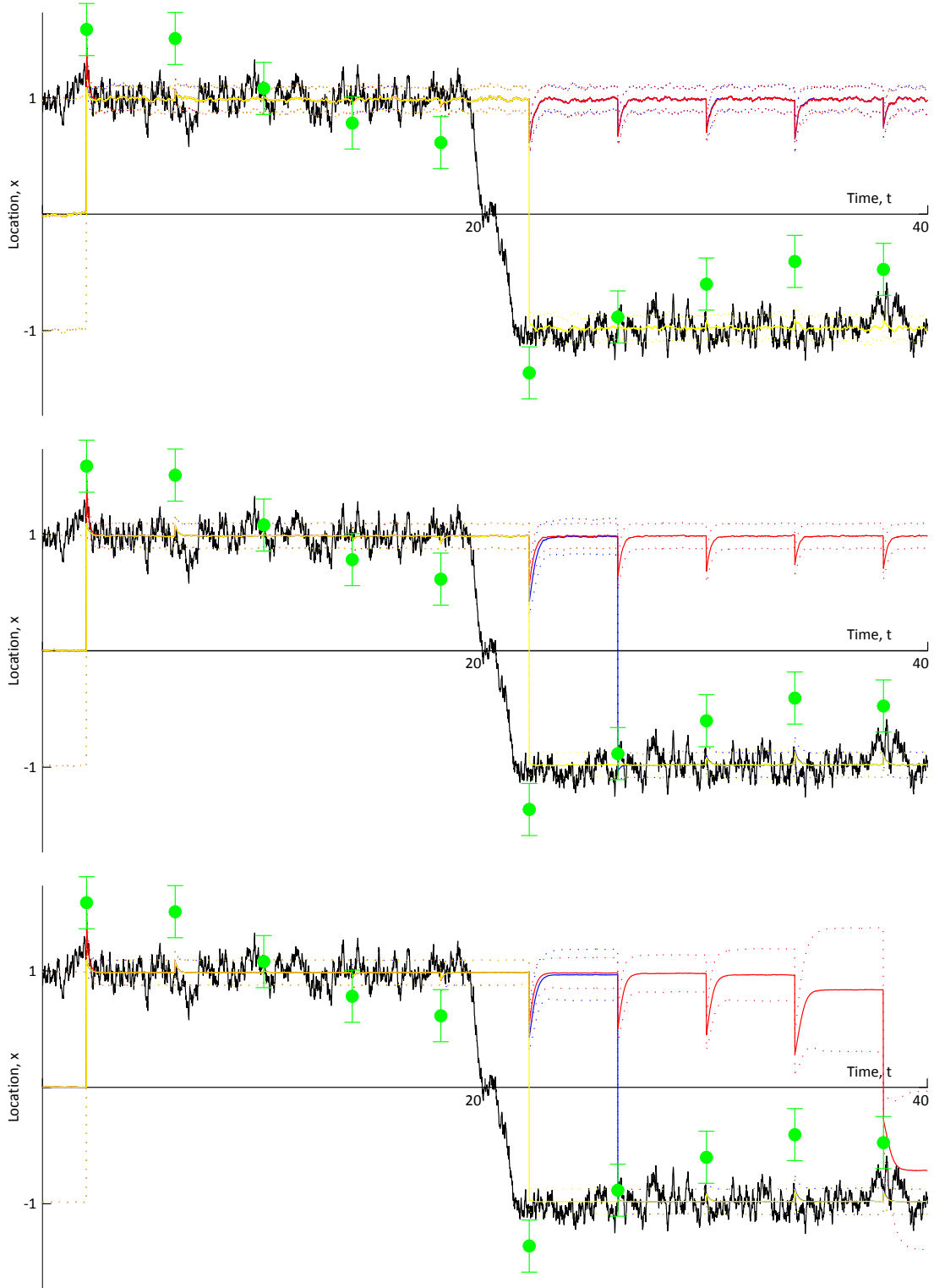


Figure 4-7: Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.4$; $\sigma_o^2 = 0.050$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.

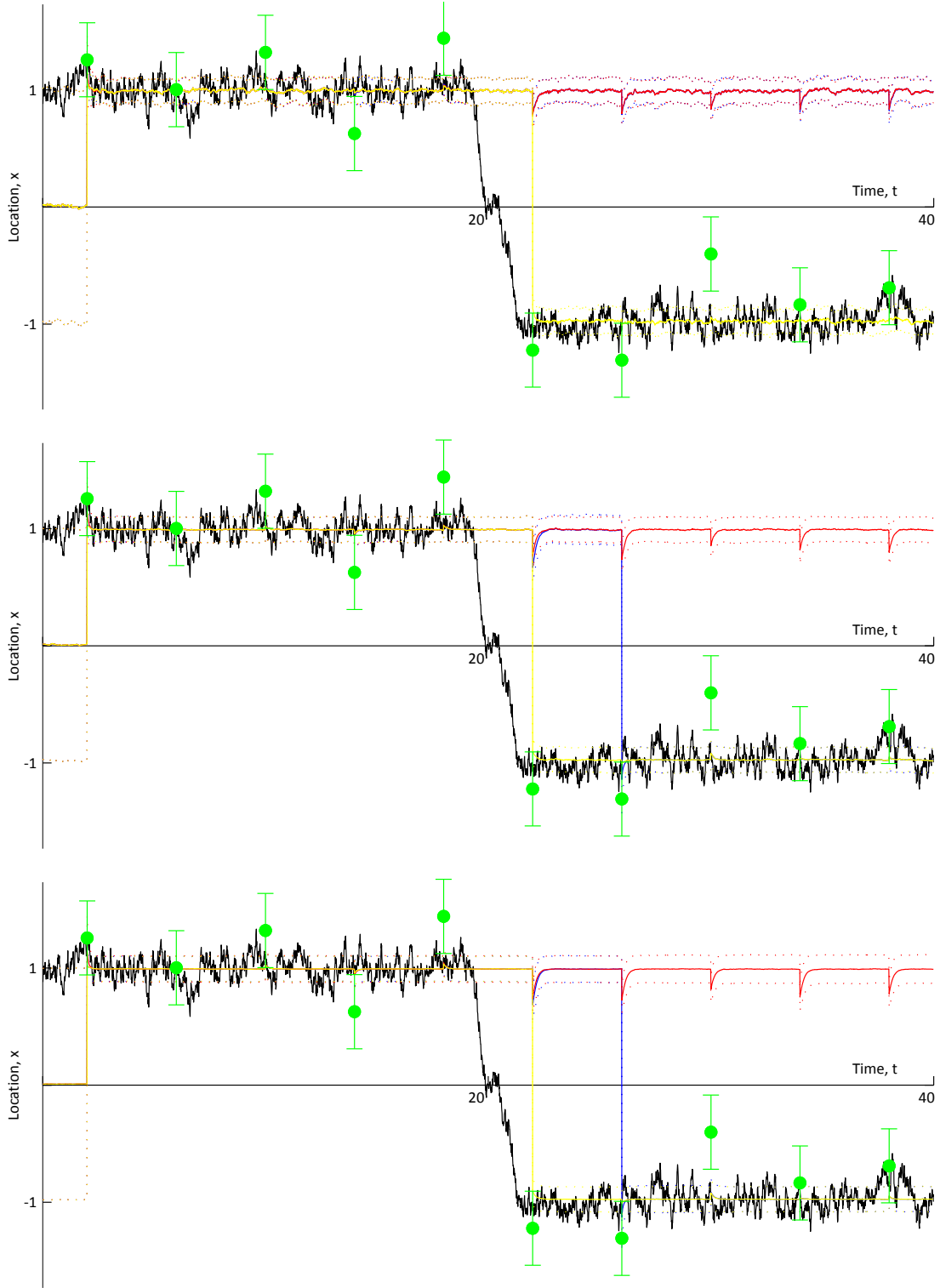


Figure 4-8: Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.4$; $\sigma_o^2 = 0.100$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.

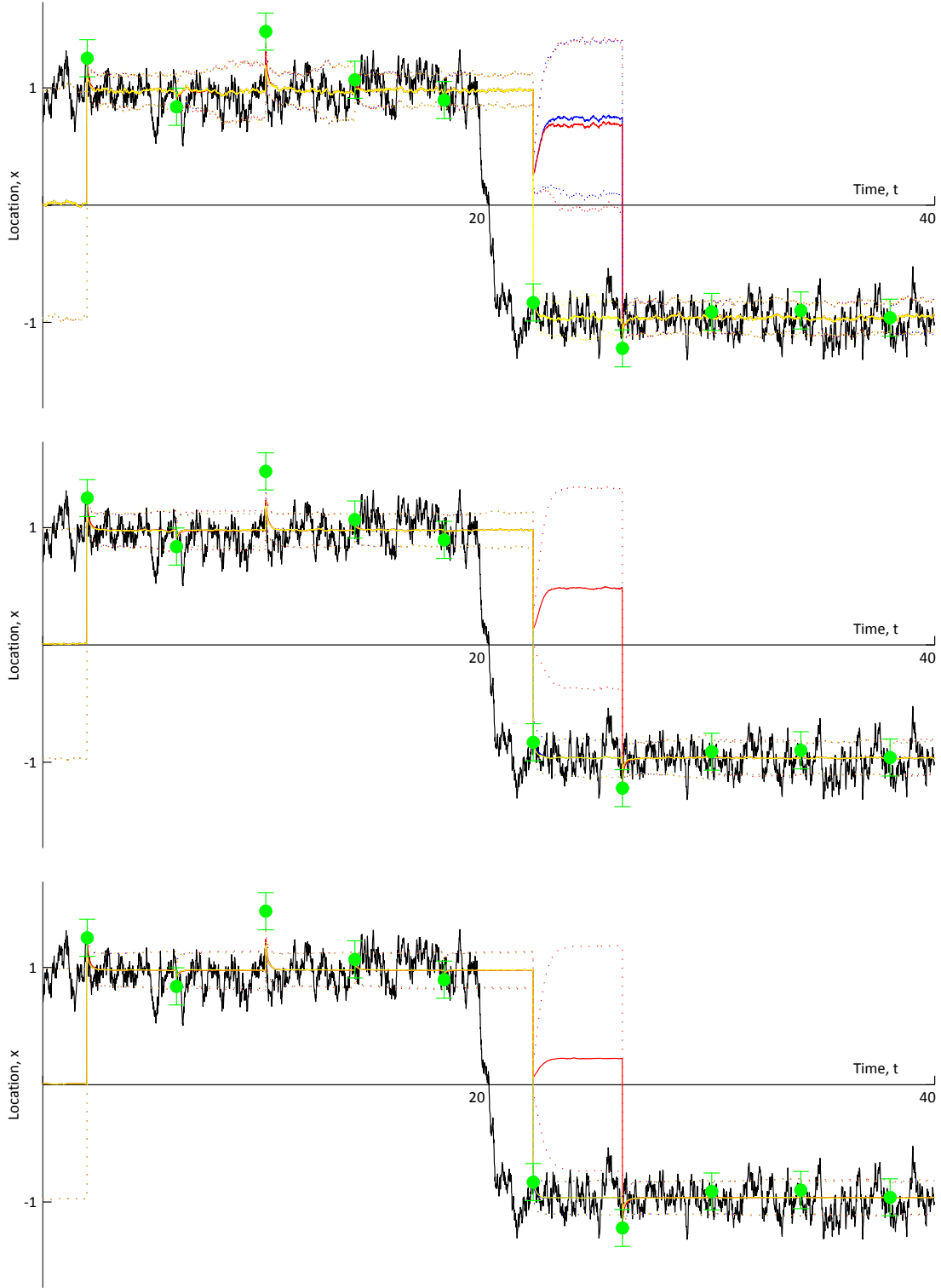


Figure 4-9: Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.5$; $\sigma_o^2 = 0.025$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.

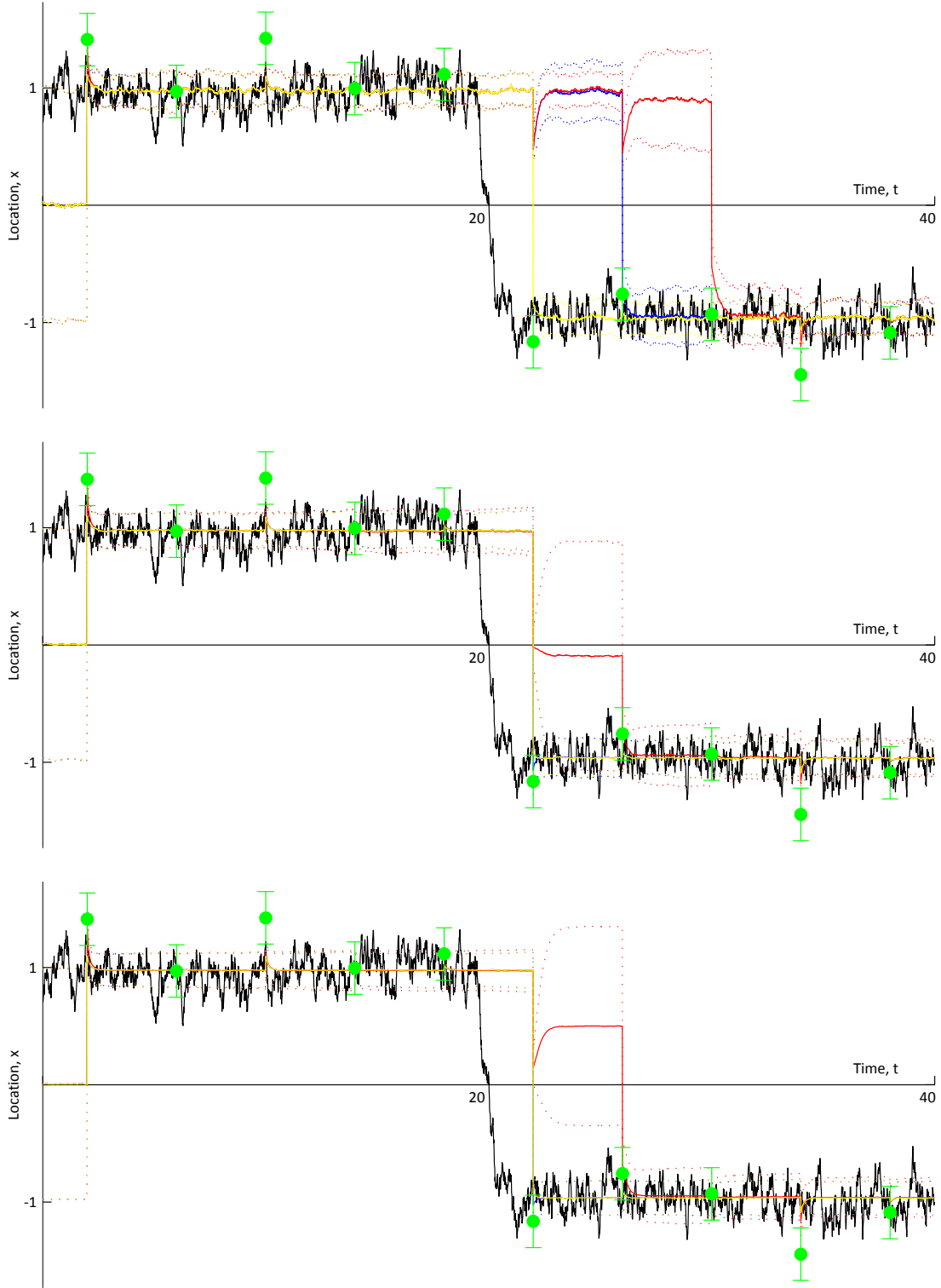


Figure 4-10: Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.5$; $\sigma_o^2 = 0.050$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.

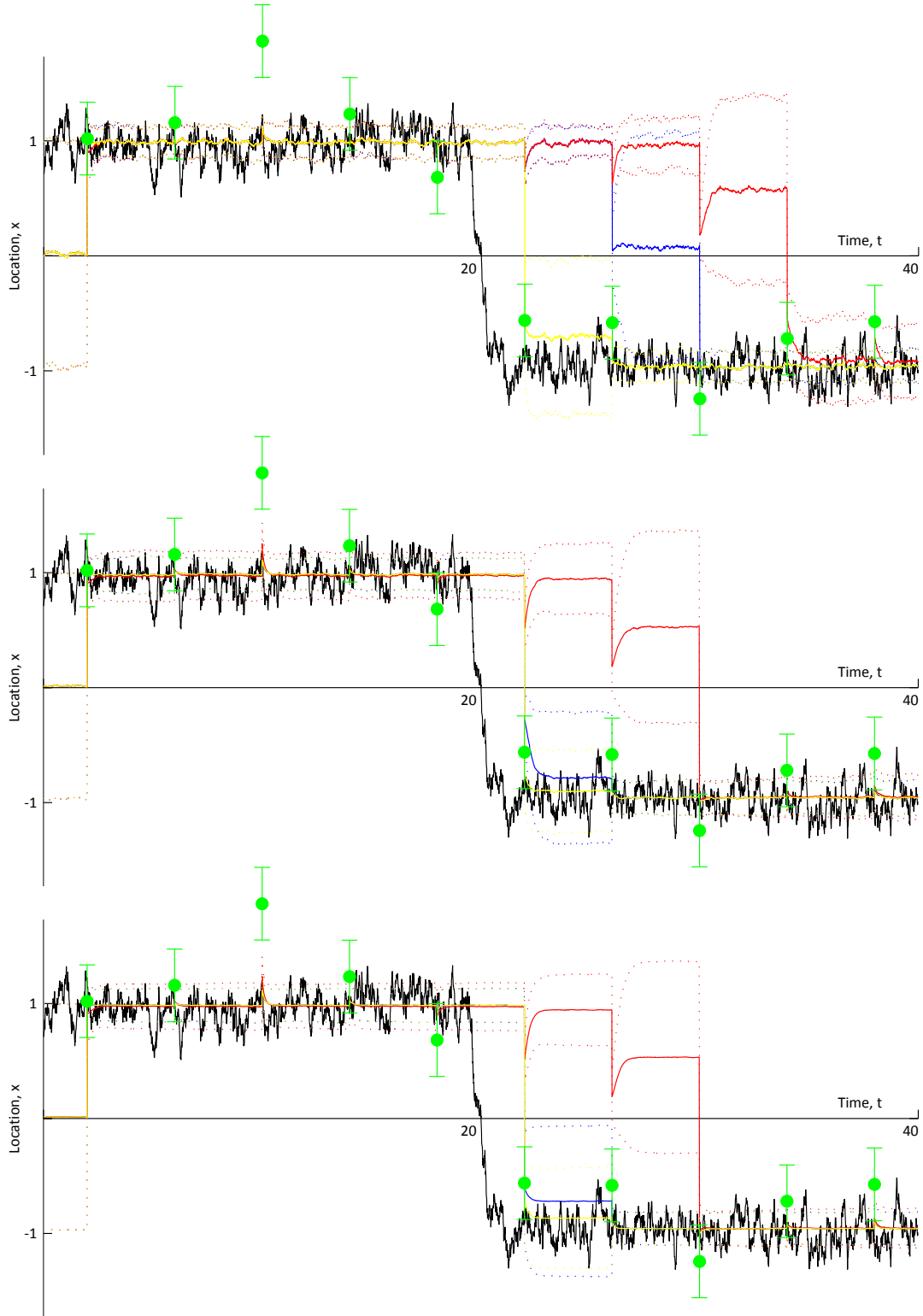


Figure 4-11: Results for MEF, GMM-DO, and EnKF with parameters $\kappa = 0.5$; $\sigma_o^2 = 0.100$; and (top) 100 particles; (middle) 1000 particles; and (bottom) 10000 particles.

On average, the GMM-DO filter provides a substantial improvement over the Ensemble Kalman filter in its ability to capture the transition of the ball from one well to the other. In fact, already with a small number of particles does the performance of the GMM-DO filter become comparable to the Maximum Entropy filter (which, for the given test case, is close to optimal, namely Bayes' filter).

The Maximum Entropy filter derives its success from accurately imposing relevant structure through the known, predetermined climatological distribution, modified only by the first two moments of the particles. The GMM-DO filter, on the other hand, attempts to infer this structure in real time by use of the EM algorithm. As a consequence, for the particular case of the Double Well Diffusion Experiment, in order for the GMM-DO filter to assign finite probability to any well, this well must have been 'explored' by the ensemble of particles at the time of fitting of the Gaussian mixture model. In agreement with our results, this exploration is enhanced as we either increase the number of particles, N , or the diffusion coefficient, κ . For instance, if, for the case of $N = 10,000$ and $\sigma_o^2 = 0.050$, we compare the results for $\kappa = 0.4$ and $\kappa = 0.5$ (i.e. the bottom panels of figures 4-7 and 4-10), we notice that for the former, two measurements are required for the GMM-DO filter to infer the transition of the ball: the first to force particles into the opposite well and the second to consequently assign this well sufficient probability. For the latter case, this forcing of particles from one well to the other occurs naturally due to the larger diffusion coefficient (and is thus recognized when fitting the Gaussian mixture model). Interestingly, these results are nearly independent of the observation error – an obvious strength of the GMM-DO filter.

As opposed to both the Maximum Entropy filter and the GMM-DO filter, the Ensemble Kalman filter consistently transitions from one well to the other over a number of assimilation steps (if and when it transitions), as evidenced, for instance, by the bottom panel in figure 4-7. Upon receiving measurements from the well opposite to that in which its probability is placed, particles are gradually forced across. The strength of forcing is derived by weighing of the prior variance against the observation noise, as defined by the Kalman gain matrix. Therefore, for observations

of relatively large variance, the Kalman gain matrix takes on a small value, causing the Ensemble Kalman filter to perform poorly, as evidenced for instance by figure 4-8. With reference to the same figure, we suitably note the comparably superior performance of both the Maximum Entropy filter and the GMM-DO filter, especially when the data error variance increases, thus emphasizing their enhanced abilities to extract information from noisy measurements.

We visualize the prior analysis by a trio of figures, 4-12 - 4-14, examining in detail the prior and posterior distributions (and their respective particle representations) assigned by each of the three filters for the case of $N = 1,000$, $\sigma_o^2 = 0.100$ and $\kappa = 0.5$ (i.e. the middle panel of figure 4-11). We center the analysis on the observation immediately prior to the true transition of the ball, as well as the two following.

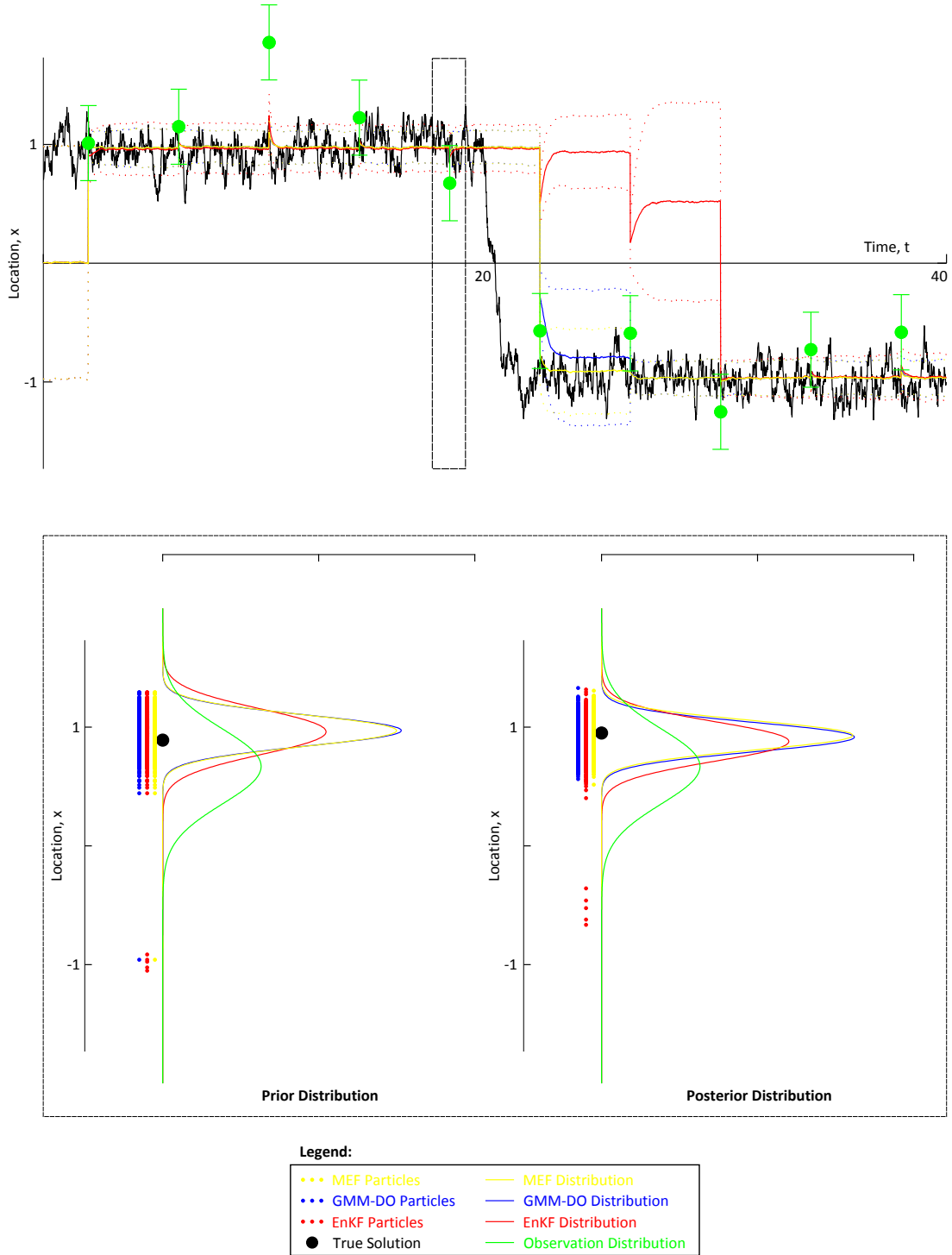


Figure 4-12: Analysis of the prior and posterior distributions (and particle representations) by each of the three filters (EnKF, GMM-DO and MEF) for the particular case of $N = 1,000$ and $\kappa = 0.5$, centered on the observation immediately prior to the true transition of the ball.

