Study of variational ensemble methods for image assimilation

Yin Yang

ANNÉE 2015

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Traitement du signal et télécommunications

École doctorale Matisse

présentée par

Yin YANG

préparée à l'unité de recherche Fluminance
Centre INRIA Rennes- Bretagne Atlantique
Université Rennes 1

Study of Variational Ensemble Methods for Image Assimilation

Thèse soutenu à Rennes
le 16 décembre 2014

devant le jury composé de :

Olivier PANNEKOUCKE
Chargé de recherche, Météo France /*Rapporteur*
Arthur VIDARD
Chargé de recherche, Inria /*Rapporteur*
Marc BOCQUET
Professeur, Ecole des Ponts ParisTech /*Examinateur*
Thomas CORPETTI
Directeur de recherche, CNRS /*Examinateur*
Dominique HEITZ
Directeur de recherche, Irstea /*Examinateur*
Etienne MEMIN
Directeur de recherche, Inria /*Directeur de thèse*

I could be bounded in a nutshell and count myself King at infinite space, were it not that I have bad dreams.

Hamlet, II, 2

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisor, Etienne, for his perfect mix of amicable advice and rigorous standards during the course of this research. It has been a privilege working with Etienne and his devotion to academia research will always be a precious example to me.

I would also like to thank all the members of the jury for their constructive evaluation of my work, especially to the rapporteurs, Olivier Pannekoucke and Arthur Vidard, for their attentive review of my manuscript.

I also would like to thank all the members of Fluimance group: Benoît, Christophe, Ioana, Kai, Pierre, Quy, Sébastien, Tudor, Valentin, Véronique, Cédric et Patrick. Special thanks to Cordelia with whom I have enjoyed many useful and entertaining discussions. Special thanks also to Dominique for his help and support throughout my doctor program as well as many insightful advice in between. Thanks Huguette for her administration support over the years.

A special mention to Christoph Heinzel for the detailed language polishing of my manuscript.

Finally I would like to thank all my friends and family. Especially Yu, for the happiest time we have had, and will have together.

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

$\boldsymbol{a}$	Quadratic variation tensor
$\mathbf{A}'$	Ensemble pertubation matrix
$\boldsymbol{\sigma}$	Diffusion tensor
$\partial_{\mathbf{x}}\mathbb{H}^*$	Adjoint linear observation operator w.r.t. $\partial_{\mathbf{x}}\mathbb{H}(\mathbf{x})$
$\partial_{\mathbf{x}}\mathbb{M}(\mathbf{x})$	Tangent linear dynamic model linearized around state $\mathbf{x}$
$\partial_{\mathbf{x}}\mathbb{M}^*$	Adjoint linear dynamic model w.r.t. $\partial_{\mathbf{x}}\mathbb{M}(\mathbf{x})$
$\partial_{\mathbf{x}}\varphi_t(\mathbf{x})$	Tangent linear flow map linearized around state $\mathbf{x}$
$\partial_{\mathbf{x}}\varphi_t^*$	Adjoint linear flow map operator w.r.t. $\partial_{\mathbf{x}}\varphi_t$
$\partial_{\mathbf{x}}\mathbb{H}(\mathbf{x})$	Tangent linear observation operator linearized around state $\mathbf{x}$
$\boldsymbol{\epsilon}$	Observation error term
$\boldsymbol{\eta}$	Background error term
$\mathbb{H}$	Nonlinear observation operator
$\mathbb{M}$	Nonlinear dynamic model
$\mathcal{C}$	Spatial correlation matrix
$\mathbf{B}$	Background error covariance matrix
$\mathbf{D}$	Innovation vector
$\mathbf{H}$	Linear observation operator
$\mathbf{K}$	Kalman gain matrix
$\mathbf{M}_l$	Local transform matrix
$\mathbf{P}^a$	Analysis error covariance matrix
$\mathbf{P}^f$	Forecast error covariance matrix
$\mathbf{q}$	Model error term
$\mathbf{R}$	Observation error covariance matrix
$\mathcal{Y}$	Observation vector
Φ	Linear dynamic discrete operator
$\boldsymbol{U}$	Eulerian velocity field
$\varphi_t(\mathbf{x}_0)$	Flow map at time level t integrated from $\mathbf{x}_0$
$\mathbf{X}$	Ensemble state matrix gathering N samples
$\mathbf{x}$	State variable vector