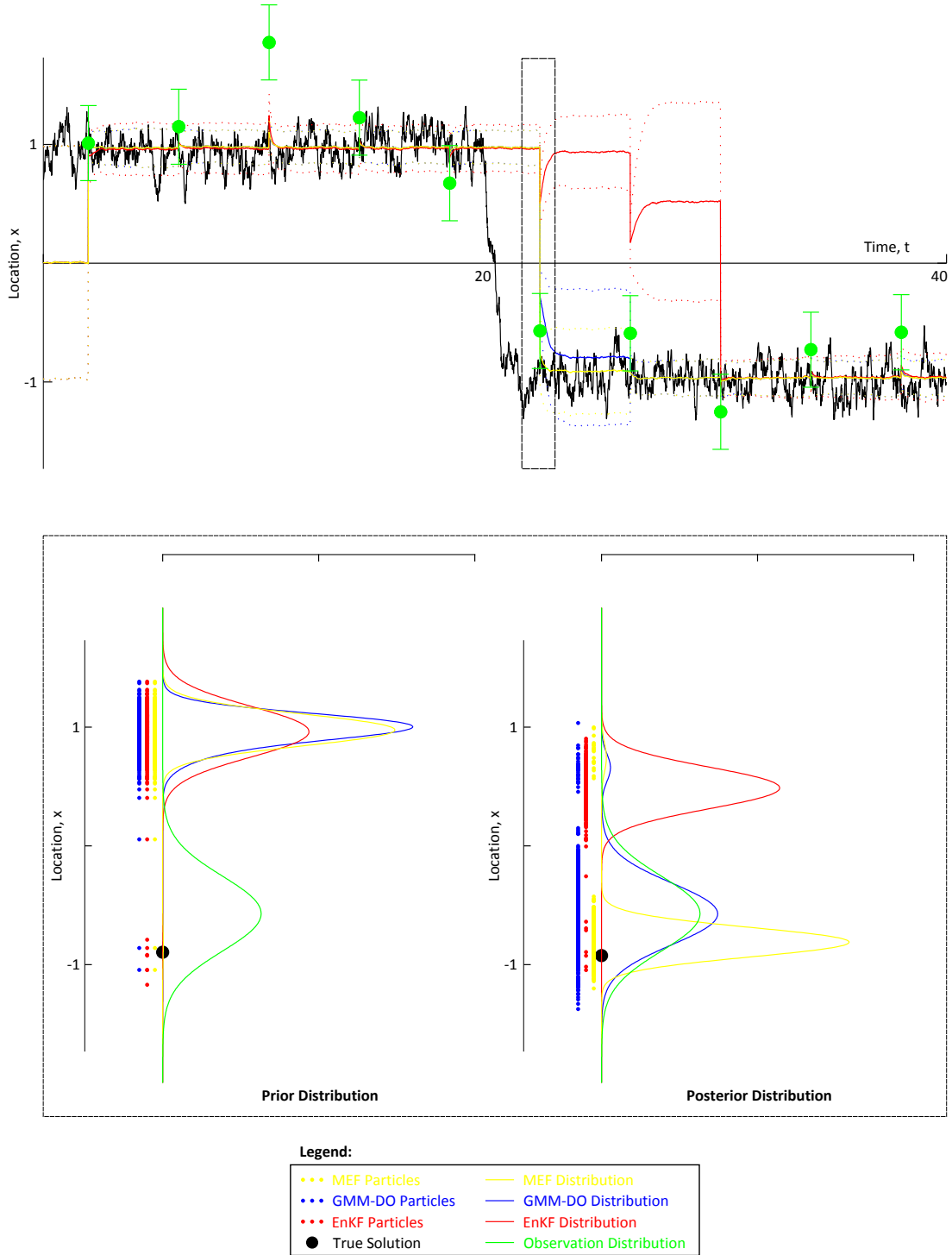


Figure 4-13: Analysis of the prior and posterior distributions (and particle representations) by each of the three filters (EnKF, GMM-DO and MEF) for the particular case of $N = 1,000$ and $\kappa = 0.5$, centered on the observation immediately following the true transition of the ball.

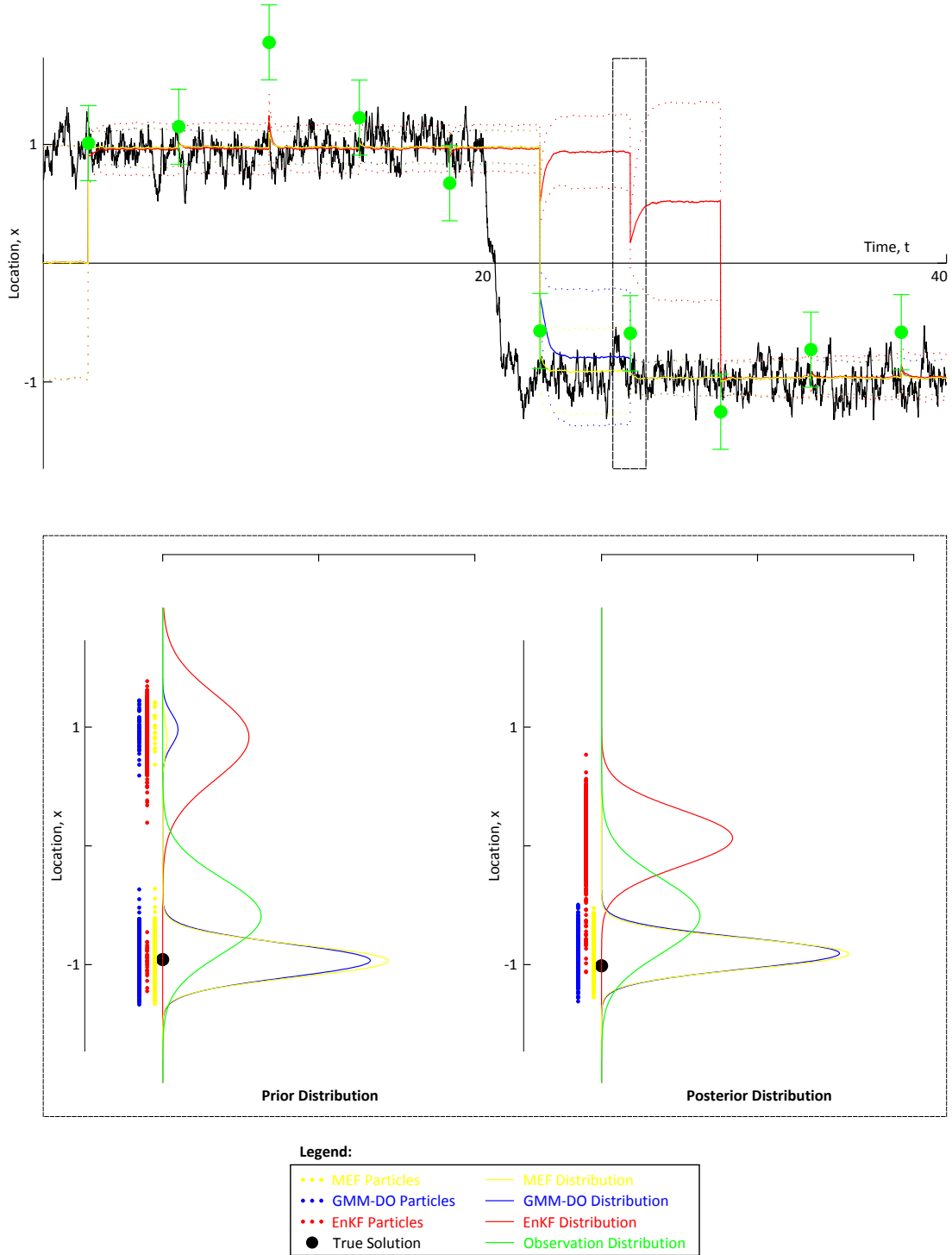


Figure 4-14: Analysis of the prior and posterior distributions (and particle representations) by each of the three filters (EnKF, GMM-DO and MEF) for the particular case of $N = 1,000$ and $\kappa = 0.5$, centered on the second observation following the true transition of the ball.

In figure 4-12, the ball has not yet transitioned and all three filters assign proba-

bility to the correct well, both prior and posterior to the recorded measurement. In figure 4-13, following the transition of the ball from one well to the other, both the two (blue) particles located in the well centered on $x = -1$ (into which the ball has transitioned) caused the GMM-DO filter to assign sufficient probability to this well such that, posterior to the assimilation of the measurement, it places the majority of its probability to the true well. Such is equally the case for the MEF. The EnKF, on the other hand, assigned insignificant prior probability to the correct well, such that the measurement had little influence. In the final figure, 4-14, while the GMM-DO filter and MEF have captured the true location of the ball, the EnKF gradually shifts its probability into the correct well. Not until the next measurement (not depicted in detail) does that EnKF correctly capture the true location of the ball.

4.4 Conclusion

For the range of parameter values investigated in the Double Well Diffusion Experiment, the GMM-DO filter has been shown to outperform the Ensemble Kalman filter in its ability to capture the transition of the ball from one well to the other. Moreover, for only a moderate number of particles is the performance of the GMM-DO filter comparable to that of the Maximum Entropy filter, the latter of which is particularly well-suited to the given test case. As we further increase the number of particles, we expect the GMM-DO filter to converge to the Bayes filter. This claim is supported by the results obtained for the case of $N = 10,000$ particles. We also note that as the observation error variance increases, the performance of the EnKF deteriorates much more rapidly than the GMM-DO filter. One can thus expect that if either measurement model errors are large or measurements are sparse, the GMM-DO filter will outperform the EnKF and other Gaussian updates. These two measurement situations are very common in ocean/atmospheric flows. For example, even if sensor errors are small, the multiscale properties of the flows and geometry are such that errors of representativeness can especially be large and so dominate the measurement model errors.

The Maximum Entropy filter shares a number of similarities with the GMM-DO filter, particularly in its use of Gaussian mixture models for approximating the prior distribution. While the Maximum Entropy filter enforces its structure through the imposed climatological distribution (modified only by the moments of the particles), the GMM-DO filter attempts to infer this structure in real time by use of the EM algorithm. As a consequence, the GMM-DO filter is substantially more generic, needing no specification of any climatological distribution. In any event, for cases in which the climatological distribution is known or may fairly well be approximated, it is not unreasonable to expect that the two schemes may be merged in a beneficial manner. This remains to be investigated, however.

The bimodal structure present in the Double Well Diffusion Experiment is reminiscent of that which arises in the dynamics of the Kuroshio current (Sekine (1990),

Miller et al. (2004)). As a consequence, many of the conclusions drawn from the previous results may reasonably be extrapolated to that of larger systems with more complicated dynamics. This is to be explored in the following chapter.

Chapter 5

Application 2: Sudden Expansion Fluid Flow

In this chapter, we examine the performance of the GMM-DO filter in a more realistic setting, namely a two-dimensional sudden expansion fluid flow. Such flows have been of considerable interest in the past (see e.g. the papers by Durst et al. (1973), Cherdron et al. (1978) and Fearn et al. (1990)) and continue to attract attention in the literature. Due to the breaking of symmetries with increasing Reynolds number and the consequent development of bimodal statistics, it provides a test case particularly well-suited to the evaluation of our proposed data assimilation scheme. We also chose this example because it corresponds to a uniform barotropic jet (flow 2D in the horizontal) exiting a Strait or an estuary, in the case of a width that is small enough for the effects of the earth rotation (Coriolis acceleration) to be neglected. Such strait or estuary flows occur in the ocean, generally leading to meanders as the jet exits the constriction. A generalization of such jets would include Coriolis and barolinic (3D) effects, which could be considered in future work.

After providing a general introduction to the test case, we will describe the numerical method used to simulate the flow. We evaluate the performance of the GMM-DO filter by application of an ‘identical twin experiment’ (Bengtsson et al., 1981): we generate a simulated true solution over a suitable time frame at a Reynolds number that allows for interesting dynamics. Based on sparse and intermittent measurements

of velocities, we ultimately wish to reconstruct the true solution with knowledge only of initial uncertainties. Specifically, we compare the GMM-DO filter against a modified ESSE scheme A. To measure and compare the accuracy of the estimates, we employ the temporal root mean square difference between the true solution and their respective mean fields. We provide detailed results at each of the assimilation times and conclude with an in-depth analysis of their performances.

5.1 Introduction

It is a well known fact that flows, symmetric both in initial conditions and geometry, may develop asymmetries with increasing Reynolds numbers, Re ; a phenomenon sometimes referred to as the "Coanda" effect (Fearn et al., 1990). A classical example of such is the development of the *von Karman vortex street* in the wake of a blunt body placed in a uniform flow (Kundu and Cohen, 2008). In this chapter we will focus on the so-called sudden expansion fluid flow which exhibits similar behavior.

The sudden expansion fluid flow, here limited to two dimensions, is perhaps most easily understood visually. We refer the reader to figure 5-1.

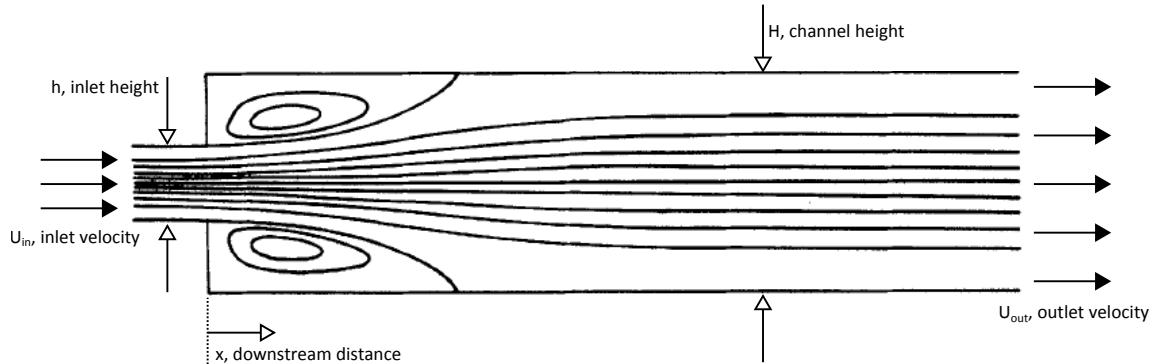


Figure 5-1: Setup of the sudden expansion test case (Fearn et al., 1990).

A developed, symmetric flow of maximum inlet velocity U_{max} in a channel of height h expands into a larger channel of height H , denoting H/h as the expansion ratio. Depending on the Reynolds number,

$$Re = \frac{(h/2)U_{max}}{\nu}, \quad (5.1)$$

where ν is the kinematic viscosity, a number of phenomena may occur. Experimental results show that for low Reynolds numbers the flow is symmetric about the channel centerline, with circulation regions formed at the corners of the expansion (Durst et al., 1973). This is the case depicted in figure 5-1, where the flow is described by streamlines. As the Reynolds number is increased, instabilities develop giving rise to steady, asymmetric flows. Cherdron et al. (1978) experimentally determined the critical Reynolds numbers at which these instabilities arise; they did so as a function of both expansion and aspect ratio, the latter referring to the ratio of channel width to channel height (appropriate only for three-dimensional flows). Their findings are provided in figure 5-2.

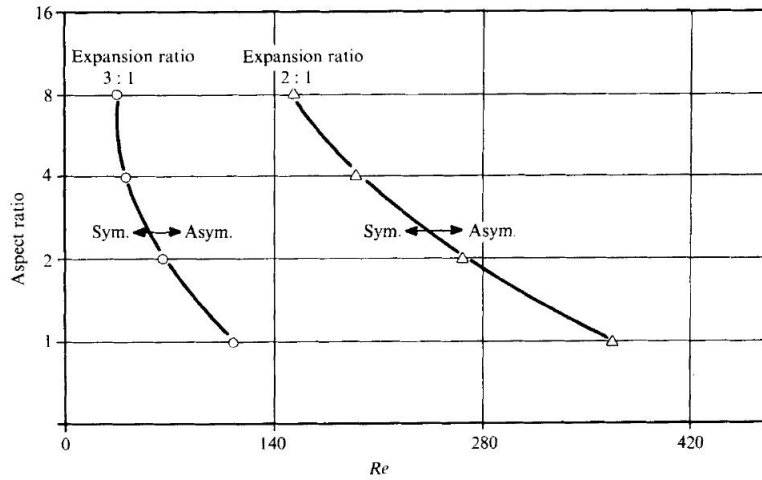


Figure 5-2: Boundaries of symmetric and asymmetric flow as a function of aspect ratio, expansion ratio and Reynolds number (Cherdron et al., 1978).

These findings were partially verified by the numerical stability analysis of Fearn et al. (1990) for 2D flows. Examples of experimental results that depict the aforementioned symmetric and asymmetric flows may be seen in figure 5-3.

In this chapter, we will be considering the case of an intermediate Reynolds number for which the 2D flow develops asymmetries, yet remains steady and laminar. Specifically, we will be working with an expansion ratio of 3 and $Re \approx 200$, for which figure 5-2 confirms the onset of asymmetries (for the case of 3D flows). We expect results similar to that predicted numerically and verified experimentally by Fearn et al. (1990) for the case of $Re = 140$, as shown in figures 5-4 and 5-5.

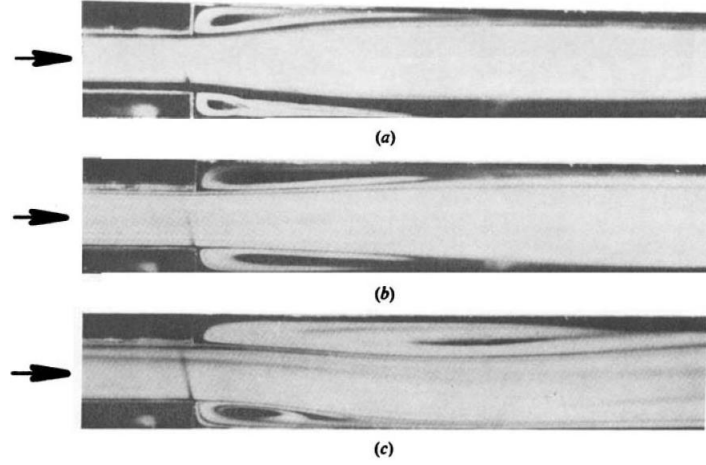


Figure 5-3: Flow patterns at different Reynolds numbers for an aspect ratio of 8 and an expansion ratio of 2. (a) $Re = 110$. (b) $Re = 150$. (c) $Re = 500$. (Cherdron et al., 1978).

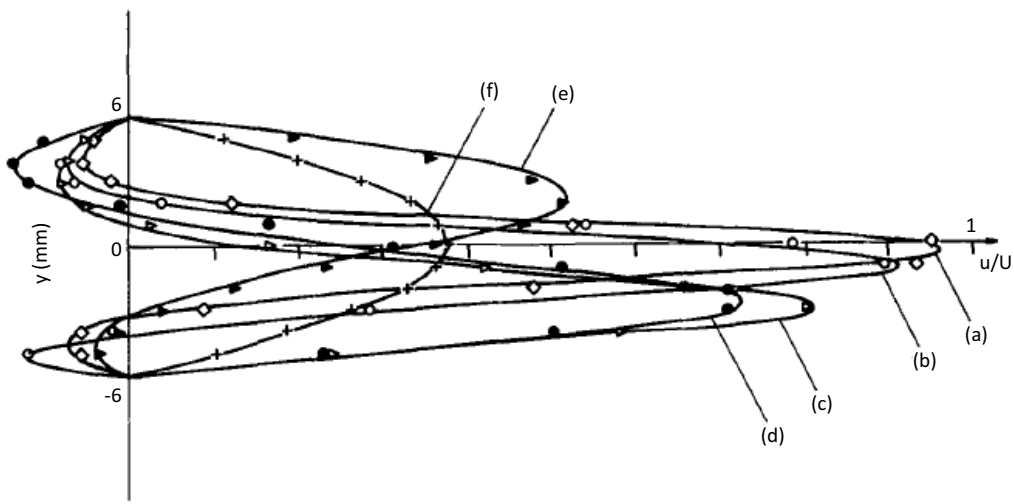


Figure 5-4: Numerical and experimental velocity plots at $Re = 140$. The numerically calculated profiles are shown as continuous curves. (a) $x/H = 1.25$; (b) $x/H = 2.5$; (c) $x/H = 5$; (d) $x/H = 10$; (e) $x/H = 20$; (f) $x/H = 40$. (Fearn et al., 1990).

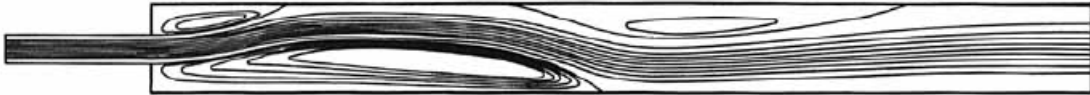


Figure 5-5: Calculated streamlines at $Re = 140$. (Fearn et al., 1990).

The symmetric inlet velocity initially breaks to one side of the centerline, visualized, in particular, by curves (c) and (d) in figure 5-4. Further downstream, at

$x/H \approx 20$, a second region of circulation forces the flow to the opposite side, depicted by curve (e), before eventually restoring its initial symmetry (see curve (f)). The full picture is given in figure 5-5. Clearly, the favored direction of the flow depends sensitively on perturbations in the initial conditions, thus giving rise to bimodal statistics.

5.2 Procedure

From here on, all figures depicting the fluid flow will be described by streamlines overlaid on a color-plot in which the color denotes the magnitude of velocity.

Physical Setup

In figure 5-6, we present the setup for our test case:

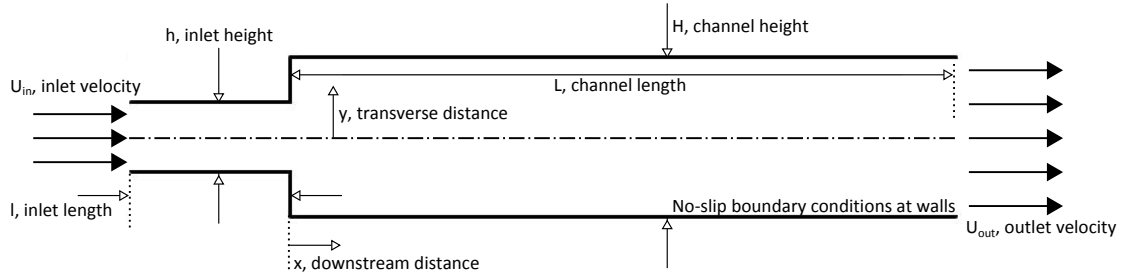


Figure 5-6: Sudden Expansion Test Setup.

Placing variables in a non-dimensional form, we let $h = \frac{1}{3}$; $l = 4$; $H = 1$; and $L = 16$. We further impose a *uniform* inlet velocity of $U_{in} = 1$. By conservation of mass and using the steady, fully developed Navier-Stokes equations, we predict the following velocity profile at $x = 0$ (see Appendix):

$$U(x = 0, y) = \frac{2}{h^3} \left(\frac{h^2}{4} - y^2 \right). \quad (5.2)$$

We therefore expect a maximum inlet velocity of

$$U_{max} = U(x = 0, y = 0) = \frac{1}{2h} = \frac{3}{2}, \quad (5.3)$$

corresponding to a Reynolds number of

$$Re = \frac{(h/2)U_{max}}{\nu} = \frac{\frac{1}{6}\frac{3}{2}}{10^{-3}} = 250. \quad (5.4)$$

This confirms our previous expectation regarding the nature of the flow, namely that it will exhibit steady asymmetries (we again refer the reader to figure 5-2).

Initialization of DO decomposition

(1) **Mean Field, $\bar{\mathbf{x}}$:** the x-component of the mean field velocity is everywhere 1 in the inlet and $\frac{1}{3}$ at any point in the channel, in accordance with continuity; the y-component of the mean field is initially zero everywhere. See figure 5-7.

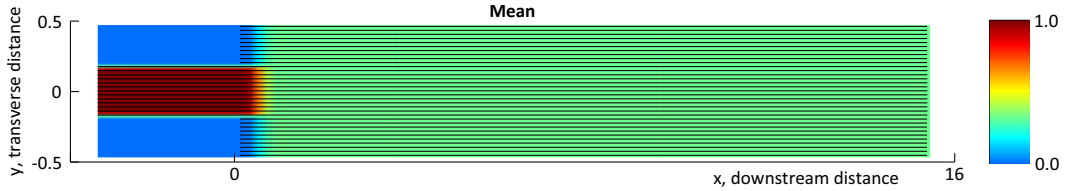


Figure 5-7: Initial mean field of the DO decomposition.

(2) **Orthonormal modes, $\tilde{\mathbf{x}}_i$:** Following Sapsis and Lermusiaux (2009), the orthonormal modes are generated by retaining the dominant eigenvectors of the correlation operator $\mathbf{C}(\cdot, \cdot)$, defined by

$$\mathbf{C}((x_1, y_1), (x_2, y_2)) = \mathcal{M}((x_1, y_1), (x_2, y_2)) C(r), \quad (5.5)$$

where r is the Euclidean distance between points (x_1, y_1) and (x_2, y_2) , and $\mathcal{M}(\cdot, \cdot)$ is a mollifier function globally taking the value 1 except at solid boundaries, at which it vanishes smoothly. We let $C(r)$ take the form

$$C(r) = (1 + 5r + \frac{5^2 r^2}{3}) e^{-5r}. \quad (5.6)$$

We create the stochastic subspace, \mathcal{X} , by retaining the twenty most dominant eigenvectors (i.e. we let $s = 20$); we hold this number constant throughout the simulation.

In the analysis section of this chapter, we justify this choice in the sense that it adequately captures the inherent uncertainties. Specifically, we studied the subspace and assimilation results with a varying number of modes, concluding that a subspace of size 20 was sufficient. In a future work, we may let s be time-variable and governed by the system dynamics, as described in Sapsis and Lermusiaux (2010). A selection of the initial modes is shown in figure 5-8.

(3) **Ensemble Members, $\{\phi\}$:** We generate 10,000 ensemble members, ϕ_i , from a zero mean, multivariate Gaussian distribution with diagonal covariance matrix. We thus initialize the modes as being statistically uncoupled with marginal variances proportional to the eigenvalues of the previously described correlation operator.

Observations

We make a total of three sets of measurements of both u- and v-velocities of the true solution at times $T_{obs} = \{50, 70, 90\}$ at the locations indicated in figure 5-9. The measurements are independent of each other and are made with an observation noise distributed according to a zero-mean Gaussian with variance $\sigma_{obs}^2 = 0.1$. Other data errors were investigated, but will not be presented here.

Generating the True Solution

We initialize the true solution by selecting an arbitrary ensemble member generated according to the aforementioned initialization scheme, restricted, however, to the five most dominant modes. Since the true solution is generated from the same statistics as the one imposed, we ensure that our initial statistics capture the true solution.

The true solution is propagated deterministically forward in time under the governing equation (i.e. the Navier-Stokes equations) for a total time of $T = 100$, after which the simulation will have settled into its steady state. In the following section, we describe the numerical implementation of the Navier-Stokes equations.

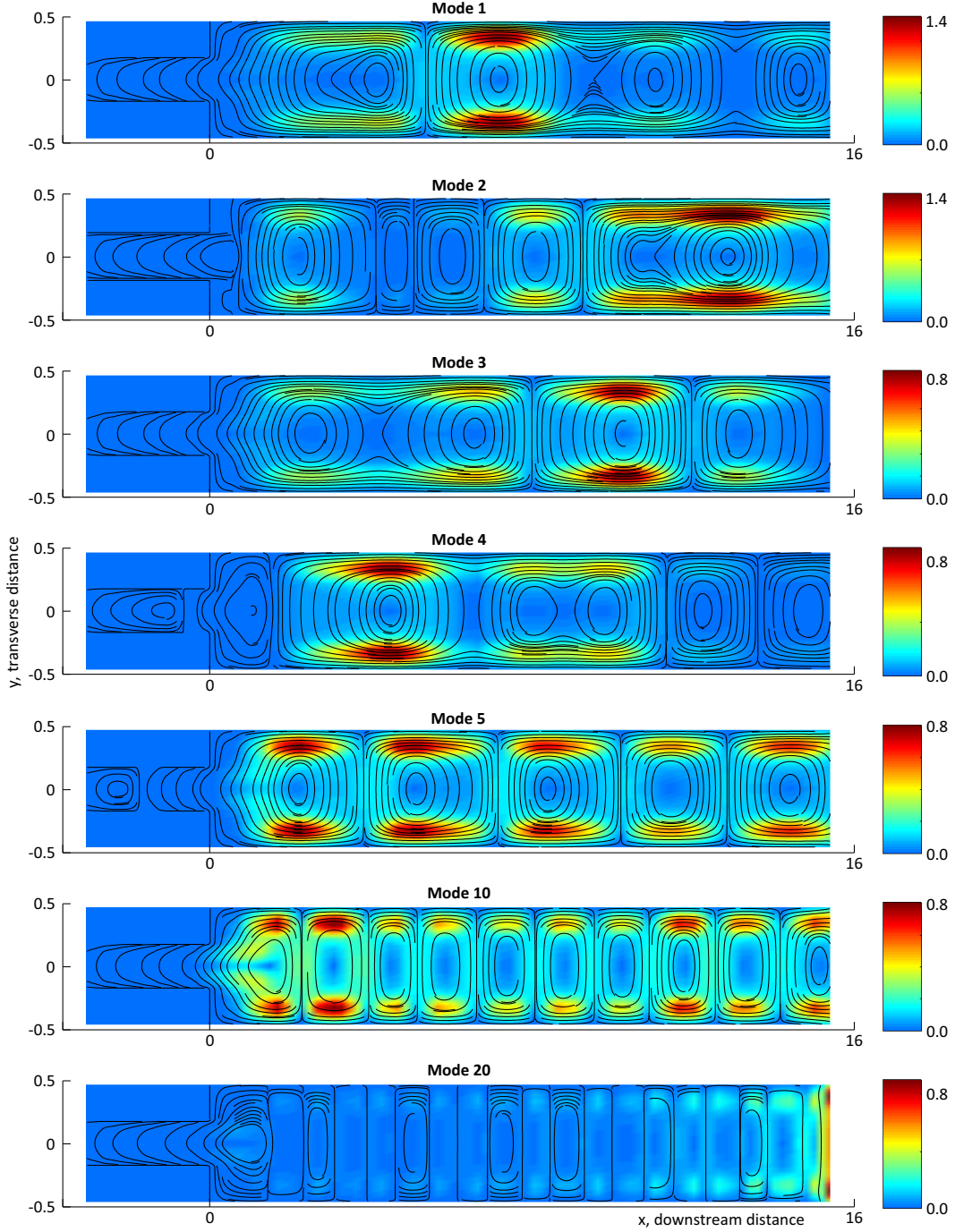


Figure 5-8: Selection of initial modes of the DO decomposition.

5.3 Numerical Method

Based on Ueckermann et al. (2011), we solve the Navier-Stokes equations numerically using a flexible, modular and efficient finite volume framework implemented in

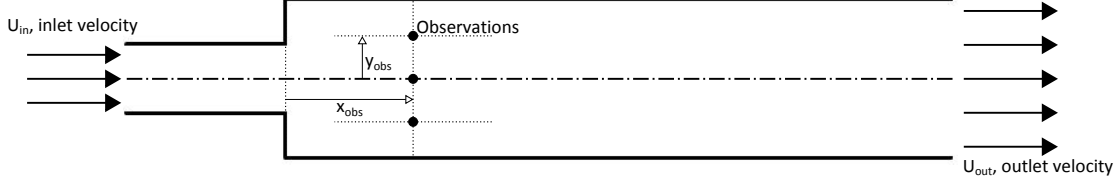


Figure 5-9: Observation Locations: $(x_{obs}, y_{obs}) = \{(4, -\frac{1}{4}), (4, 0), (4, \frac{1}{4})\}$.

MATLAB.

Geometry

The Sudden Expansion geometry is discretized on a uniform, two-dimensional, structured grid of 40 by 30 elements in the x- and y-direction, respectively. A staggered c-grid is specifically utilized to avoid spurious pressure modes.

Discretization in Space

The diffusion operator is approximated using a second order central differencing scheme, while the advection operator makes use of a Total Variation Diminishing (TVD) scheme with a monotonized central (MC) limiter (van Leer, 1977).

Discretization in Time

The time discretization uses a first-order accurate, semi-implicit Projection method, where the diffusion and pressure terms are treated implicitly, and the advection is treated explicitly (for details see Ueckermann et al. (2011)). In all cases we limit the time step in accordance with the Courant-Friedrichs-Lewy (CFL) condition.

Boundary Conditions

As depicted in figure 5-6, we assume no-slip boundary conditions at all solid boundaries, while imposing a uniform velocity of 1 across the inlet opening. At the open, outlet boundary we restrict the flow by eliminating the first x-derivative of the v-velocities and the second x-derivative of both pressure and u-velocities (i.e. $\frac{\partial v}{\partial x} = 0$, $\frac{\partial^2 u}{\partial x^2} = 0$ and $\frac{\partial^2 p}{\partial x^2} = 0$).

5.4 Results and Analysis

In what follows, we plot at every 10 time units the true solution against a condensed representation of the full DO decomposition, using notation identical to that presented in chapter 3. Specifically, we display

1. the mean field, $\bar{\mathbf{x}}$;
2. the first two modes, $\tilde{\mathbf{x}}_1$ and $\tilde{\mathbf{x}}_2$;
3. the marginal probability density functions of the stochastic coefficients ψ_1 and Φ_2 using MATLAB's 'ksdensity' function;
4. a scatter plot of the ensemble set, $\{\phi\} = \{\phi_1, \dots, \phi_N\}$, projected onto the pair of modes: $(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2)$;
5. a time history of the variances of all the stochastic coefficients, Φ_j ; and
6. a time history of the RMS error of both the GMM-DO filter and a modified ESSE scheme A, as described in our introduction. The latter refers to the GMM-DO filter with a mixture complexity of one, i.e. $M = 1$. In what follows, we give it the term "DO-ESSE Scheme A".

These plots will allow the reader to appreciate the way in which the flow develops, ultimately settling into its steady state. It will equally clarify the manner in which the DO equations evolve the state representation.

At the time of new measurements (i.e. $T_{obs} = \{50, 70, 90\}$), we expand the representation of the DO decomposition by plotting

1. the mean field, $\bar{\mathbf{x}}$;
2. the first four modes, $\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \tilde{\mathbf{x}}_3$ and $\tilde{\mathbf{x}}_4$;
3. the marginal probability density functions of the stochastic coefficients Φ_1, Φ_2, Φ_3 and Φ_4 using MATLAB's 'ksdensity' function;

4. a scatter plot of the ensemble set, $\{\phi\} = \{\phi_1, \dots, \phi_N\}$, projected onto the pairs of modes: $(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2)$, $(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_3)$, $(\tilde{\mathbf{x}}_2, \tilde{\mathbf{x}}_3)$ and $(\tilde{\mathbf{x}}_3, \tilde{\mathbf{x}}_4)$;

Superimposed onto (3) and (4), we further display the Gaussian mixture model identified as the appropriate prior distribution (as part of the GMM-DO filter procedure). For the latter, we specifically display the one-standard-deviation contours of each individual mixture.

We finally plot both the true solution and its associated observation against the prior distribution at each of the measurement locations. In the same figure, we present the appropriate posterior distributions, as arrived at using Bayes'. Finally, we once again display the posterior DO decomposition using the original, condensed representation.

With this, we proceed with the results:

$T = 0$

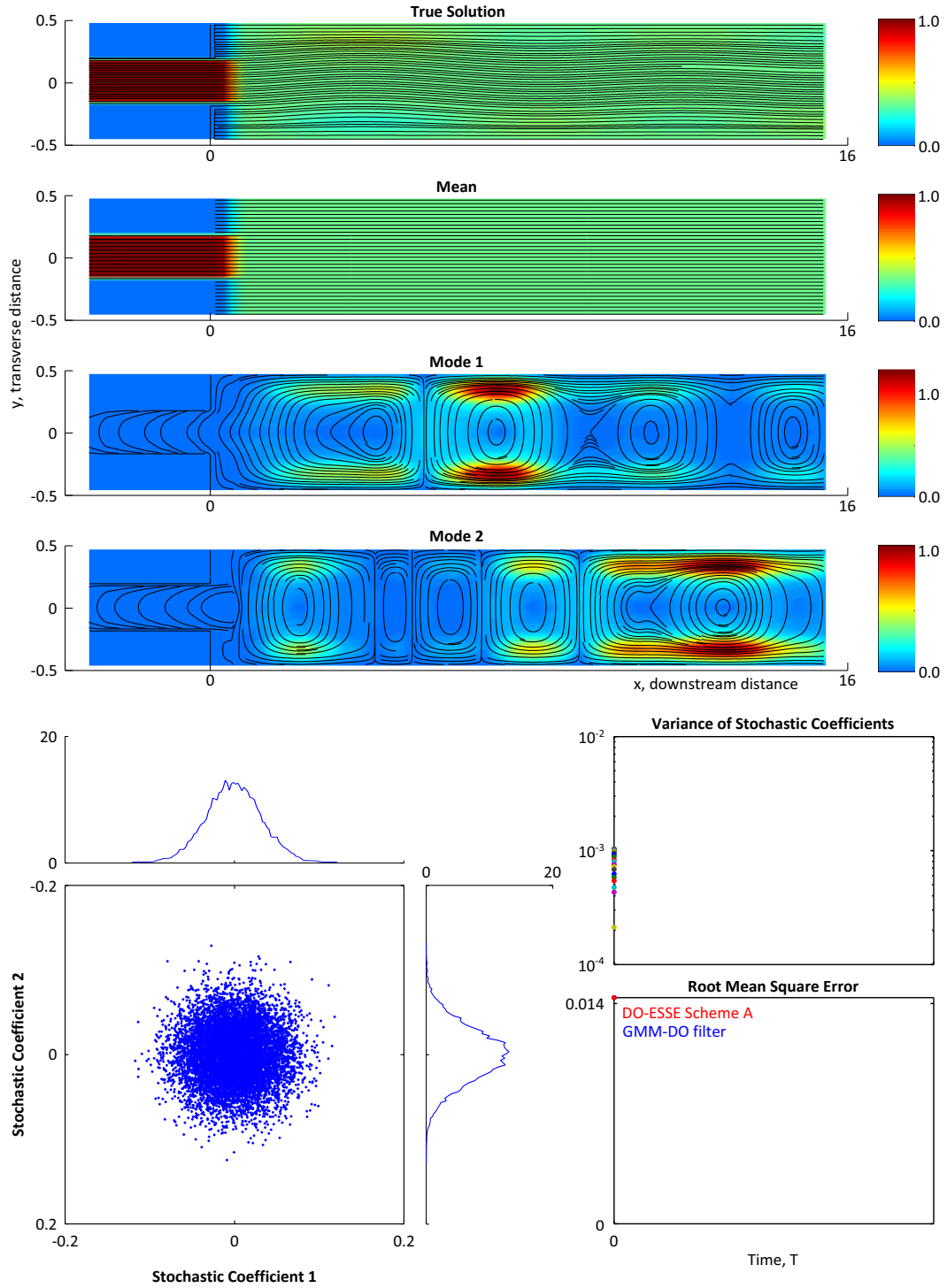


Figure 5-10: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 0$.

$T = 10$

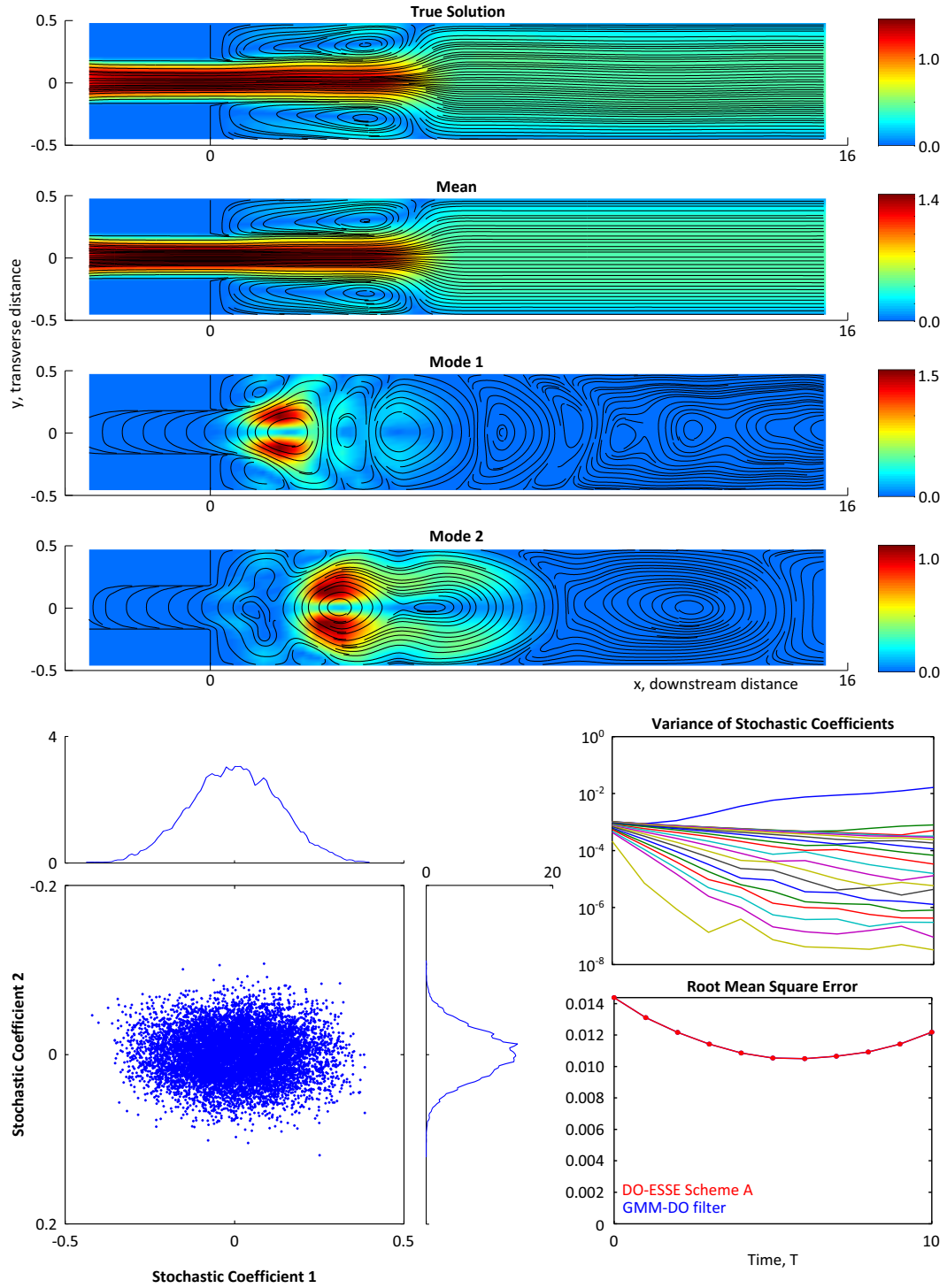


Figure 5-11: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 10$.

$T = 20$

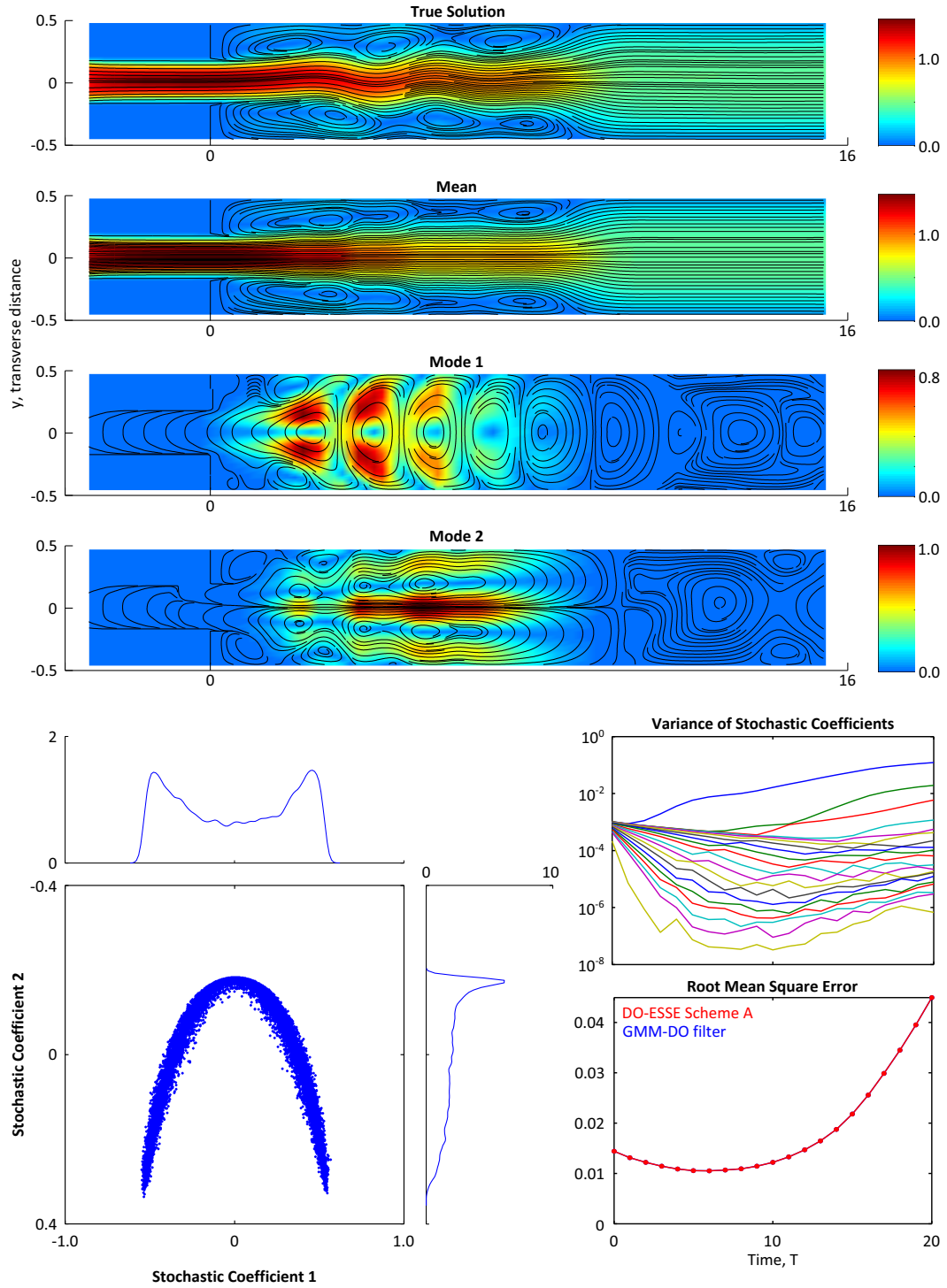


Figure 5-12: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 20$.