$\mathbf{X}'$	Ensemble anomaly state matrix
x	Space index
$\boldsymbol{X}$	Fluid flow trajectory map and Lagrangian displacement field
$\mathbf{Z}$	Ensemble of control vector
$\mathbf{z}$	Control vector
m	Dimension of observation space
n	Dimension of state space
t	Time index

Résumé en Français

Context de Travaux

Le principe de l'assimilation de données décrit une méthode d'estimation de l'état vrai initial $\mathbf{x}^t(t_0)$ de système $\mathbf{x}(t,x) \in \mathbb{R}^n$ à partir d'un état initial *a priori* $\mathbf{x}_0^b$, un modèle dynamique $\mathbb{M}$ et des mesures associées au système $\mathcal{Y} \in \mathbb{R}^m$. L'état initial est souvent appelé le nom de l'état d'ébauche ou de prévision en fonction du contexte. L'état initial estimé est appelée l'analyse. Le système d'assimilation est donnée par les trois équations suivantes,

$$\partial_t \mathbf{x}(t,\mathrm{x}) + \mathbb{M}(\mathbf{x}(t,\mathrm{x}),u) = \mathbf{q}(t,\mathrm{x}), \tag{1}$$

$$\mathbf{x}_0(\mathrm{x}) = \mathbf{x}_0^b(\mathrm{x}) + \boldsymbol{\eta}(\mathrm{x}), \tag{2}$$

$$\mathcal{Y}(t,\mathrm{x}) = \mathbb{H}(\mathbf{x}(t,\mathrm{x})) + \boldsymbol{\epsilon}(t,\mathrm{x}). \tag{3}$$

La première équation est le modèle dynamique. Il est tout simplement la forme numérique des lois physiques à prescrire l'évolution du système. L'intégration de l'état initial entre temps initial t_0 et temps finale t_f fournit une trajectoire de variables d'état. L'opérateur dynamique $\mathbb{M}$ est dans le cas général d'un opérateur différentiel non linéaire, qui pourrait également dépendre des paramètres inconnus u. Dans certains cas, les paramètres inconnus peuvent être évaluées de la même manière que l'état initial. Le modèle dynamique est appelée imparfaite, si l'on considère un terme d'erreur du modèle $\mathbf{q}$.

La seconde équation est généralement nécessaire lorsqu'on a connaissance *a priori* de l'état initial. On peut modéliser l'état initial (vrai) comme la somme de l'état initial *a priori* et un bruit $\boldsymbol{\eta} \in \mathbb{R}^n$.

La dernière équation relie les observations, $\mathcal{Y}$, et la variable d'état $\mathbf{x}$ par un opérateur d'observation $\mathbb{H} \in \mathbb{R}^{m\times n}$, plus un terme de bruit $\boldsymbol{\epsilon} \in \mathbb{R}^m$. Le bruit d'observation $\boldsymbol{\epsilon}$ est une erreur additif aléatoire. Remarque que $\boldsymbol{\epsilon}$ est souvent liée à l'erreur instrumentale et en effet inconnu, cependant, on suppose que il est connu *a priori*.

L'idée de l'assimilation de données remonte à la mise en place d'algorithmes d'interpolation au début de la prévision météorologique numérique.

La théorie du contrôle optimal suscite une autre approche, appelée la méthode variationnelle (Var), qui elle-même imposée comme probablement la méthode la plus utilisée de nos jours. L'assimilation variationnelle peut être grossièrement classés en méthodes variationnelle 3D (3DVar) et 4D (4DVar) selon qu'elle considère la fenêtre temporelle et la dynamique du système ou non. Un autre groupe important de ces méthodes, qui comprend le filtre de Kalman comme un élément représentatif, est désigné en tant que méthodes séquentielles. Toutes ces approches ont pour but de corriger l'état d'ébauche, le cas échéant, étant donné les observations.