$T = 30$

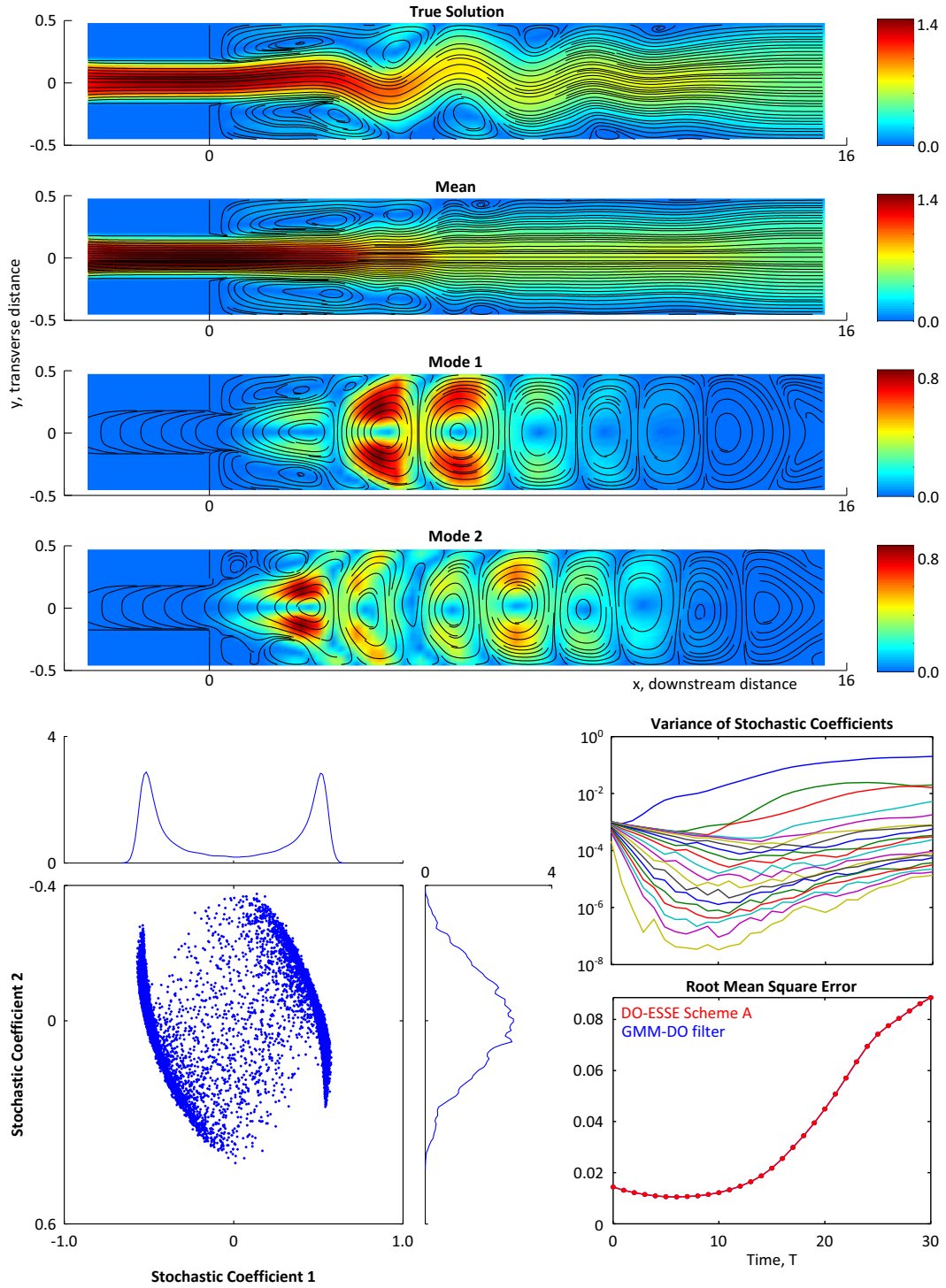


Figure 5-13: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 30$.

$T = 40$

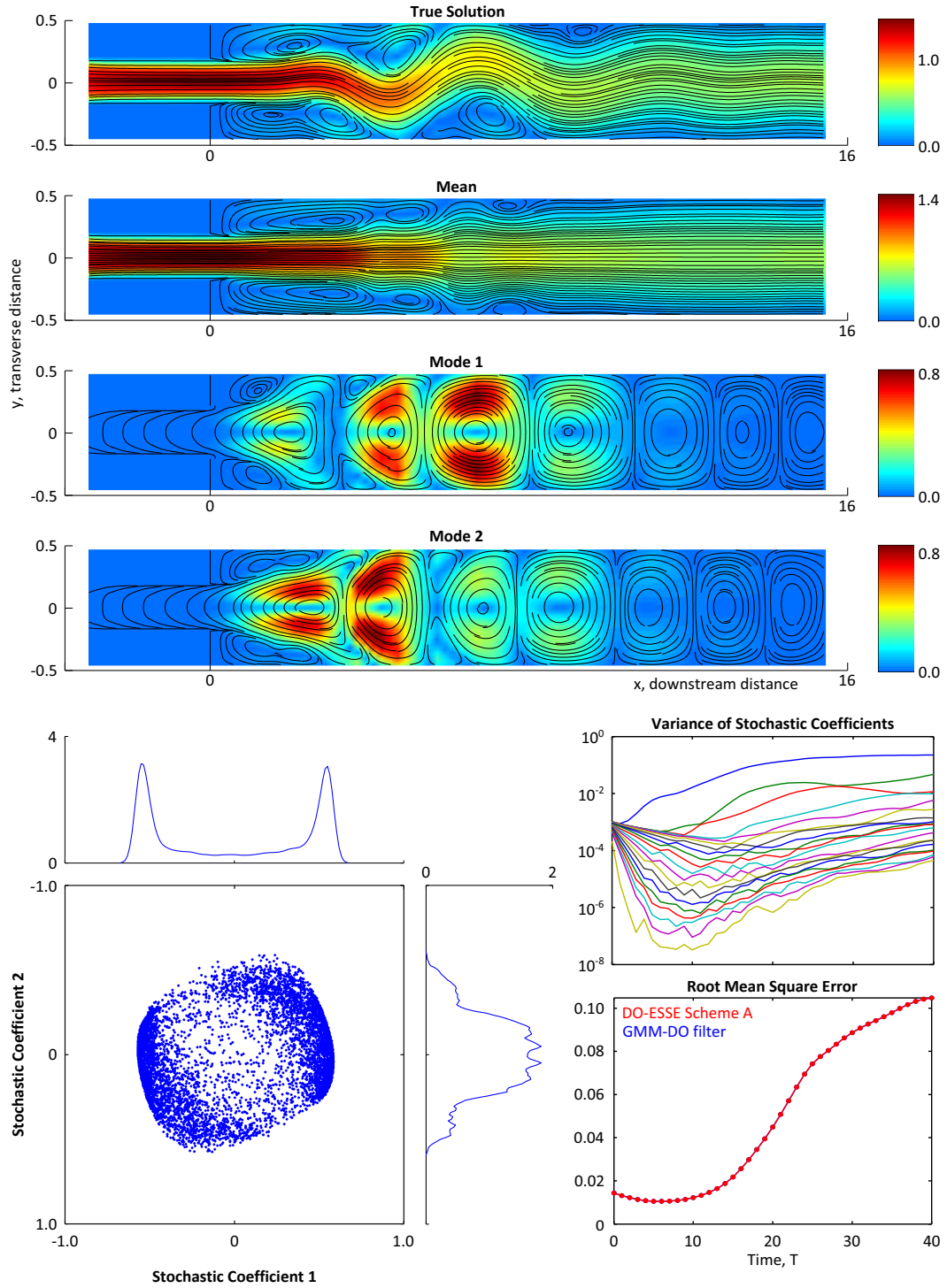


Figure 5-14: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 40$.

$T = 50$: Assimilation 1

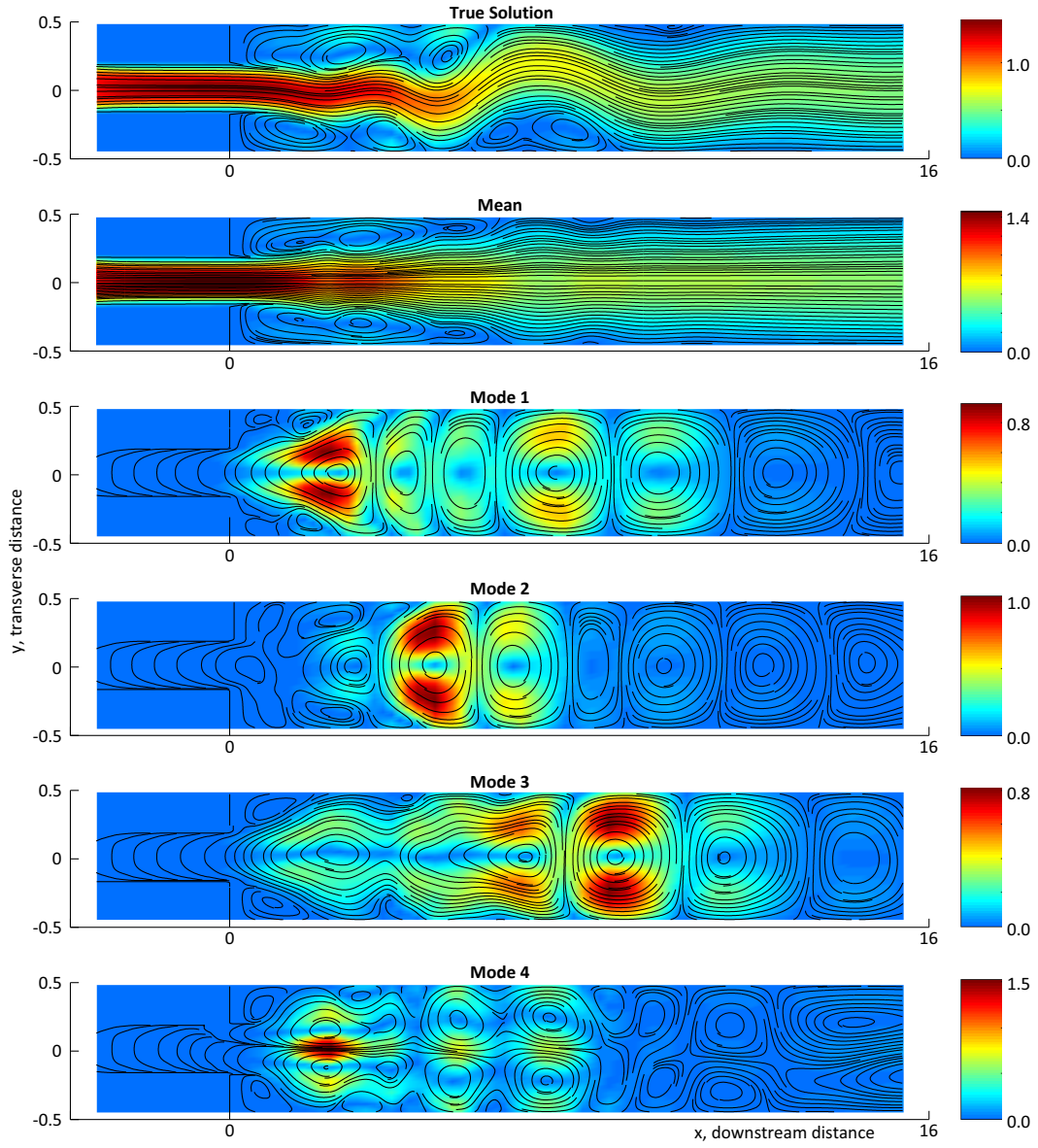


Figure 5-15: True solution; DO mean field; and first four DO modes at the first assimilation step, time $T = 50$.

$T = 50$: (i) Prior Distribution

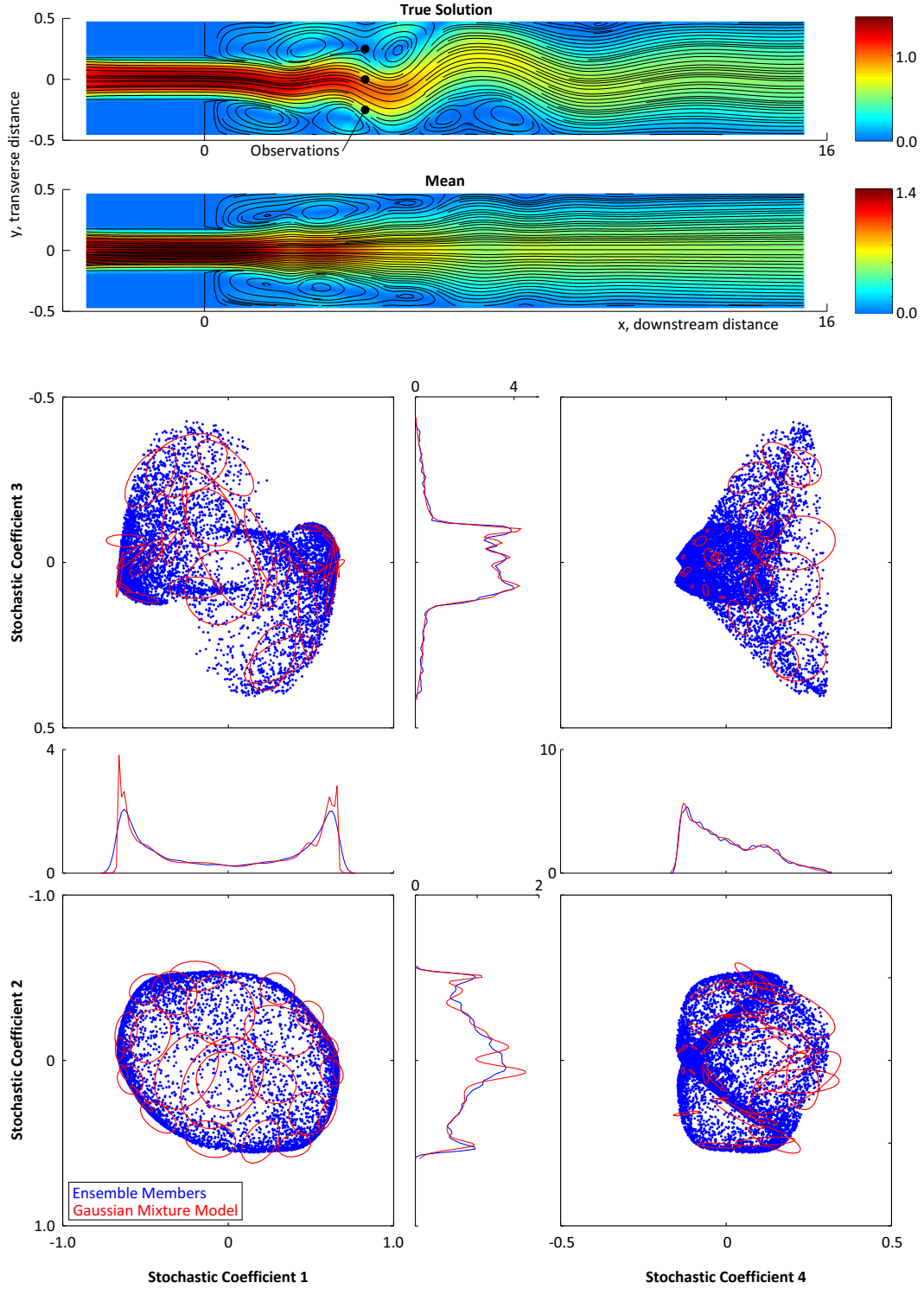


Figure 5-16: True solution; DO mean field; and joint and marginal prior distributions, identified by the Gaussian mixture model of complexity 29, and associated ensembles of the first four modes at the first assimilation step, time $T = 50$.

$T = 50$: (ii) Observations and Local Distributions

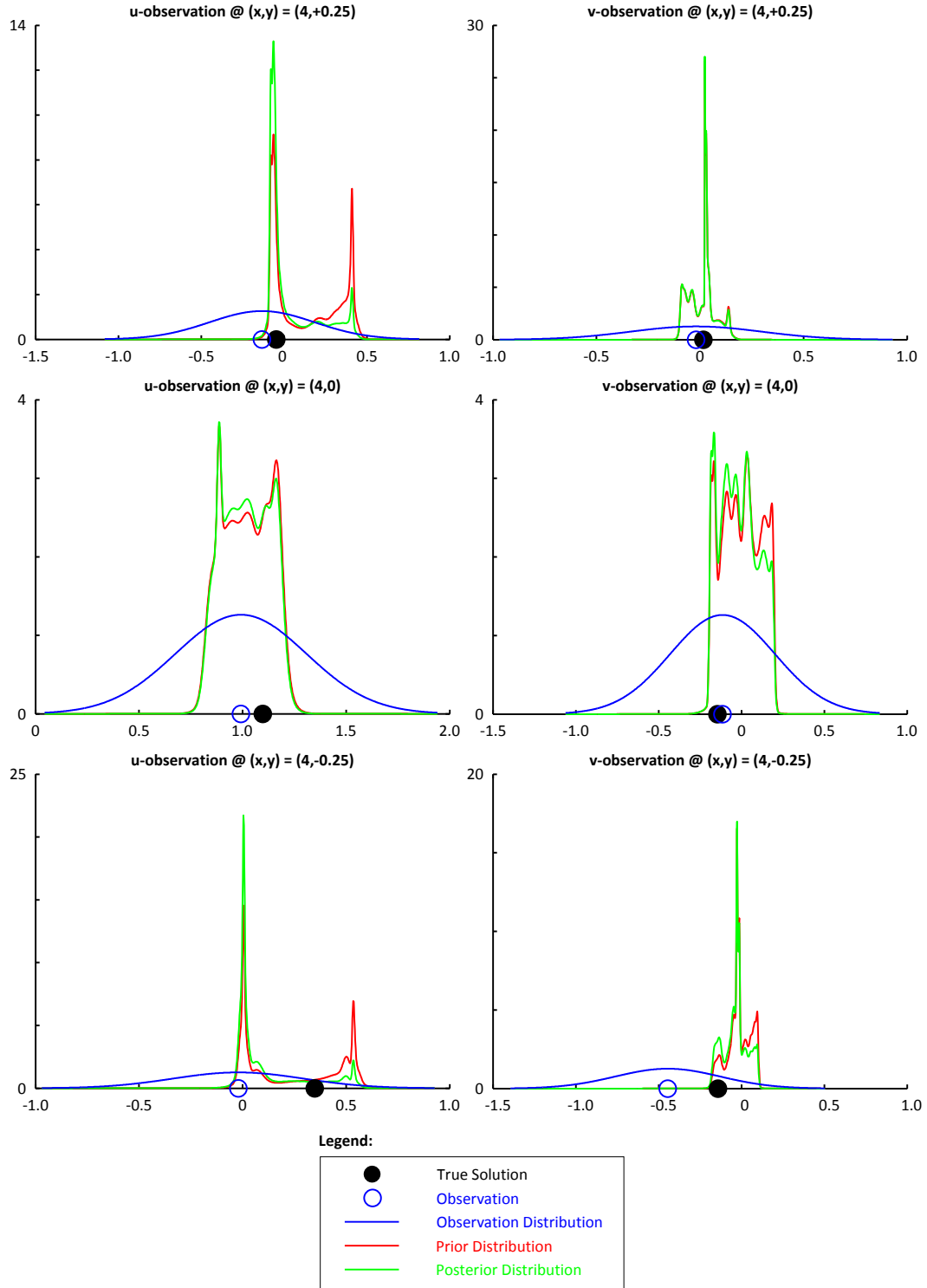


Figure 5-17: True solution; observation and its associated Gaussian distribution; and the prior and posterior distributions at the observation locations at time $T = 50$.

$T = 50$: (iii) Posterior Distribution

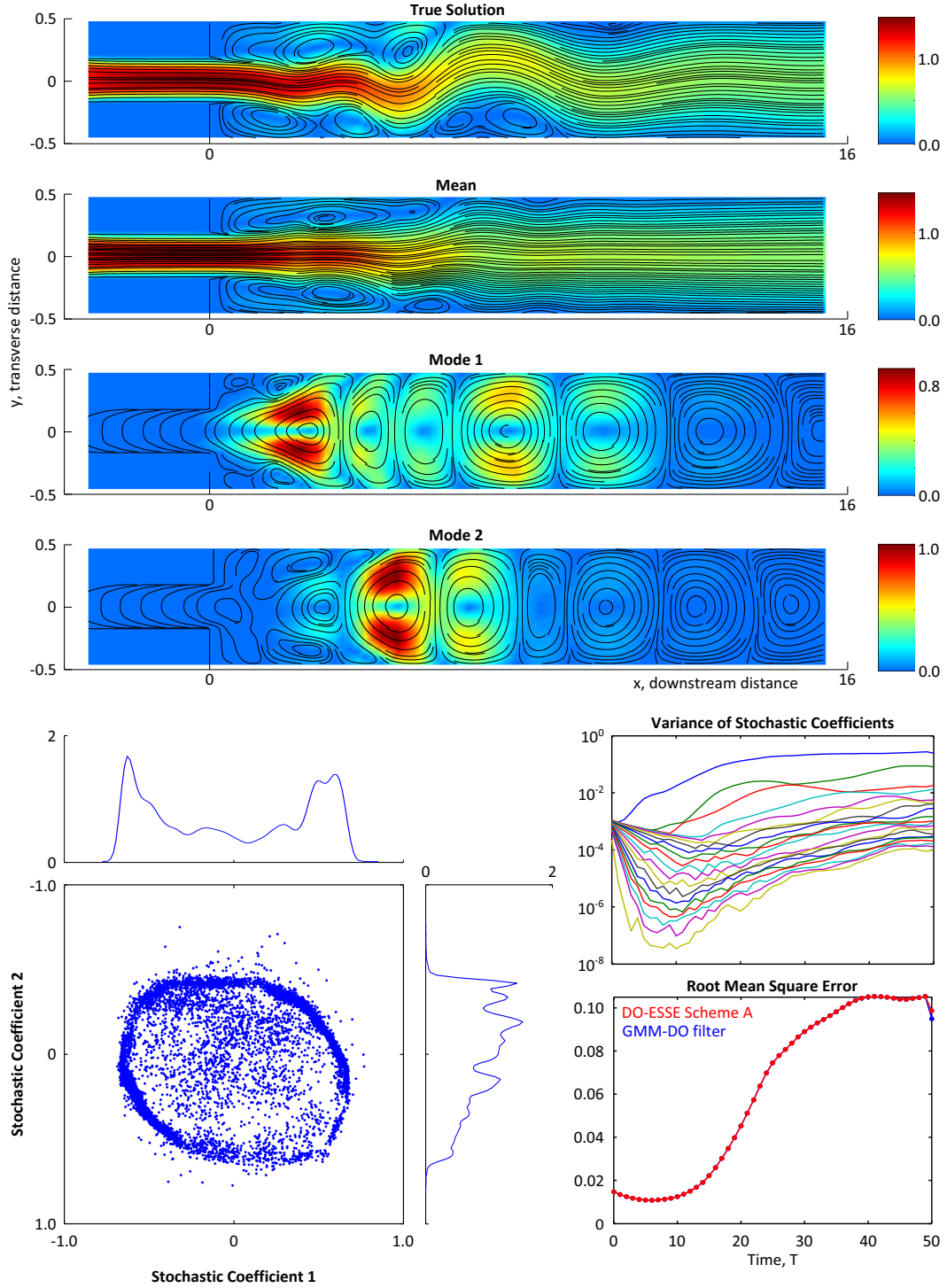


Figure 5-18: True solution; condensed representation of the posterior DO decomposition; and root mean square errors at time $T = 50$.

$T = 60$

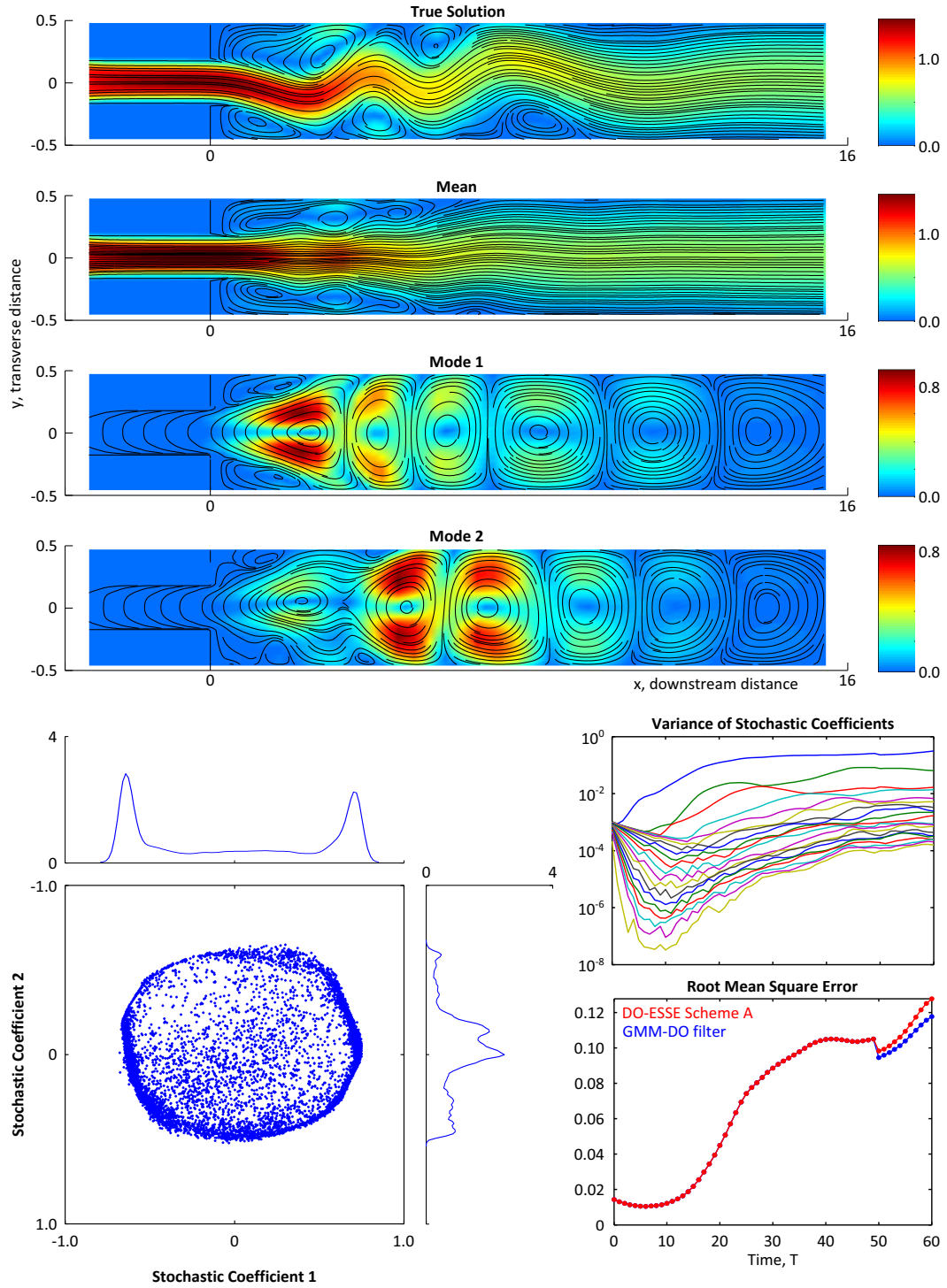


Figure 5-19: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 60$.

$T = 70$: Assimilation 2

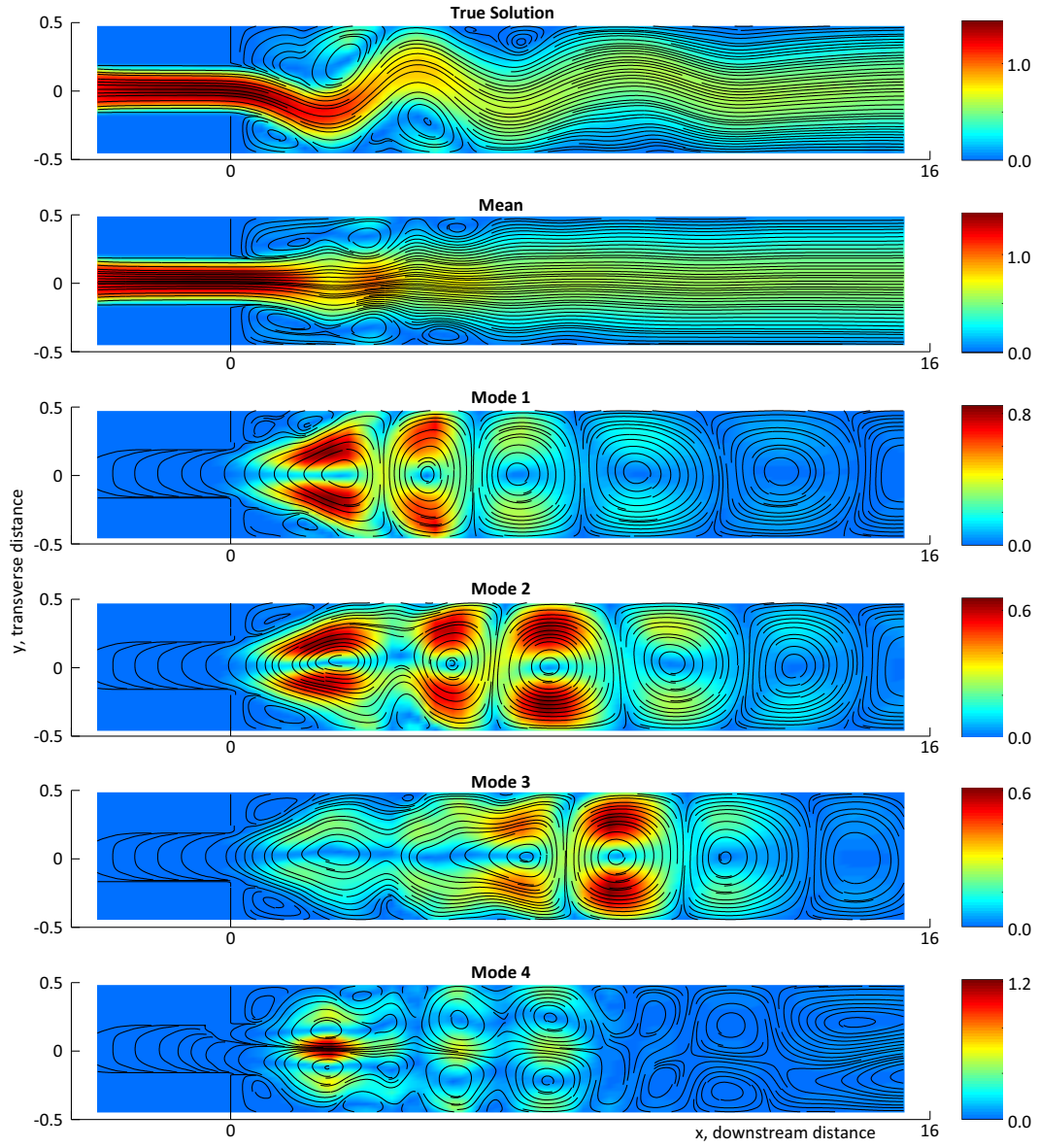


Figure 5-20: True solution; DO mean field; and first four DO modes at the second assimilation step, time $T = 70$.

$T = 70$: (i) Prior Distribution

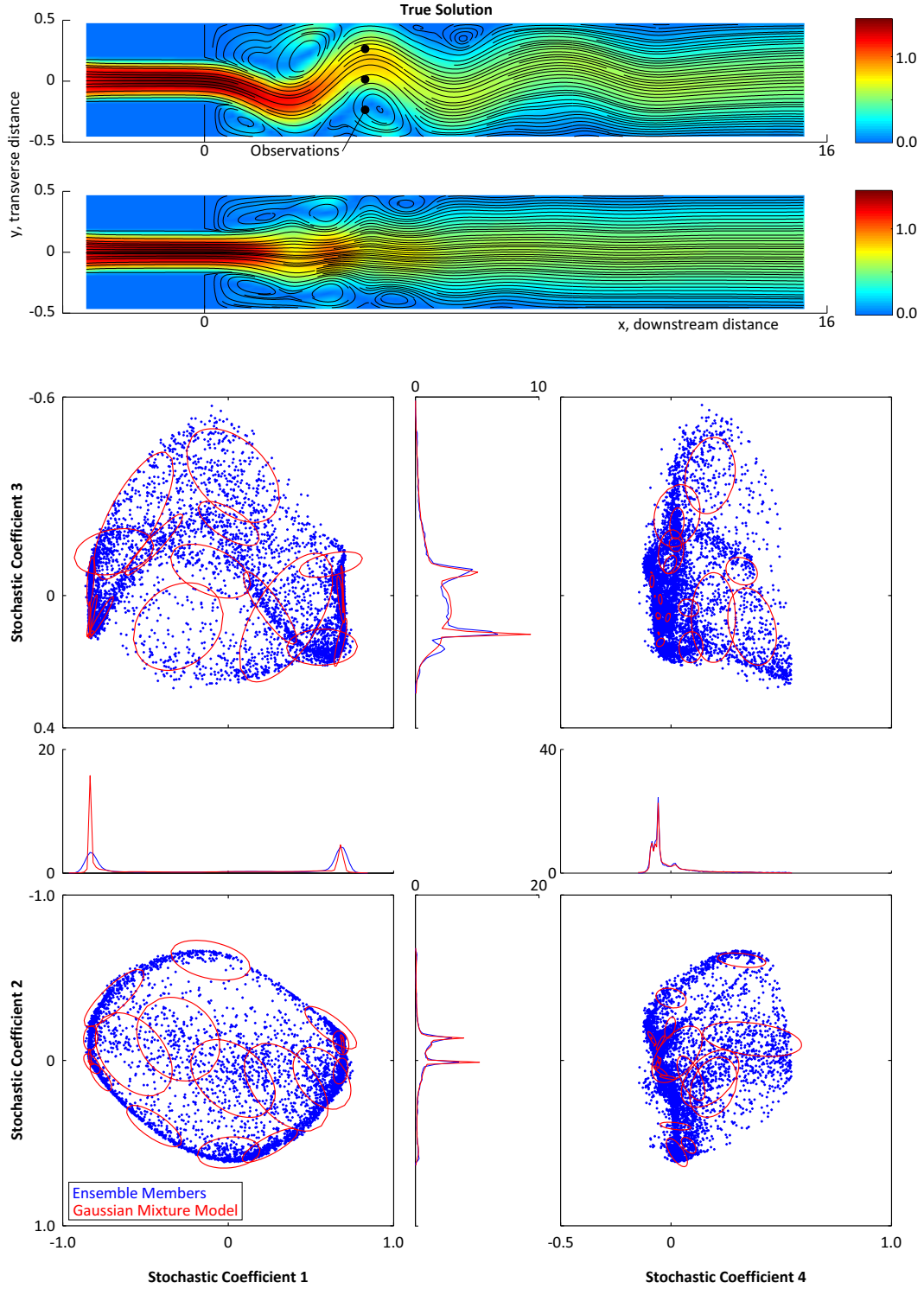


Figure 5-21: True solution; DO mean field; and joint and marginal prior distributions, identified by the Gaussian mixture model of complexity 20, and associated ensembles of the first four modes at the second assimilation step, time $T = 70$.

$T = 70$: (ii) Observations and Local Distributions

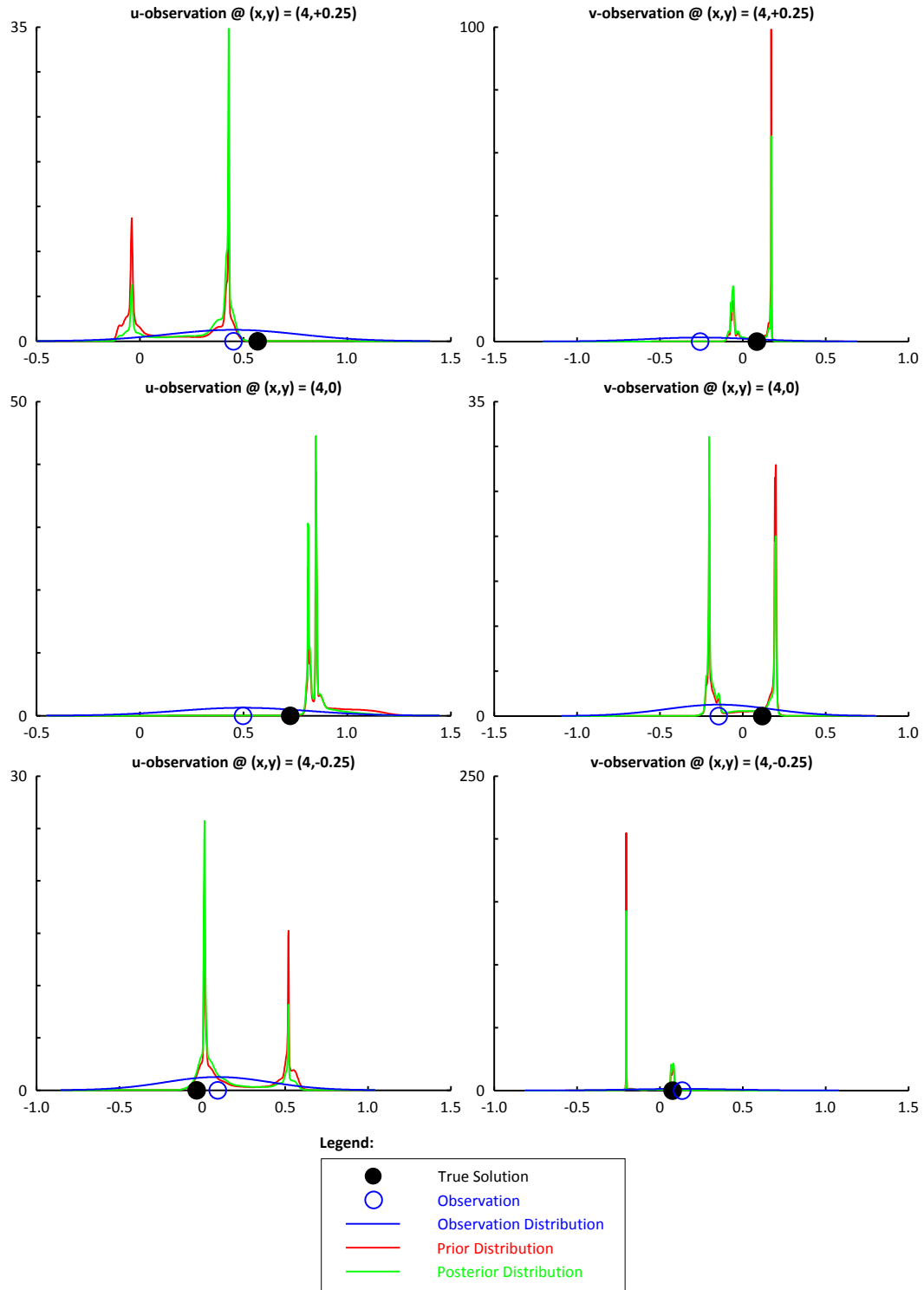


Figure 5-22: True solution; observation and its associated Gaussian distribution; and the prior and posterior distributions at the observation locations at time $T = 70$.

$T = 70$: (iii) Posterior Distribution

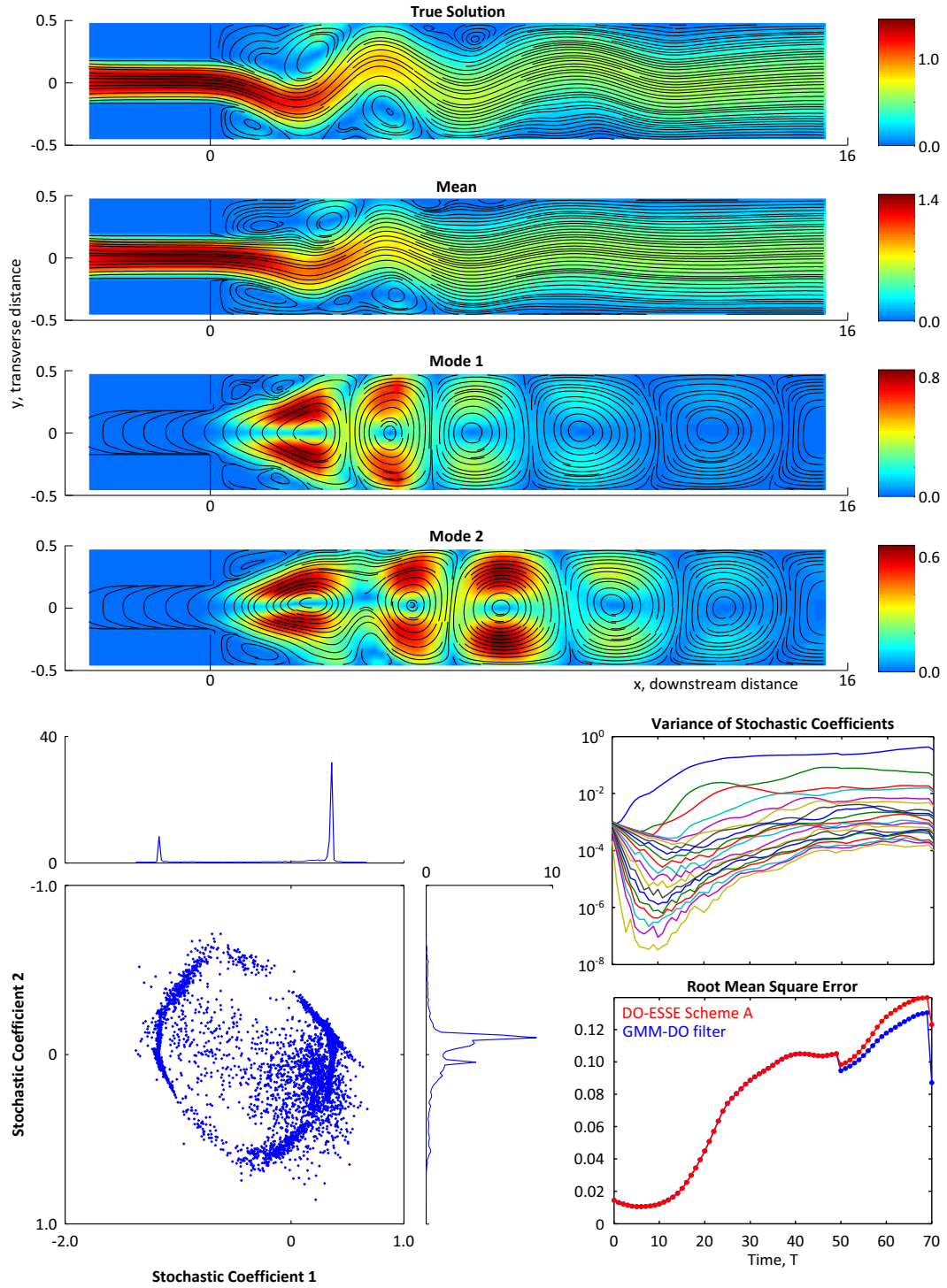


Figure 5-23: True solution; condensed representation of posterior DO decomposition; and root mean square errors at time $T = 70$.

$T = 80$

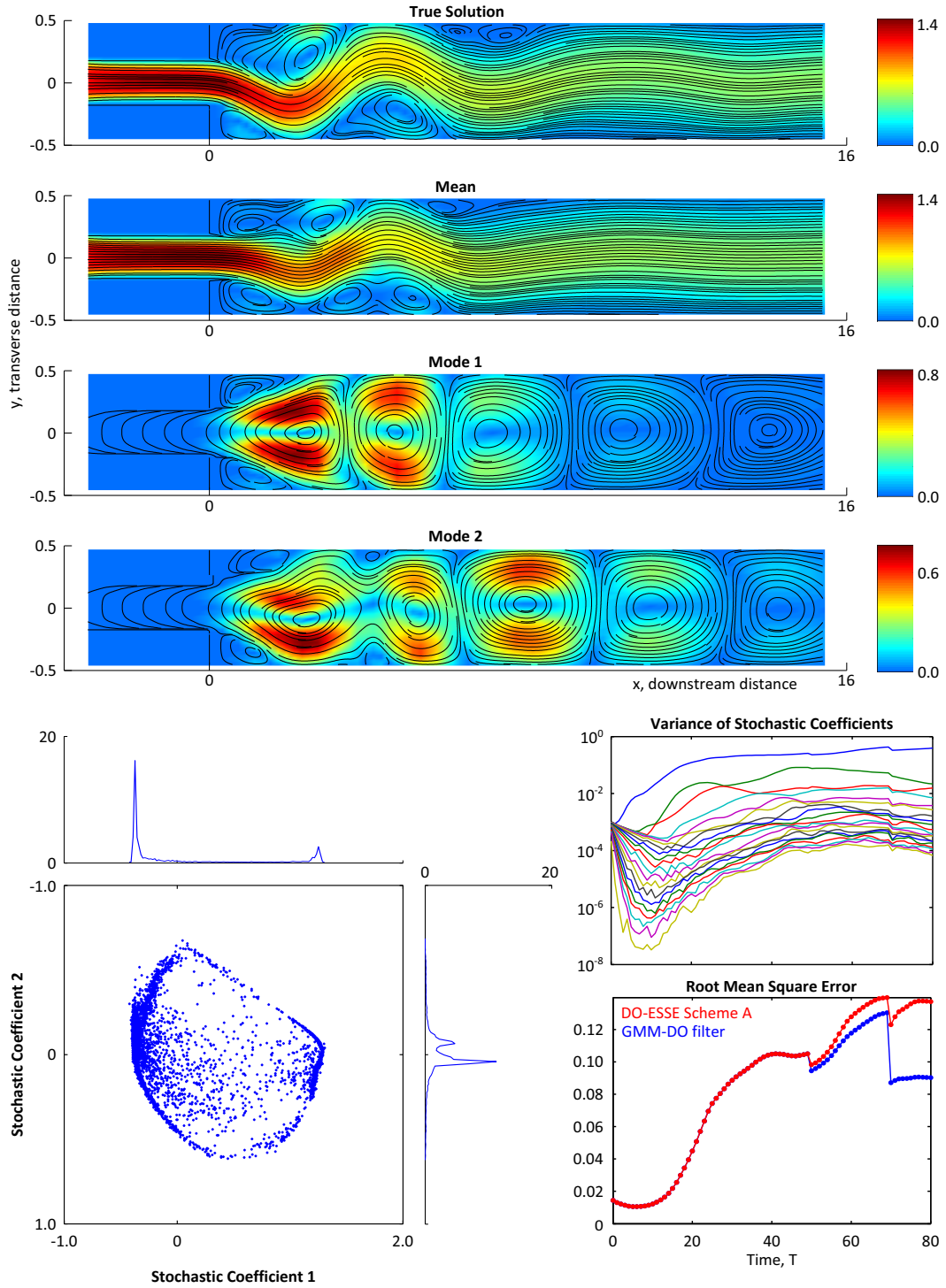


Figure 5-24: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 80$.

$T = 90$: Assimilation 3

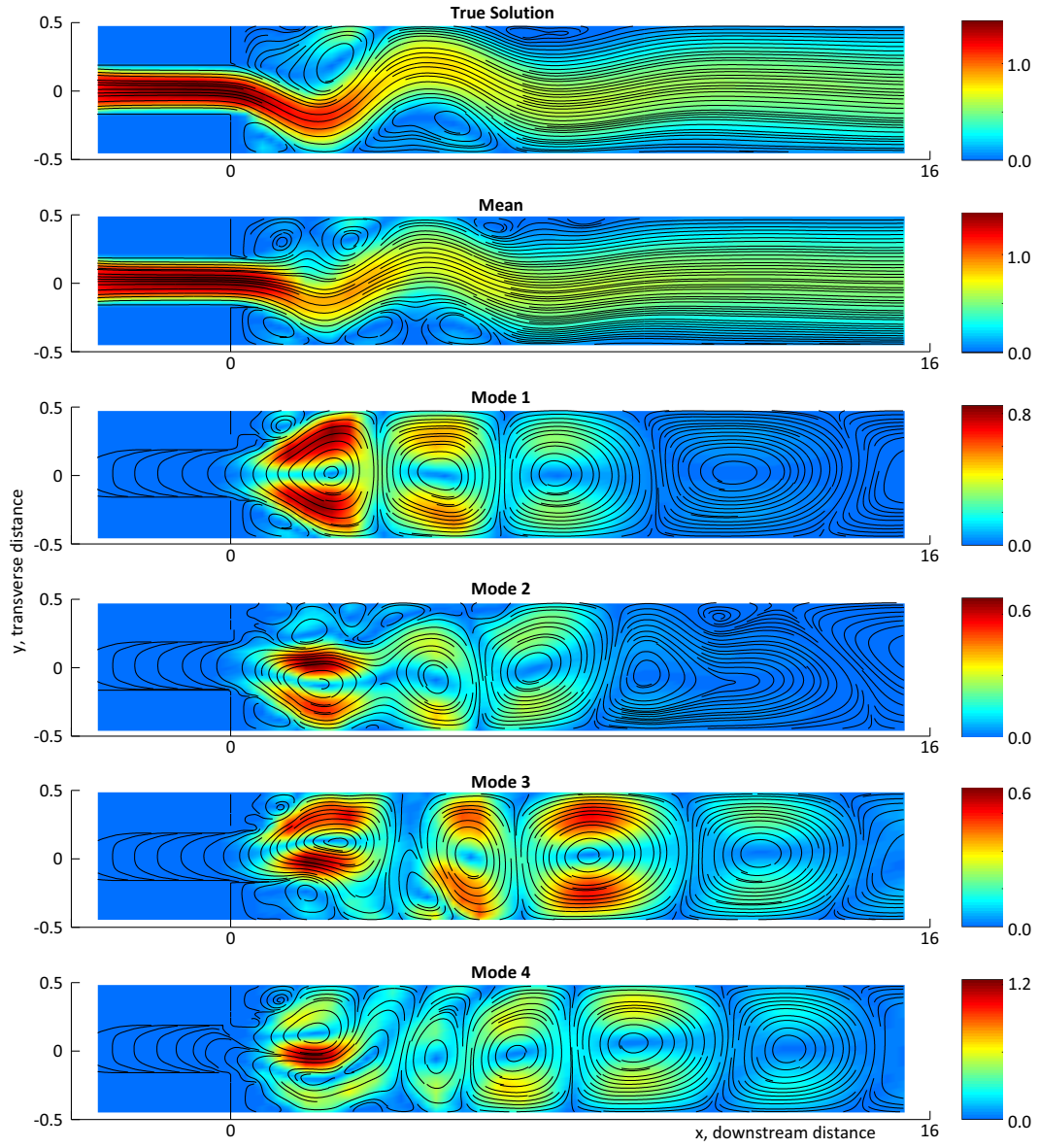


Figure 5-25: True solution; DO mean field; and first four DO modes at the third assimilation step, time $T = 90$.