Ensemble Kalman filter

Le filter Kalman d'ensemble (EnKF) est une formulation de Monte-Carlo du filtre de Kalman standard. Dans ce cadre, on définit la matrice $\mathbf{X}_t = (\mathbf{x}_t^{(1)}, \ldots, \mathbf{x}_t^{(i)}, \ldots, \mathbf{x}_t^{(N)}) \in \mathbb{R}^{n \times N}$ regroupant les échantillons d'ensemble de l'état. Le EnKF est un filtre en deux phases: dans la phase de prévision, un nuage d'états possibles est généré à partir d'une randomisation de la dynamique ou de ses paramètres. Cet ensemble d'échantillons permet de calculer une approximation de rang fiable de la matrice d'erreur de covariance $\mathbf{P}^f$.

$$\mathbf{P}^f \approx \frac{1}{N-1}(\mathbf{X}_t^f - \langle \mathbf{X}_t \rangle)(\mathbf{X}_t^f - \langle \mathbf{X}_t \rangle)^T. \tag{4}$$

Deux types principaux ont été proposés pour la deuxième étape, l'étape d'analyse. La première se repose sur une approche de Monte Carlo directe, qui introduit des échantillons de observation avec bruit (Burgers et al., 1998; van Leeuwen and Evensen, 1996; Houtekamer and Mitchell, 1998). Le second correspond à la technique du filtre racine carrée. (Anderson, 2003; Bishop et al., 2001; Ott et al., 2004; Tippett et al., 2003; Whitaker and Hamill, 2002). Ces derniers régimes évitent les problèmes d'échantillonnage associés à l'ensemble de petite taille. Ce résultat est obtenu en limitant l'analyse dans l'espace engendré par les perturbations d'ensemble de prévisions centrées. Une réalisation possible est l'filtre de Kalman d'Ensemble Transforme(ETKF), proposé initialement par Bishop et al. (2001):

$$\mathbf{A}'^a = \mathbf{A}'^f \mathbf{T}, \tag{5}$$

où $\mathbf{A}'^f = \frac{1}{\sqrt{N-1}}(\mathbf{X}_t^f - \langle \mathbf{X}_t \rangle)^f$ représente la matrice de la racine carrée de la matrice $\mathbf{P}^f$ and $\mathbf{T} \in \mathbb{R}^{N \times N}$.

Variational methods

Le problème de l'assimilation de données dans le cadre des méthodes variationnelles peut être reformulé comme suit: On a l'intention d'estimer le vecteur d'état, $\mathbf{x}$, à partir de la Eq.(3) soumis à la *contrainte* dans la forme d'équation du modèle dynamique d'évolution (1) et l'équation de modélisation de l'état initial (2). Lorsque les opérateurs concernés sont non-linéaires, la procédure d'assimilation variationnelle consiste à effectuer une linéarisation de la dynamique autour d'une trajectoire actuelle et à manœuvrer l'optimisation par rapport à une solution de l'incrémentation. La fonction coût en termes de l'incrémentation $\delta\mathbf{x}_0$ est donc défini comme suit:

$$J(\delta\mathbf{x}_0) = \frac{1}{2}\|\delta\mathbf{x}_0\|^2_{\mathbf{B}} + \frac{1}{2}\int_{t_0}^{t_f} \|\mathbb{H}(\mathbf{x}(\mathrm{x},t)) - \mathcal{Y}(\mathrm{x},t)\|^2_{\mathbf{R}} dt. \tag{6}$$

Habituellement, un système de deux boucles imbriquées est employé pour compte de l'involution des termes d'ébauche par la dynamique non linéaire tout en gardant la simplicité de trouver une incrémentation optimale déterminée par la dynamique linéaire tangent et son adjoint. Une comparaison schématique d'assimilation variationnelle et d'assimilation séquentielle est présentée dans la figure 1.