$T = 90$: (i) Prior Distribution

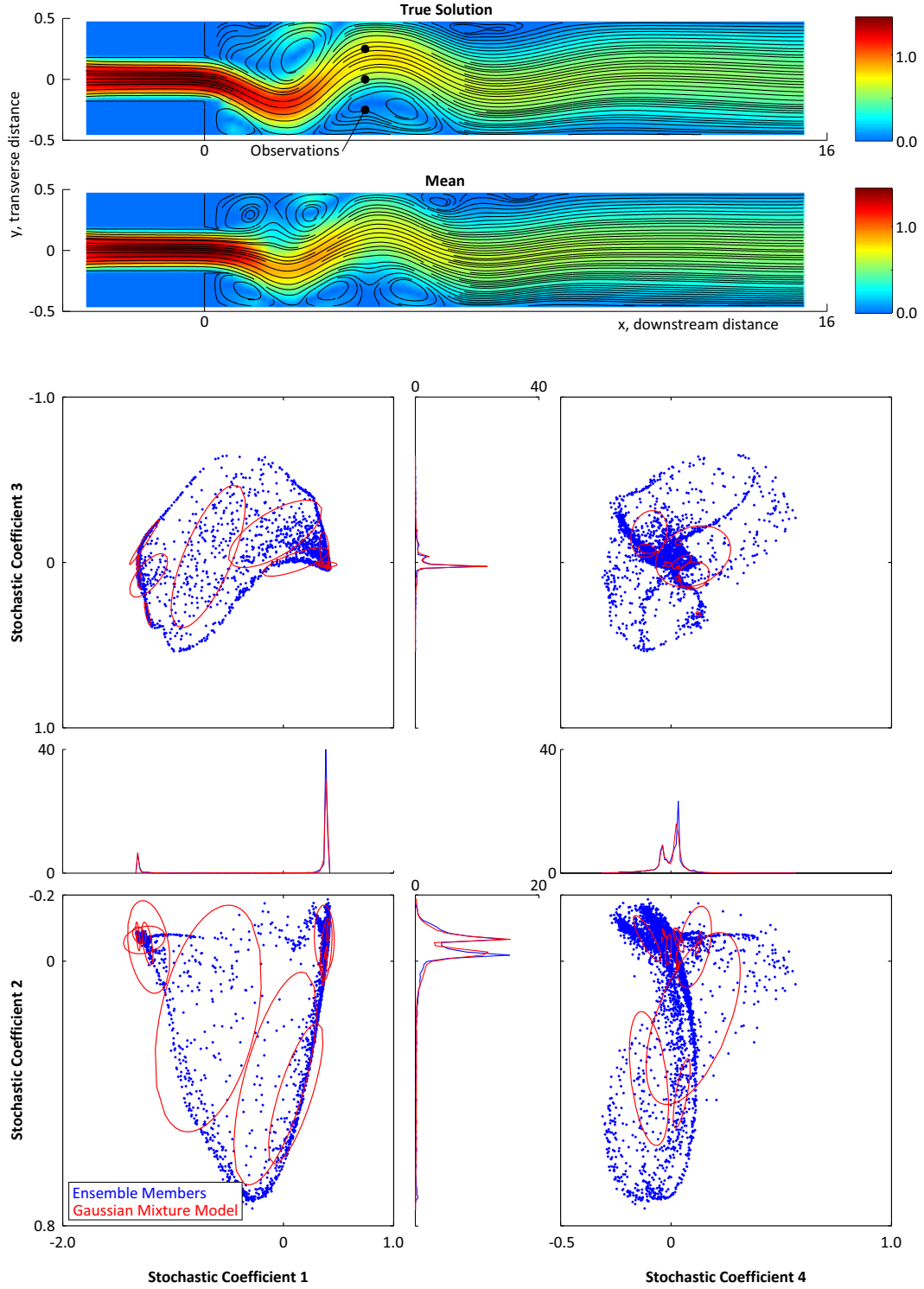


Figure 5-26: True solution; DO mean field; and joint and marginal prior distributions, identified by the Gaussian mixture model of complexity 14, and associated ensembles of the first four modes at the third assimilation step, time $T = 90$.

$T = 90$: (ii) Observations and Local Distributions

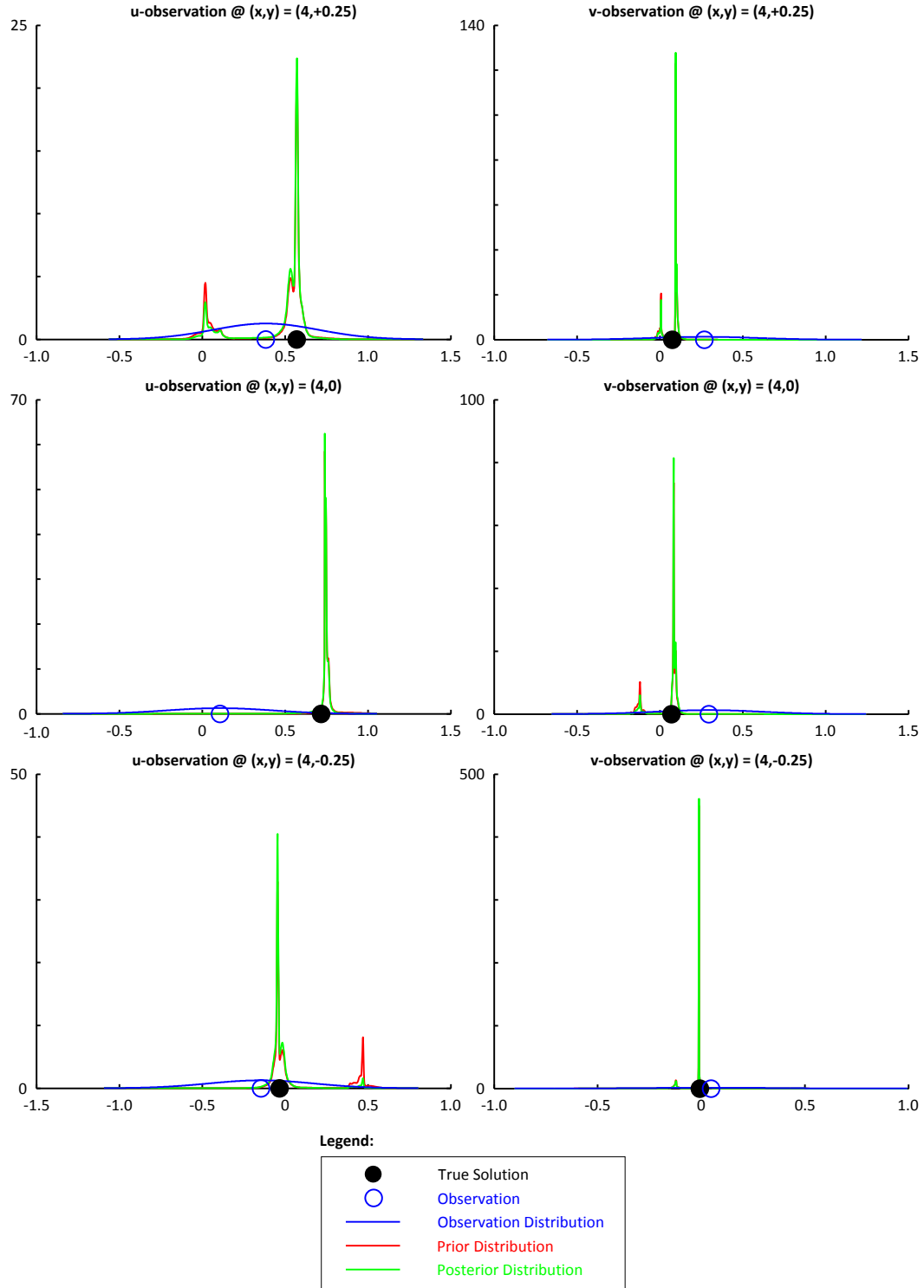


Figure 5-27: True solution; observation and its associated Gaussian distribution; and the prior and posterior distributions at the observation locations at time $T = 90$.

$T = 90$: (iii) Posterior Distribution

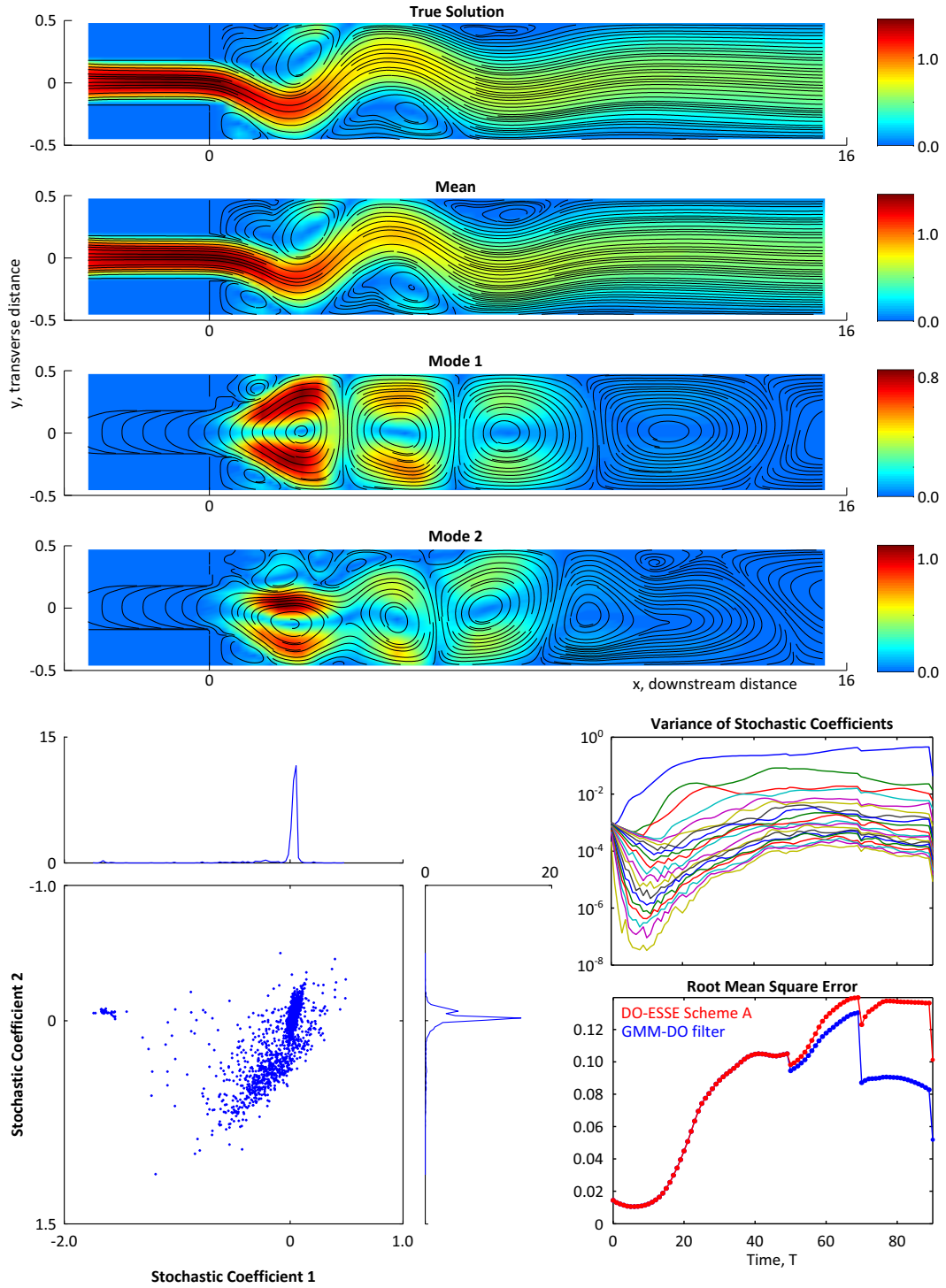


Figure 5-28: True solution; condensed representation of posterior DO decomposition; and root mean square errors at time $T = 90$.

$T = 100$

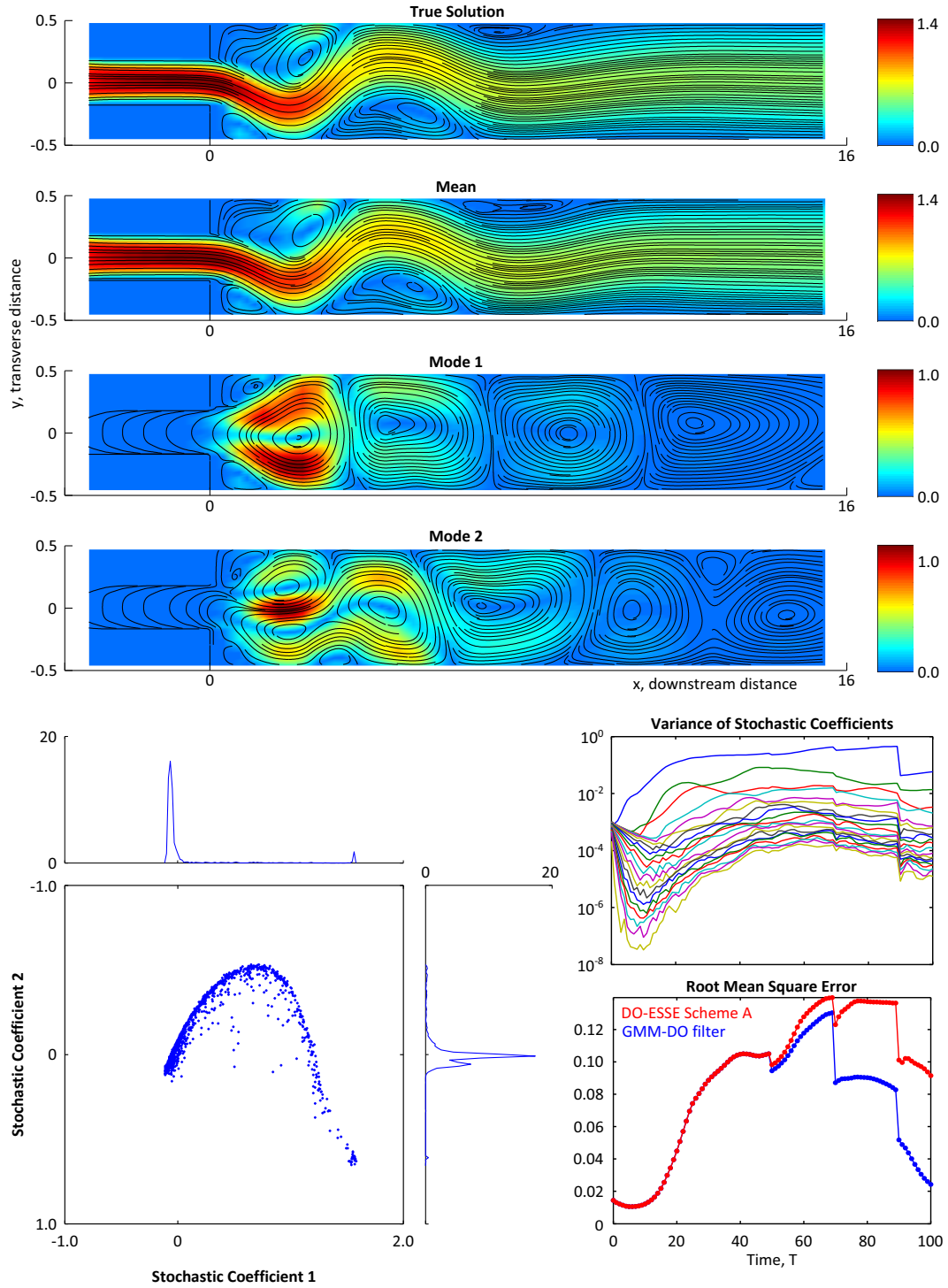


Figure 5-29: True solution; condensed representation of DO decomposition; and root mean square errors at time $T = 100$.

The previous series of figures clearly indicate the favorable results obtained by use of the GMM-DO filter, specifically when compared with the DO-ESSE scheme A. We’ve quantified this performance using the root mean square error, whose final plot we conveniently repeat in figure 5-30. We particularly note that the GMM-DO filter shows a four-fold improvement over the DO-ESSE Scheme A at the final time step, $T = 100$. Of course, these results depend on the specific truth chosen and on the properties and realizations of the observations, but we obtained relatively similar improvements for the varied examples we ran.

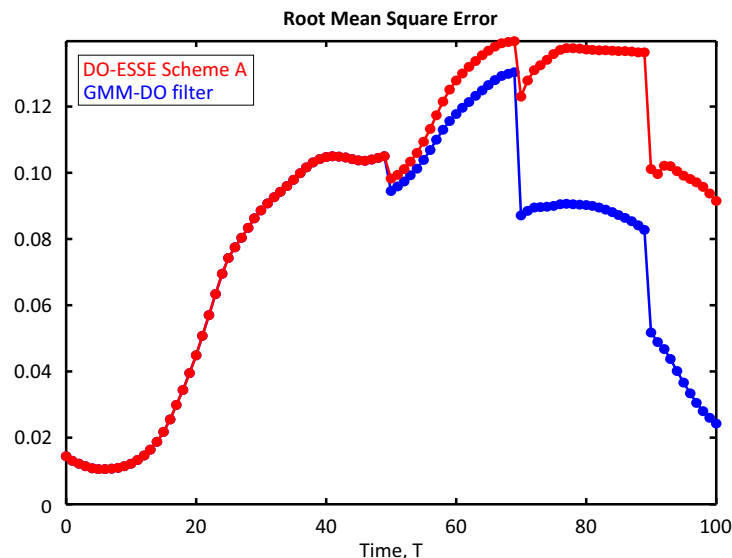


Figure 5-30: Time-history of the root means square errors for the GMM-DO filter and DO-ESSE Scheme A.

Based on the figure above, an important observation is due: the performance of the DO-ESSE scheme is comparable to that of the GMM-DO filter up until the *second* assimilation step (i.e. $T = 70$), after which the latter shows marked improvements. (This trend has been supported by runs not included in this thesis.) We attribute this observation to the GMM-DO filter’s ability to retain non-Gaussian structures upon the assimilation of data, in accordance with the exact Bayesian update, such that the state representation remains statistically accurate at later assimilation times. Focusing, for instance, on the distribution for the most dominant stochastic coefficient, Φ_1 , the GMM-DO filter suitably preserves its bimodality – however weighted – throughout the simulation, as evidenced by the appropriate marginal distribution

in the previous series of figures, 5-10 - 5-29 (an instance of which we repeat in figure 5-31 for clarification).

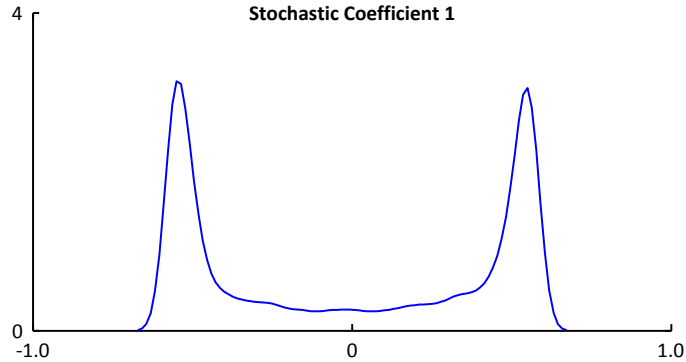


Figure 5-31: Bimodal distribution for the most dominant stochastic coefficient, Φ_1 , at $T = 40$. The GMM-DO filter captures and retains this bimodality throughout the simulation of the sudden expansion fluid flow, resulting in its superior performance.

Clearly resembling the structure of the Double Well Diffusion Experiment, we argue that the bimodality of the most dominant stochastic coefficient reflects the ambiguity of direction with which the sudden expansion fluid flow prefers to break, as alluded to in our introduction to this test case. As such, it is crucial for the filter to not only capture these non-Gaussian structures when approximating the prior distribution, but equally to preserve them following the Bayesian update. This, we make possible with the GMM-DO filter. While we do not include detailed results for the DO-ESSE Scheme A in this thesis, it is evident that the fitting of single Gaussian distributions to ensemble sets, $\{\phi\}$, of such complexities removes any non-Gaussian structure, thus resulting in its relatively poor performance.

The ability of the fitted Gaussian mixture model to accurately capture the multi-dimensional distribution given by the ensemble set, $\{\phi\}$, at the time of assimilation of new data is visualized in figures 5-16, 5-21 and 5-26. For the sake of convenience, we repeat figure 5-16 in figure 5-32, and refer to the latter in what follows. Using MATLAB's 'ksdensity' function (represented by the blue marginal distributions in figure 5-32) as an appropriate approximation for the optimal marginal distribution based on an arbitrary data set, we note the remarkable accuracy with which the Gaussian mixture model captures this distribution. With figure 5-32, we equally attempt

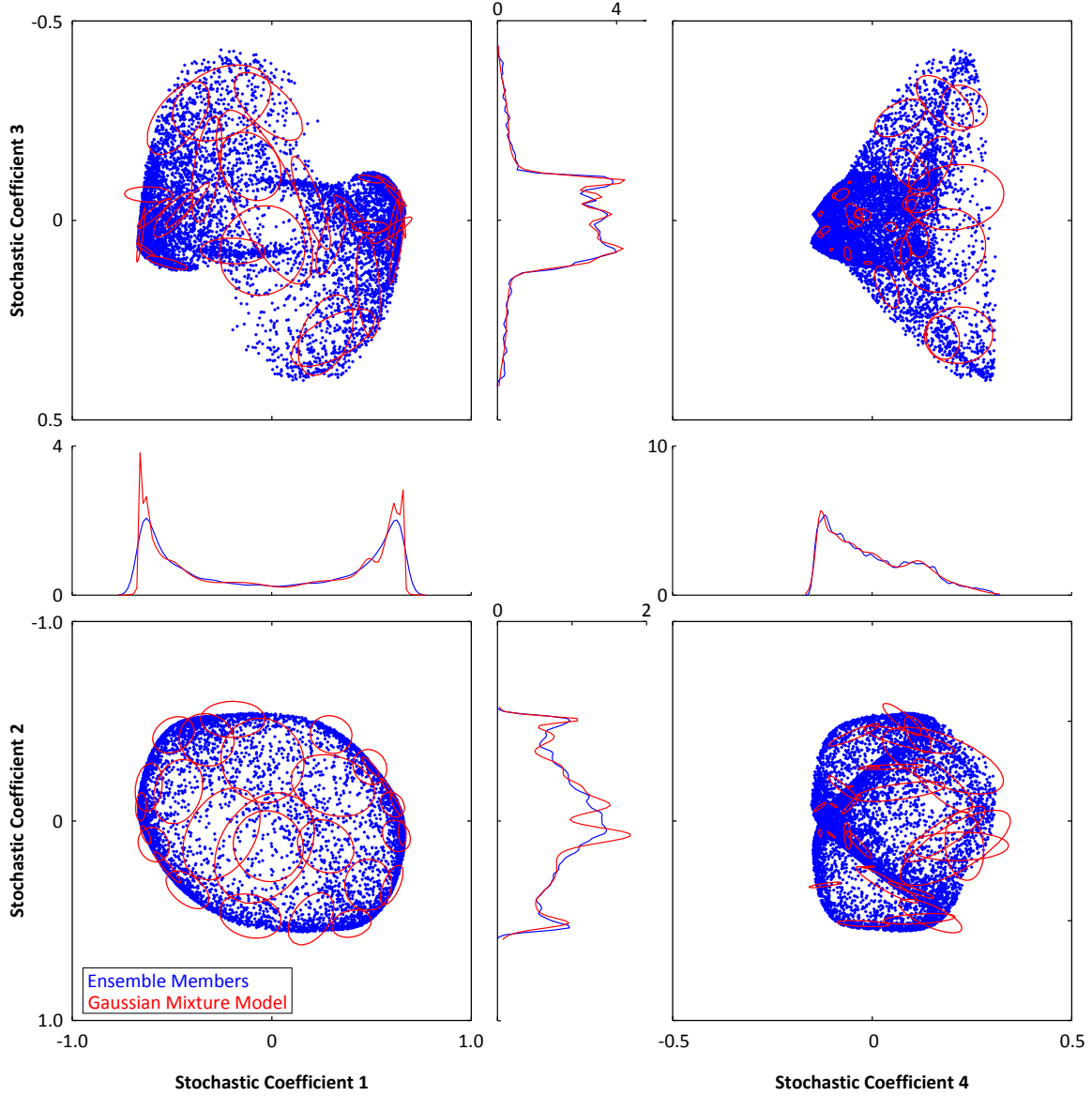


Figure 5-32: Gaussian mixture model approximation of the ensemble set, $\{\phi\}$, at the time of the first assimilation of observations, $T = 50$. Assuming MATLAB's 'ksdensity' function to represent an appropriate approximation to the true marginal densities, we note the satisfactory approximation of the Gaussian mixture model.

to convey the manner in which this accuracy extends to the multidimensional case. In figures 5-16, 5-21 and 5-26, we further note how the mixture complexity reflects the complexity of the ensemble set, approximating the latter at the first assimilation step by 29 mixtures; the second by 20 mixtures; and the third by 14 mixtures – as determined by the BIC. This adaptability allows the scheme to fully capture the non-Gaussian structures in an optimal way, again as visualized in figure 5-32.

A clear strength of the GMM-DO filter is its ability to statistically converge to the true solution, as visualized particularly in figures 5-17, 5-22 and 5-27. Such is the consequence of combining the DO equations for evolving the state representation with Gaussian mixture models and the EM-BIC algorithm for approximating the given ensemble set. In the following analysis, we focus in particular on the top left panel of figure 5-17, repeated for convenience in figure 5-33.

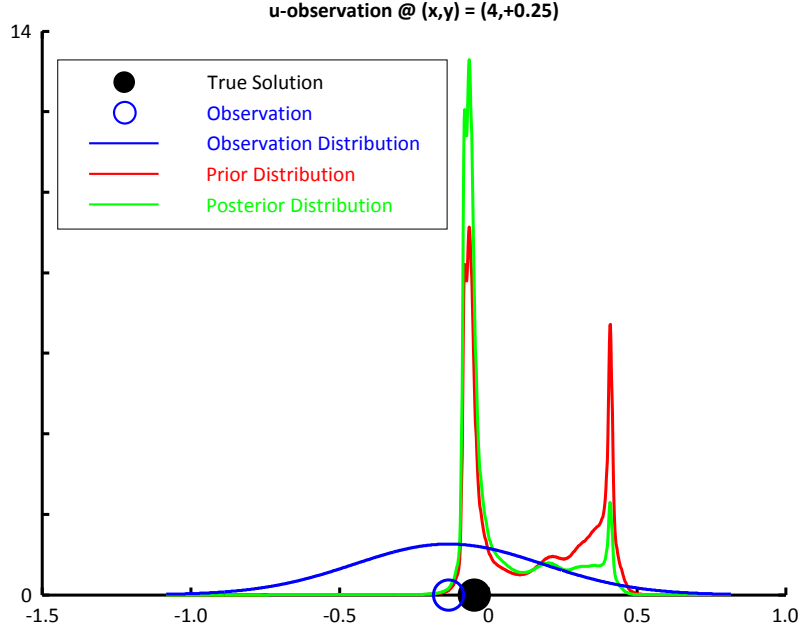


Figure 5-33: An example of the manner in which the GMM-DO filter captures the true solution through its use of Gaussian mixture models. We equally note the increased weights placed on the mixtures surrounding the true solution following the Bayesian update, depicted by the green curve.

From the figure above, we point out a number of observations. To our satisfaction, the prior distribution – bimodal in nature – perfectly captures the true solution. Had we instead used a Gaussian approximation for the prior distribution, the true solution would merely have been nested within the tail of the Gaussian and thus inadequately represented. Of further notice is the shape of the posterior distribution, placing greater weight on the mixtures surrounding the true solution that make up the left lobe of the bimodal distribution. We again wish to emphasize that the update is done exactly under Bayes Law. A similar example, in which the true solution is nested within a set of mixtures of smaller – yet finite – weight is given figure 5-34. Again, we

note the shape of the posterior distribution, placing greater weight on these mixtures. This would not be captured when resorting to a regular Gaussian distribution.

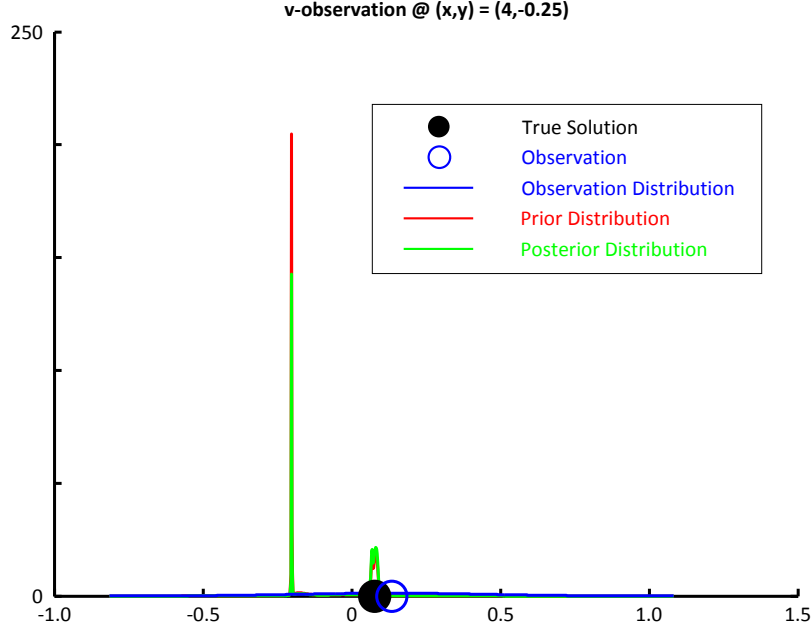


Figure 5-34: A second example of the manner in which the GMM-DO filter captures the true solution through its use of Gaussian mixture models. Here, however, the true solution is contained within a mode of small – but finite – probability. Note the increased weights placed on the mixtures surrounding the true solution following the Bayesian update.

Finally, as alluded to in the introduction to this chapter, choosing the dimensionality of the stochastic subspace, s , is crucial to ensure that the GMM-DO filter suitably captures the true statistics of the system. For the purposes of this test case, we had allowed this dimensionality to be constant with $s = 20$, supported by a number of prior convergence tests. An example of such is visualized in figure 5-35, in which we display the part of the true solution (termed ‘error’) orthogonal to the stochastic subspace for the case of 15 and 20 modes at the time of the first assimilation. We note that in practice such comparisons with the true solution can for obvious reasons not be done, and instead statistical comparisons with the observations and their inherent uncertainties must be made. Furthermore, referring to recent work by Sapsis and Lermusiaux (2010), it is intended in a future work to let the stochastic dimensionality, s , be variable and driven by the dynamics of the system.

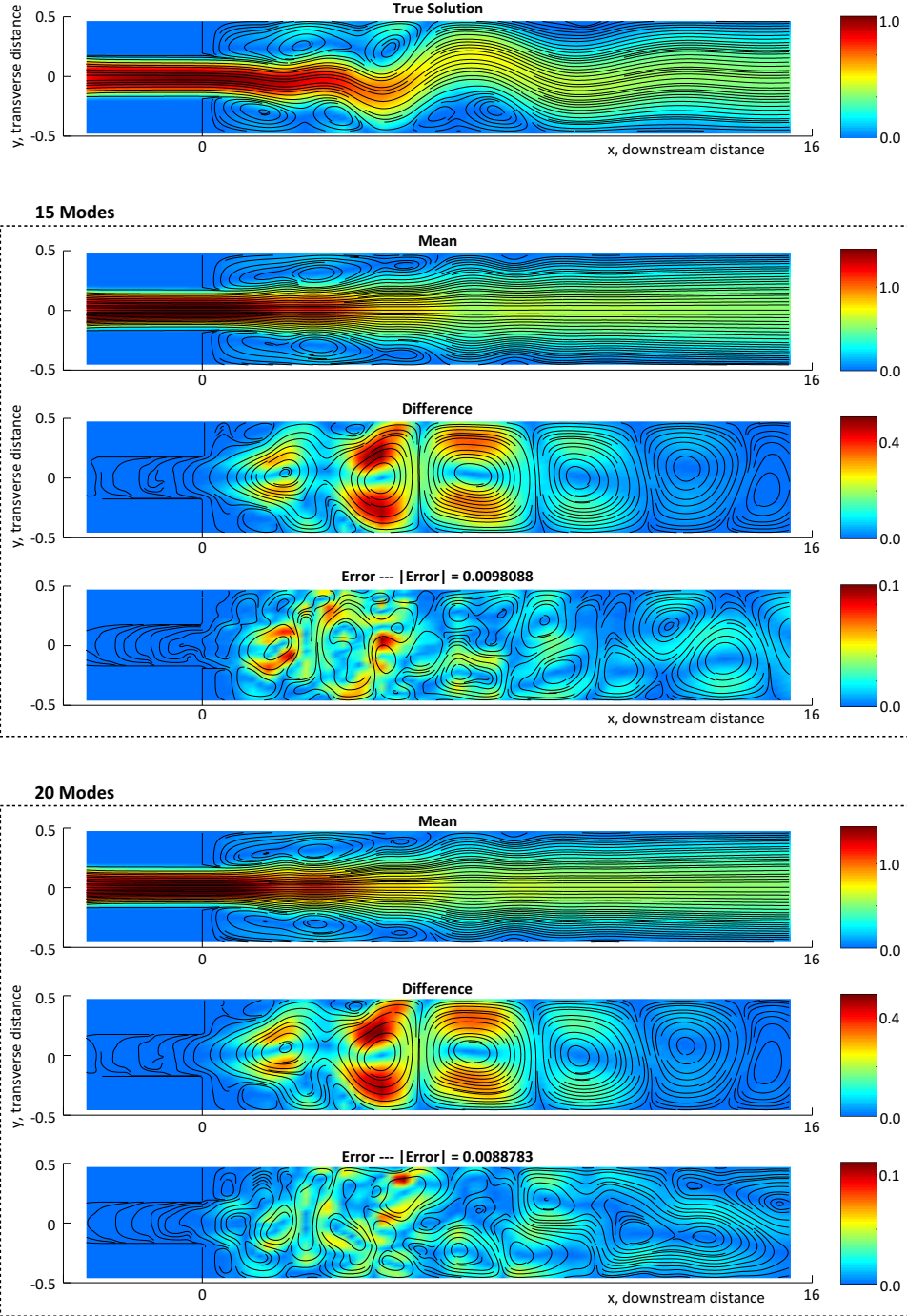


Figure 5-35: We show the part of the true solution orthogonal to the stochastic subspace for the case of 15 and 20 modes at the time of the first assimilation. ‘Difference’ refers to the difference between the true solution and the mean field; ‘error’ to the part of the true solution not captured by the GMM-DO filter. We note that as we increase the number of modes, the norm of the error marginally decreases, indicative of convergence.

5.5 Conclusion

We have examined the application of the GMM-DO filter to a two-dimensional sudden expansion fluid flow of aspect ratio 3 and $Re = 250$, at which the test case admits a steady, asymmetric flow. Given the sensitivity of the preferred direction of breaking to initial perturbations, the flow admits complex, far-from-Gaussian distributions and as such is particularly well-suited to evaluating the performance of the GMM-DO filter.

Based on the root mean square error, we found the GMM-DO filter to significantly outperform the DO-ESSE Scheme A. Specifically, utilizing temporally and spatially sparse measurements of relatively large uncertainty, the GMM-DO filter accurately predicted the structure of the true solution at the final time, $T = 100$, made evident in figure 5-29. We attribute this performance to the GMM-DO filter’s ability to accurately capture and retain the inherent far-from-Gaussian statistics, both prior and posterior to the melding of data, in exact accordance with the Bayesian update.

With the sudden expansion fluid flow, we have proven the applicability of the GMM-DO filter to realistic 2D Navier-Stokes flow test cases of non-trivial dimensionality, made possible by application of the DO equations. By focusing on a dominant stochastic subspace of the full state space, we allow the fitting of Gaussian mixture models with the EM-BIC algorithm – an otherwise computationally intractable procedure. For test cases admitting complex statistics, we have shown the latter to crucially improve the filtering skill. In future work, the GMM-DO equations and filtering schemes can be implemented for a full, 3D ocean model and their performance evaluated in multiscale ocean simulations (Haley and Lermusiaux, 2010).

Chapter 6

Conclusion

In an introductory chapter we emphasized the importance played by the forecast covariance matrix in appropriately assimilating and distributing information due to sparse observations. In this context, progress has more recently been made in identifying the advantages of adopting Gaussian mixture models for approximating the prior distribution, allowing the update step to capture and retain potential non-Gaussian structures. Its success has been shown using a number of simplified test cases, including the classic Lorenz-63 model (Lorenz, 1963). Later publications, specifically those due to Smith (2007) and Dovera and Rossa (2010), further introduced both the EM algorithm and model selection criteria in a Monte Carlo setting for arriving at the optimal mixture parameters, resulting in a more accurate resolution of the true probability density function. All of this is equally utilized in the GMM-DO filter.

One novelty of the GMM-DO filter lies in having identified the necessity to couple the previous concepts with an efficient reduced order model, specifically the Dynamically Orthogonal field equations due to Sapsis and Lermusiaux (2009). By limiting our attention to a dominant stochastic subspace of the total state space, we thus bridge an important gap previously identified in the literature caused by the dimensionality of the state space. Particularly, with this, we make obsolete ad hoc localization procedures previously adopted – with limited success - by other filters introduced in this thesis. With the GMM-DO filter, we further stray from the redundant operating on ensemble members during the update step; rather, we manipulate directly the

determined Gaussian mixture model exactly under Bayes' Law with the assumption that the aforementioned distribution sufficiently captures the present non-Gaussian structures.

In this thesis, we successfully applied the GMM-DO filter to two test cases: (1) the Double Well Diffusion Experiment and (2) the Sudden Expansion fluid flow. With the former, we proved the validity of utilizing the combination of Gaussian mixture models, the EM algorithm and Bayes Information Criterion in a classical filtering context. Specifically, for the range of parameter values investigated, we found that the GMM-DO filter outperformed the Ensemble Kalman filter in its ability to capture the transition of the ball from one well to the other. Furthermore, for only a moderate number of particles was the performance of the GMM-DO filter comparable to that of the Maximum Entropy filter, the latter of which is particularly well-suited to the given test case.

With the application of the GMM-DO filter to the Sudden Expansion fluid flow, we proved its applicability to realistic two-dimensional Navier-Stokes test cases of non-trivial dimensionality. The GMM-DO filter was shown to consistently capture the far-from-Gaussian statistics associated with the test case, resulting in its superior performance over filters that would invoke the Gaussian assumption.

In conclusion, we present the GMM-DO filter as an efficient, data-driven assimilation scheme, focused on a dominant stochastic subspace of the total state space, that respects nonlinear dynamics and captures non-Gaussian statistics, obviating the use of heuristic arguments.

Appendix A

Jensen's Inequality and Gibbs' Inequality

The following presentation is completely due to Wornell (2010). We focus on discrete random variables in the proofs; the statements are also true for continuous random variables.

Definition: Convex Set

Let \mathcal{V} be a convex set. Function $\phi(\cdot) : \mathcal{V} \rightarrow \mathbb{R}$ is convex if for any $v_1, v_2 \in \mathcal{V}$ and any $\lambda \in [0, 1]$,

$$\phi(\lambda v_1 + (1 - \lambda)v_2) \leq \lambda\phi(v_1) + (1 - \lambda)\phi(v_2) \quad (\text{A.1})$$

A function that satisfies the definition with a strict inequality for all $\lambda \neq 0, 1$ is called *strictly convex*. We call a function $\phi(\cdot)$ *concave* if $-\phi(\cdot)$ is convex.

With this we are ready to state and prove the inequalities of interest.

Definition: Jensen's Inequality

If $\phi(\cdot)$ is a concave function and V is a random variable defined over alphabet \mathcal{V} (i.e. that values which it can take), then

$$\mathcal{E}[\phi(V)] \leq \phi(\mathcal{E}[V]) \quad (\text{A.2})$$

If $\phi(\cdot)$ is strictly concave, equation (A.2) holds with equality if and only if V is a

deterministic constant.

Proof

We will prove this inequality by induction on the size of the alphabet \mathcal{V} . First, we consider $|\mathcal{V}| = 2$ (i.e. random variable V can only take on two values) and let v_1 and v_2 be the two elements in \mathcal{V} . The definition of concavity implies

$$\mathcal{E}[\phi(V)] = p_V(v_1)\phi(v_1) + p_V(v_2)\phi(v_2) \leq \phi(p_V(v_1)v_1 + p_V(v_2)v_2) = \phi(\mathcal{E}[V]), \quad (\text{A.3})$$

where $p_V(v)$ is the probability density function for random variable V .

We now consider $\mathcal{V} = \{v_1, \dots, v_M\}$, $M > 2$, and assume that equation (A.2) holds for all random variables defined over alphabets smaller than M elements. Suppose V is not deterministic, i.e. there exists v_i such that $p_V(v_i)$ is neither 0 nor 1. In this case, we can perform the following algebraic manipulations:

$$\mathcal{E}[\phi(V)] = \sum_{m=1}^M p_V(v_m)\phi(v_m) = p_V(v_i)\phi(v_i) + \sum_{m \neq i} p_V(v_m)\phi(v_m) \quad (\text{A.4})$$

$$= p_V(v_i)\phi(v_i) + (1 - p_V(v_i)) \sum_{m \neq i} \frac{p_V(v_m)}{1 - p_V(v_i)} \phi(v_m). \quad (\text{A.5})$$

It is easy to see that the sum in equation (A.5) is equal to $\mathcal{E}[\phi(V) \mid V \neq v_i]$. By induction we have $\mathcal{E}[\phi(V) \mid V \neq v_i] \leq \phi(\mathcal{E}[V \mid V \neq v_i])$ and

$$\mathcal{E}[\phi(V)] = p_V(v_i)\phi(v_i) + (1 - p_V(v_i)) \sum_{m \neq i} \frac{p_V(v_m)}{1 - p_V(v_i)} \phi(v_m) \quad (\text{A.6})$$

$$\leq p_V(v_i)\phi(v_i) + (1 - p_V(v_i))\phi\left(\sum_{m \neq i} \frac{p_V(v_m)}{1 - p_V(v_i)}\right) \quad (\text{A.7})$$

$$\leq \phi\left(p_V(v_i)\phi(v_i) + (1 - p_V(v_i)) \sum_{m \neq i} \frac{p_V(v_m)}{1 - p_V(v_i)}\right) \quad (\text{A.8})$$

$$= \phi\left(\sum_{m=1}^M p_V(v_m)\right) = \phi(\mathcal{E}[V]), \quad (\text{A.9})$$

where equation (A.7) follows from the induction step for $|\mathcal{V}| = M - 1$, and equation (A.8) follows from the induction step for $|\mathcal{V}| = 2$.

If $\phi(\cdot)$ is strictly concave, the only way we can get equalities in the derivation above is to make sure that $p_V(v_i) = 0$ or $p_V(v_i) = 1$, and furthermore, conditioned on $V \neq v_i$, V is deterministic. These conditions are satisfied if and only if V is deterministic. \square

Definition: Gibbs' Inequality

Let V be a random variable distributed according to distribution $p_V(\cdot)$. Then for any distribution $q_V(\cdot)$,

$$\mathcal{E} [\log p_V(v) \mid p_V(v)] \geq \mathcal{E} [\log q_V(v) \mid p_V(v)], \quad (\text{A.10})$$

with equality if and only if $q \equiv p$.

Before proving this result, we note that we use the notation $\mathcal{E}[\cdot \mid p_V(v)]$ to emphasize the distribution with respect to which the expectation is being taken.

Proof

By concavity of the logarithm,

$$\mathcal{E} [\log q_V(v) \mid p_V(v)] - \mathcal{E} [\log p_V(v) \mid p_V(v)] = \mathcal{E} \left[\log \frac{q_V(v)}{p_V(v)} \mid p_V(v) \right] \quad (\text{A.11})$$

$$\leq \log \mathcal{E} \left[\frac{q_V(v)}{p_V(v)} \mid p_V(v) \right] \quad (\text{A.12})$$

$$= \log \left(\sum_V p_V(v) \cdot \frac{q_V(v)}{p_V(v)} \right) \quad (\text{A.13})$$

$$= 0, \quad (\text{A.14})$$

where the inequality in equation (A.12) follows from Jensen's inequality. \square

Appendix B

The covariance matrix of a multivariate Gaussian mixture model

We obtain the (full) covariance matrix of a Gaussian mixture model by applying the Law of Total Variance, defined as follows (see e.g. Bertsekas and Tsitsiklis (2008)):

$$\text{var}(\mathbf{X}) = \mathcal{E} [\text{var}(\mathbf{X}|\mathbf{Y})] + \text{var}(\mathcal{E} [\mathbf{X}|\mathbf{Y}]). \quad (\text{B.1})$$

For the Gaussian mixture model

$$p_{\mathbf{X}}(\mathbf{x}) = \sum_{j=1}^M \pi_j \times \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j, \mathbf{P}_j), \quad (\text{B.2})$$

let us define the discrete random variable, I , with probability mass function

$$\Pr(I = i) = \pi_i. \quad (\text{B.3})$$

With this, we may write using the Law of Total Probability:

$$p_{\mathbf{X}}(\mathbf{x}) = \sum_{j=1}^M p_{\mathbf{X}|I}(\mathbf{x}|I = j) \Pr(I = j), \quad (\text{B.4})$$

where

$$p_{\mathbf{X}|I}(\mathbf{x}|I=j) = \mathcal{N}(\mathbf{x}; \bar{\mathbf{x}}_j, \mathbf{P}_j). \quad (\text{B.5})$$

We therefore have:

$$\text{var}(\mathbf{X}|I) = \mathbf{P}_I \quad (\text{B.6})$$

$$\mathcal{E}[\mathbf{X}|I] = \bar{\mathbf{x}}_I, \quad (\text{B.7})$$

and thus, by the Law of Total Variance:

$$\text{var}(\mathbf{X}) = \mathcal{E}[\text{var}(\mathbf{X}|I)] + \text{var}(\mathcal{E}[\mathbf{X}|I]) \quad (\text{B.8})$$

$$= \mathcal{E}[\mathbf{P}_I] + \text{var}(\bar{\mathbf{x}}_I) \quad (\text{B.9})$$

$$= \sum_{j=1}^M \pi_j \mathbf{P}_j + \sum_{j=1}^M \pi_j (\bar{\mathbf{x}}_j - \bar{\mathbf{x}})(\bar{\mathbf{x}}_j - \bar{\mathbf{x}})^T, \quad \bar{\mathbf{x}} = \sum_{j=1}^M \pi_j \bar{\mathbf{x}}_j. \quad (\text{B.10})$$

Appendix C

Maximum Entropy Filter

In this section, we describe the Maximum Entropy filter (Kim et al. (2003) and Eyink and Kim (2006)), developed in the Ph.D. thesis by Sangil Kim (2005).

The motivation behind the author's work lay in extending the framework of the Ensemble Kalman filter to handle far-from-Gaussian distributions. As with the Ensemble Kalman filter, the Maximum Entropy filter operates on an ensemble of particles. It is *only* applicable to cases in which a climatological distribution for the system exists, is known, and further can be well approximated by a (semi-)parametric distribution that allows for tractable Bayesian updates.

For simplicity of analysis, in what follows we will restrict our attention to univariate distributions. We note, however, that the analysis generically extends to the multivariate case.

C.1 Formulation

We assume that our system is defined such that a stationary distribution exists and is known. While the method holds for arbitrary distributions, for the purposes of this thesis we further model it as a Gaussian mixture of complexity M :

$$q_X(x) = \sum_{m=1}^M w_m \mathcal{N}(x; \mu_m, \sigma_m^2). \quad (\text{C.1})$$

For a system modeled as a non-periodic Markov Chain with a single recurrent class, it can be shown that any distribution, $p_X(\cdot)$, forced under the transition kernel converges to the stationary distribution of the system, $q_X(\cdot)$ (Cover and Thomas, 2006). We write this, here, as:

$$\lim_{k \rightarrow \infty} D_X(p^k \parallel q) = 0, \quad (\text{C.2})$$

where k is a discrete time index, and $D_X(p \parallel q)$ denotes the Kullback-Leibler divergence (Kullback, 1968) between distributions $p_X(\cdot)$ and $q_X(\cdot)$:

$$D_X(p \parallel q) = \int_{\mathcal{X}} p_X(x) \log \frac{p_X(x)}{q_X(x)} dx. \quad (\text{C.3})$$

Based on this observation, leading up to a Bayesian update we choose to model the prior probability distribution of the system as an *information projection*,

$$\hat{p}_X^k(\cdot) = \underset{p \in \mathcal{S}_k}{\operatorname{argmin}} D_X(p \parallel q), \quad (\text{C.4})$$

with \mathcal{S}_k denoting a chosen set of distributions consistent with particle moment constraints. Qualitatively, we understand (C.4) as finding the distribution, $p_X(\cdot)$, satisfying the moment constraints given by \mathcal{S}_k , that is "closest" to the climatological distribution, $q_X(\cdot)$, having chosen the Kullback-Leibler divergence as the appropriate measure of distance. We adopt the hat notation on the probability density function, \hat{p} , to remind the reader that it has arisen through an information projection.

For the purposes of tractability, we will concern ourselves only with the first and second moments, i.e.

$$\mathcal{S}_k = \{p_X(\cdot) : \mathcal{E}[X \mid p_X(\cdot)] = \bar{x}_k, \operatorname{var}(X \mid p_X(\cdot)) = s_k^2\}, \quad (\text{C.5})$$

although, the analysis holds for arbitrary constraints. We note that \bar{x}_k and s_k^2 refer to the sample mean and variance, respectively, at discrete time k . When limiting our attention to the first two moments, we will show that the prior distribution, too,

takes the form of a Gaussian mixture.

With \mathcal{S}_k defined as in (C.5), it can be shown that $\hat{p}_X^k(x)$ is a member of the following exponential family (Wornell, 2010):

$$\hat{p}_X^k(x) = q_X(x) \frac{e^{\lambda_1 x + \lambda_2 x^2}}{Z(\lambda_1, \lambda_2)}, \quad (\text{C.6})$$

with λ_1 and λ_2 chosen such that (C.5) is satisfied (i.e. $\lambda_1 = \lambda_1(\bar{x}_k, s_k^2)$ and $\lambda_2 = \lambda_2(\bar{x}_k, s_k^2)$), and where $Z(\lambda_1, \lambda_2)$ is the partition function. With this, we will show that $\hat{p}_X(\cdot)$ (having dropped the explicit notation of time with the understanding the the update occurs at discrete time k) takes the form of a Gaussian mixture distribution. We write:

$$\begin{aligned} \hat{p}_X(x) &= q_X(x) \frac{e^{\lambda_1 x + \lambda_2 x^2}}{Z(\lambda_1, \lambda_2)} \\ &= \left(\sum_{m=1}^M w_m \mathcal{N}(x; \mu_m, \sigma_m^2) \right) \frac{e^{\lambda_1 x + \lambda_2 x^2}}{Z(\lambda_1, \lambda_2)} \\ &= \left(\sum_{m=1}^M \frac{w_m}{\sqrt{2\pi\sigma_m^2}} e^{-\frac{(x-\mu_m)^2}{2\sigma_m^2}} \right) \frac{e^{\lambda_1 x + \lambda_2 x^2}}{Z(\lambda_1, \lambda_2)} \\ &= \frac{1}{Z(\lambda_1, \lambda_2)} \left(\sum_{m=1}^M \frac{w_m}{\sqrt{2\pi\sigma_m^2}} e^{-\frac{(x-\mu_m)^2}{2\sigma_m^2} + \lambda_1 x + \lambda_2 x^2} \right) \\ &= \frac{1}{Z(\lambda_1, \lambda_2)} \left(\sum_{m=1}^M \frac{w_m}{\sqrt{2\pi\sigma_m^2}} e^{-\frac{1}{2\sigma_m^2} \left((1-2\sigma_m^2\lambda_2) \left(x - \frac{\mu_m + \sigma_m^2\lambda_1}{1-2\sigma_m^2\lambda_2} \right)^2 - \frac{(\mu_m + \sigma_m^2\lambda_1)^2}{1-2\sigma_m^2\lambda_2} + \mu_m^2 \right)} \right) \end{aligned}$$

by completing the square,

$$= \sum_{m=1}^M \hat{w}_m \mathcal{N}(x; \hat{\mu}_m, \hat{\sigma}_m^2)$$

with

$$\begin{aligned}
\hat{w}_m &= \frac{w_m}{Z(\lambda_1, \lambda_2) \sqrt{1 - 2\sigma_m^2 \lambda_2}} e^{-\frac{1}{2\sigma_m^2} \left(\mu_m^2 - \frac{(\mu_m + \sigma_m^2 \lambda_1)^2}{1 - 2\sigma_m^2 \lambda_2} \right)} \\
\hat{\mu}_m &= \frac{\mu_m + \sigma_m^2 \lambda_1}{1 - 2\sigma_m^2 \lambda_2} \\
\hat{\sigma}_m^2 &= \frac{\sigma_m^2}{1 - 2\sigma_m^2 \lambda_2}.
\end{aligned} \tag{C.7}$$

Having determined the prior distribution (here, left as a function of $\lambda_1(\bar{x}, s^2)$ and $\lambda_2(\bar{x}, s^2)$), we proceed to evaluate the Bayesian update:

$$p_{X|Y}(x|y) = \frac{p_{Y|X}(y|x)p_X(x)}{p_Y(y)}. \tag{C.8}$$

But for a Gaussian observation model,

$$p_{Y|X}(y|x) = \mathcal{N}(y; x, \sigma_o^2), \tag{C.9}$$

we have already shown that this again takes the form of a Gaussian mixture. Using (3.4) - (3.7), we therefore have for the posterior distribution:

$$p_{X|Y}(x|y) = \sum_{m=1}^M \tilde{w}_m \mathcal{N}(x; \tilde{\mu}_m, \tilde{\sigma}_m^2), \tag{C.10}$$

where

$$\begin{aligned}
\tilde{w}_m &= \frac{\hat{w}_m \mathcal{N}(y; \hat{\mu}_m, \sigma_o^2 + \hat{\sigma}_m^2)}{\sum_{i=1}^M \hat{w}_i \mathcal{N}(y; \hat{\mu}_i, \sigma_o^2 + \hat{\sigma}_i^2)} \\
\tilde{\mu}_m &= \hat{\mu}_m + \frac{\hat{\sigma}_m^2}{\sigma_o^2 + \hat{\sigma}_m^2} (y - \hat{\mu}_m) \\
\tilde{\sigma}_m^2 &= \frac{\hat{\sigma}_m^2 \sigma_o^2}{\hat{\sigma}_m^2 + \sigma_o^2}.
\end{aligned} \tag{C.11}$$

At this point, we generate a new set of particles from the updated Gaussian mixture model and evolve these in time using the governing equation for the system. This completes the details of the Maximum Entropy filter. In what follows, we apply

it to the Double Well Diffusion Experiment.

C.2 Double Well Diffusion Experiment

By symmetry of the Double Well Diffusion Experiment, we have already noted that the parameters for the climatological distribution, modeled as a Gaussian mixture of complexity two, takes the form:

$$\begin{aligned} w_1 &= w_2 = 0.5 \\ -\mu_1 &= \mu_2 = \mu \\ \sigma_1^2 &= \sigma_2^2 = \sigma^2. \end{aligned}$$

At any discrete time, k , the prior distribution thus takes the slightly simplified form:

$$\hat{p}_X^k(x) = \sum_{m=1}^M \hat{w}_m \mathcal{N}(x; \hat{\mu}_m, \hat{\sigma}_m^2)$$

with parameters

$$\begin{aligned} \hat{w}_1 &= \frac{0.5}{Z(\lambda_1, \lambda_2) \sqrt{1 - 2\sigma^2 \lambda_2}} e^{-\frac{1}{2\sigma^2} \left(\mu^2 - \frac{(\mu - \sigma^2 \lambda_1)^2}{1 - 2\sigma^2 \lambda_2} \right)} \\ \hat{w}_2 &= \frac{0.5}{Z(\lambda_1, \lambda_2) \sqrt{1 - 2\sigma^2 \lambda_2}} e^{-\frac{1}{2\sigma^2} \left(\mu^2 - \frac{(\mu + \sigma^2 \lambda_1)^2}{1 - 2\sigma^2 \lambda_2} \right)} \\ \hat{\mu}_1 &= \frac{\sigma^2 \lambda_1 - \mu}{1 - 2\sigma^2 \lambda_2} \\ \hat{\mu}_2 &= \frac{\sigma^2 \lambda_1 + \mu}{1 - 2\sigma^2 \lambda_2} \\ \hat{\sigma}_1^2 &= \hat{\sigma}_2^2 = \hat{\sigma}^2 = \frac{\sigma^2}{1 - 2\sigma^2 \lambda_2}. \end{aligned}$$

By normalization, $\hat{w}_1 + \hat{w}_2 = 1$, giving for the partition function:

$$Z(\lambda_1, \lambda_2) = \frac{0.5}{\sqrt{1-2\sigma^2\lambda_2}} \left(e^{-\frac{1}{2\sigma^2} \left(\mu^2 - \frac{(\mu-\sigma^2\lambda_1)^2}{1-2\sigma^2\lambda_2} \right)} + e^{-\frac{1}{2\sigma^2} \left(\mu^2 - \frac{(\mu+\sigma^2\lambda_1)^2}{1-2\sigma^2\lambda_2} \right)} \right) \quad (\text{C.12})$$

$$= \frac{0.5}{\sqrt{1-2\sigma^2\lambda_2}} e^{-\frac{\mu^2}{2\sigma^2}} \left(e^{\frac{(\mu-\sigma^2\lambda_1)^2}{2\sigma^2(1-2\sigma^2\lambda_2)}} + e^{\frac{(\mu+\sigma^2\lambda_1)^2}{2\sigma^2(1-2\sigma^2\lambda_2)}} \right), \quad (\text{C.13})$$

such that the prior distribution equivalently writes:

$$\hat{p}_X^k(x) = \frac{\mathcal{N}(x; \hat{\mu}_1, \hat{\sigma}^2) + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}} \mathcal{N}(x; \hat{\mu}_2, \hat{\sigma}^2)}{1 + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}}. \quad (\text{C.14})$$

Now, by applying the moment constraints of (C.5), we have:

$$\begin{aligned} \mathcal{E}[X | \hat{p}_X^k(\cdot)] &= \hat{w}_1 \hat{\mu}_1 + \hat{w}_2 \hat{\mu}_2 \\ &= \frac{\frac{\sigma^2\lambda_1 - \mu}{1-2\sigma^2\lambda_2} + \frac{\sigma^2\lambda_1 + \mu}{1-2\sigma^2\lambda_2} e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}}{1 + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}} \\ &= \frac{\sigma^2\lambda_1}{1-2\sigma^2\lambda_2} - \frac{\mu}{1-2\sigma^2\lambda_2} \left(\frac{1 - e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}}{1 + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}} \right) \\ &= \bar{x}_k. \end{aligned} \quad (\text{C.15})$$

By the Law of Total Variance:

$$\begin{aligned} \text{var}(X | \hat{p}_X^k(x)) &= \mathcal{E}[\text{var}(X|I)] + \text{var}(\mathcal{E}[X|I]) \\ &= \sum_{i=1}^2 \hat{w}_i \hat{\sigma}_i^2 + \sum_{i=1}^2 \hat{w}_i (\hat{\mu}_i - \bar{x}_k)^2 \\ &= \hat{\sigma}^2 + \hat{w}_1 (\hat{\mu}_1 - \bar{x}_k)^2 + \hat{w}_2 (\hat{\mu}_2 - \bar{x}_k)^2 \\ &= \hat{\sigma}^2 + \hat{w}_1 \hat{\mu}_1^2 + \hat{w}_2 \hat{\mu}_2^2 - 2\bar{x}_k (\hat{w}_1 \hat{\mu}_1 + \hat{w}_2 \hat{\mu}_2) + \bar{x}_k^2 \\ &= \hat{\sigma}^2 + \hat{w}_1 \hat{\mu}_1^2 + \hat{w}_2 \hat{\mu}_2^2 - \bar{x}_k^2 \\ &= \frac{\sigma^2}{1-2\sigma^2\lambda_2} + \frac{\sigma^4\lambda_1^2 + \mu^2}{(1-2\sigma^2\lambda_2)^2} - \frac{2\sigma^2\lambda_1\mu}{(1-2\sigma^2\lambda_2)^2} \left(\frac{1 - e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}}{1 + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}} \right) - \bar{x}_k^2 \\ &= s_k^2. \end{aligned}$$

We thus have the following set of nonlinear simultaneous equations for $\lambda_1(\bar{x}_k, s_k^2)$ and $\lambda_2(\bar{x}_k, s_k^2)$, whose roots may be found by any means available (we used the ‘fsolve’ function in Matlab):

$$\begin{aligned} \frac{\sigma^2 \lambda_1}{1 - 2\sigma^2 \lambda_2} - \frac{\mu}{1 - 2\sigma^2 \lambda_2} \left(\frac{1 - e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}}{1 + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}} \right) &= \bar{x}_k \\ \frac{\sigma^2 - 2\sigma^4 \lambda_2 + \sigma^4 \lambda_1^2 + \mu^2}{(1 - 2\sigma^2 \lambda_2)^2} - \frac{2\sigma^2 \lambda_1 \mu}{(1 - 2\sigma^2 \lambda_2)^2} \left(\frac{1 - e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}}{1 + e^{\frac{2\mu\lambda_1}{1-2\sigma^2\lambda_2}}} \right) &= s_k^2 + \bar{x}_k^2 \end{aligned}$$

With this, using (C.11), we proceed with the Bayesian update to retrieve the posterior distribution.

Appendix D

Sudden Expansion Inlet Velocity Profile

The steady, fully developed, planar Navier-Stokes equations reduce to the following familiar expression:

$$\frac{d^2u}{dy^2} = -\frac{1}{\mu} \frac{dp}{dx}. \quad (\text{D.1})$$

By integrating in y , we obtain

$$\frac{du}{dy} = -\frac{y}{\mu} \frac{dp}{dx} + A \quad (\text{D.2})$$

$$u(y) = -\frac{y^2}{2\mu} \frac{dp}{dx} + Ay + B \quad (\text{D.3})$$

where A and B are integration constants. By applying the boundary conditions associated with the sudden expansion fluid flow, we have:

$$\left. \frac{du}{dy} \right|_{y=0} = 0 \quad \text{by symmetry} \rightarrow A = 0 \quad (\text{D.4})$$

and

$$u(h/2) = 0 \rightarrow B = \frac{h^2}{8\mu} \frac{dp}{dx} \quad (\text{D.5})$$

We therefore obtain the familiar parabolic velocity profile

$$u(y) = \frac{1}{2\mu} \frac{dp}{dx} \left(\frac{h^2}{4} - y^2 \right) \quad (\text{D.6})$$

Now, by conservation of mass, we require that the mass flux due to the developed profile equals that due to the inlet conditions,

$$Q_{in} = hU_{in} = \frac{1}{3} \times 1 = \frac{1}{3}. \quad (\text{D.7})$$

Therefore, mathematically we require

$$\begin{aligned} Q_{out} &= \int_{-h/2}^{h/2} u(y) dy \\ &= \int_{-h/2}^{h/2} \frac{1}{2\mu} \frac{dp}{dx} \left(\frac{h^2}{4} - y^2 \right) dy \\ &= \frac{1}{2\mu} \frac{dp}{dx} \left[\frac{yh^2}{4} - \frac{y^3}{3} \right]_{-h/2}^{h/2} \\ &= \frac{h^3}{12\mu} \frac{dp}{dx} \\ &= Q_{in} \end{aligned} \quad (\text{D.8})$$

giving

$$\frac{dp}{dx} = \frac{4\mu}{h^3} \quad (\text{D.9})$$

With this, we obtain the final expression for the velocity profile at the expansion:

$$U(y) = \frac{2}{h^3} \left(\frac{h^2}{4} - y^2 \right). \quad (\text{D.10})$$

Bibliography

- Alspach, D. L. and Sorenson, H. W. (1972). Nonlinear bayesian estimation using gaussian sum approximations. *Ieee Transactions on Automatic Control*, AC17(4):439–.
- Anderson, J. L. and Anderson, S. L. (1999). A monte carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. *Monthly Weather Review*, 127(12):2741–2758.
- Asada, H. (2011). 2.160 identification, estimation and learning. 2.160 MIT class notes.
- Bengtsson, L., Ghil, M., and Källén, E. (1981). *Dynamic Meteorology: Data Assimilation Methods*. Springer-Verlag.
- Bengtsson, T., Snyder, C., and Nychka, D. (2003). Toward a nonlinear ensemble filter for high-dimensional systems. *Journal of Geophysical Research-Atmospheres*, 108(D24).
- Bertsekas, D. P. and Tsitsiklis, J. N. (2008). *Introduction to Probability*. Athena Scientific, second edition.
- Casella, G. and Berger, R. L. (2001). *Statistical Inference*. Duxbury.
- Cherdron, W., Durst, F., and Whitelaw, J. H. (1978). Asymmetric flows and instabilities in symmetric ducts with sudden expansions. *J. Fluid Mech.*, 84(1):13–31.
- Cover, T. M. and Thomas, J. A. (2006). *Elements of information theory*. Wiley-Interscience, New York, NY, USA.
- Cushman-Roisin, B. and Beckers, J.-M. (2007). *Introduction to Geophysical Fluid Dynamics - Physical and numerics aspects*. Academic Press.
- Doucet, A., Freitas, N. D., and Gordon, N. J. (2001). *Sequential Monte Carlo Methods in Practice*. Springer.
- Dovera, L. and Rossa, E. D. (2010). Multimodal ensemble kalman filtering using gaussian mixture models. *Computational Geosciences*, pages 1–17.
- Durst, F., Melling, A., and Whitelaw, J. H. (1973). Low reynolds number flow over a plane symmetric sudden expansion. *J. Fluid Mech.*, 64:111–128.

- Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using monte-carlo methods to forecast error statistics. *Journal of Geophysical Research-Oceans*, 99(C5):10143–10162.
- Eyink, G. L. and Kim, S. (2006). A maximum entropy method for particle filtering. *Journal of Statistical Physics*, 123(5):1071–1128.
- Fearn, R. M., Mullin, T., and Cliffe, K. A. (1990). Nonlinear flow phenomena in a symmetric sudden expansion. *J. Fluid Mech.*, 211:595–608.
- Frey, B. J. and Dueck, D. (2007). Clustering by passing messages between data points. *Science*, 315:972–976. www.psi.toronto.edu/affinitypropagation.
- Gelb, A. (1974). *Applied optimal estimation*. MIT Press.
- Haley, P. J. and Lermusiaux, P. F. J. (2010). Multiscale two-way embedding schemes for free-surface primitive-equations in the multidisciplinary simulation, estimation and assimilation system. *Ocean Dynamics*, 60:1497–1537.
- Heubel, E. V. (2008). Parameter estimation and adaptive modeling studies in ocean mixing. Master’s thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering.
- Higham, D. J. (2001). An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM review*, 43(3):525–546.
- Hoteit, I., Pham, D. T., Triantafyllou, G., and Korres, G. (2007). A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography. *Monthly Weather Review*, 136(1):317–334.
- Jaakkola, T. (2006). 6.867 machine learning. Course material for 6.867 Machine Learning, Fall 2006. MIT OpenCourseWare, Massachusetts Institute of Technology. Downloaded on [25 02 2011]. <http://ocw.mit.edu/>.
- Jazwinski, A. H. (1970). *Stochastic Processes and Filtering Theory*. Academic Press.
- Julier, S. J. and Uhlmann, J. K. (1997). A new extension of the kalman filter to nonlinear systems. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 3068 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 182–193.
- Kalman, R. E. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45.
- Kalnay, E. (2003). Atmospheric modeling, data assimilation and predictability.
- Kim, S. (2005). *Ensemble filtering methods for nonlinear dynamics*. PhD thesis, The University of Arizona, Department of Applied Mathematics.

- Kiml, S., Eyink, G. L., Restrepo, J. M., Alexander, F. J., and Johnson, G. (2003). Ensemble filtering for nonlinear dynamics. *Monthly Weather Review*, 131(11):2586–2594.
- Kullback, S. (1968). *Information Theory and Statistics*. Dover Publications, Inc.
- Kundu, P. and Cohen, I. (2008). *Fluid Mechanics*. Academic Press, 4 edition.
- Lermusiaux, P. F. J. (1997). *Data Assimilation via Error Subspace Statistical Estimation*. PhD thesis, Harvard University, Department of Science.
- Lermusiaux, P. F. J. (1999). Estimation and study of mesoscale variability in the strait of sicily. *Dynamics of Atmospheres and Oceans*, 29:255–303.
- Lermusiaux, P. F. J. (2006). Uncertainty estimation and prediction for interdisciplinary ocean dynamics. *Journal of Computational Physics*, 29:176–199.
- Lermusiaux, P. F. J., Chiu, C. S., Gawarkiewicz, G. G., Abbot, P., Robinson, A. R., Miller, R. N., Haley, P. J., Leslie, W. G., Majumdar, S. J., Pang, A., and Lekien, F. (2006). Quantifying uncertainties in ocean predictions. *Oceanography*, 19:92–105.
- Lorenz, E. (1963). Deterministic nonperiodic flow. *J. Atmos. Sci.*, 20:130–141.
- MacKay, D. J. C. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.
- Marzouk, Y. M. and Wang, Q. (2010). 16.949 numerical methods for spdes. 16.949 MIT class notes.
- McLachlan, G. and Peel, D. (2000). *Finite Mixture Models*. John Wiley & Sons, Inc.
- McLachlan, G. J. and Basford, K. E. (1988). *Mixture Models: Inference and applications to clustering*. Marcel Dekker, Inc.
- McLachlan, G. J. and Krishnan, T. (1997). *The EM algorithm and extensions*. John Wiley & Sons, Inc.
- Miller, A. J., Chai, F., Chiba, S., Moisan, J. R., and Neilson, D. J. (2004). Decadal-scale climate and ecosystem interactions in the north pacific ocean. *Journal of Oceanography*, 60:163–188.
- Miller, R. N., Ghil, M., and Gauthiez, F. (1994). Advanced data assimilation in strongly nonlinear dynamical-systems. *Journal of the Atmospheric Sciences*, 51(8):1037–1056.
- NASA (2008). Nasa ames research center, moffett field, calif., history related to the apollo moon program and lunar prospector mission.
- Petersen, K. B. and Pedersen, M. S. (2008). The matrix cookbook. Version 20081110.

- Sapsis, T. (2010). *Dynamically Orthogonal field equations*. PhD thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering.
- Sapsis, T. and Lermusiaux, P. F. J. (2009). Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D*, 238:2347–2360.
- Sapsis, T. and Lermusiaux, P. F. J. (2010). Dynamical criteria for the evolution of the stochastic dimensionality in flows with uncertainty. Submitted.
- Sekine, Y. (1990). A numerical experiment on the path dynamics of the kuroshio with reference to the formation of the large meander path south of japan. *Deep-Sea Research Part a-Oceanographic Research Papers*, 37(3):359–380.
- Silverman, B. (1992). *Density Estimation for Statistics and Data Analysis*. Chapman & Hall.
- Smith, K. W. (2007). Cluster ensemble kalman filter. *Tellus Series a-Dynamic Meteorology and Oceanography*, 59:749–757.
- Ueckermann, M. P., Sapsis, T. P., and Lermusiaux, P. F. J. (2011). Dynamically orthogonal navier-stokes equations for stochastic fluid flows: An efficient finite volume scheme.
- van Leer, B. (1977). Towards the ultimate conservative difference scheme iii. upstream-centered finite-difference schemes for ideal compressible flow. *J. Comp. Phys.*, 23:263–275.
- van Leeuwen, P. J. (2009). Particle filtering in geophysical systems. *Monthly Weather Review*, 137(12):4089–4114.
- Wornell, G. (2010). 6.437 inference and information. 6.437 MIT class notes.