Méthode Proposée et Applications

Récemment, plusieurs schémas destinées à coupler les avantages des méthodes d'ensemble et les stratégies d'assimilation variationnelle ont été proposés. Un groupe de ces méthodes hybrides conserve le formalisme des procédures d'un algorithme du gradient itératif

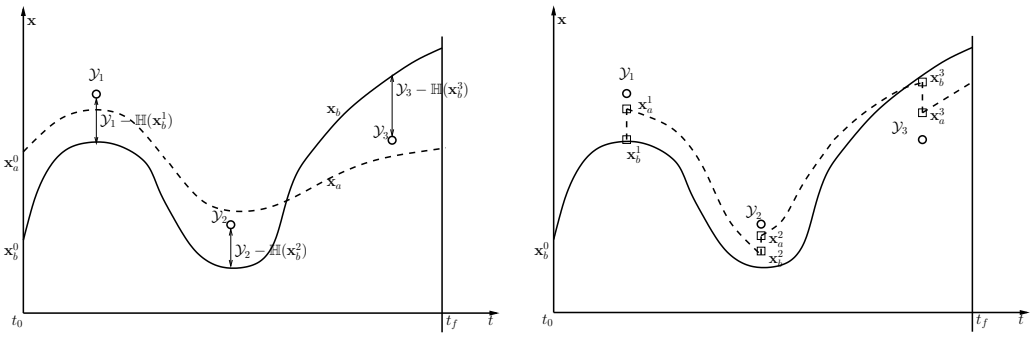


Figure 1 – Scheme comparison between 4DVar and EnKF

de la fonction coût dérivée dans le cadre de méthodes variationnelles. Dans ce groupe, l'algorithme est généralement construit par l'incorporation dans la fonction coût variationnelle d'un covariance d'ébauche d'ensemble(Hamill and Snyder, 2000; Lorenc, 2003; Buehner, 2005; Liu et al., 2008, 2009; Zhang et al., 2009; Buehner et al., 2010a,b; Krysta et al., 2011; Clayton et al., 2012; Fairbairn et al., 2013; Buehner et al., 2013; Desroziers et al., 2014). L'autre groupe des méthodes hybrides conserve le formalisme du filtre de Kalman d'ensemble, et vise en principe à l'assimilation des données asynchrones (Hunt et al., 2004; Zupanski, 2005; Fertig et al., 2007; Hunt et al., 2007; Sakov et al., 2010). Un autre attribut remarquable de ce groupe est que l'étape d'analyse explicite ou de mise à jour en termes de filtre de Kalman tend à être remplacée par une procédure d'un algorithme du gradient itératif de certains fonction coût (Zupanski, 2005; Solonen et al., 2012; Sakov et al., 2012).

L'ossature de cette thèse se repose sur une méthode variationnelle basée sur l'ensemble. Cette méthode tombe dans la catégorie de la méthode hybride dans lequel le but est de bénéficier des avantages des techniques d'assimilation variationnelle (4DVar) et du filtre de Kalman d'ensemble (EnKF) tout en contournant leurs faiblesses. Une description complète de la stratégie proposée est listé dans le chapitre 5 suivie d'une validation sur le modèle de Shallow Water dans le chapitre 6. Notre procédé consiste plusieurs améliorations par rapport aux autres procédés existants. On a proposé un nouveau schéma de la boucle imbriquée dans laquelle la matrice de covariance de erreur d'ébauche est mis à jour pour chaque boucle externe. Notons ici k que l'indice de boucle externe. Au début de la k-ième boucle externe, on a proposé d'intégrer les champs de l'état ensemble entières tout au long de la fenêtre d'assimilation. Et la mise à jour de la matrice de covariance d'erreur d'ébauche est calculée à partir de cet ensemble:

$$\mathbf{X}_t^{b,k+1} = \varphi_t(\mathbf{X}_0^{b,k}), \tag{7}$$

$$\mathbf{A}_b'^{k+1} \approx \frac{1}{\sqrt{N-1}}((\mathbf{x}_0^{(1)b,k+1} - \langle \mathbf{x}_0^{b,k+1} \rangle, \ldots, \mathbf{x}_0^{(N)b,k+1} - \langle \mathbf{x}_0^{b,k+1} \rangle). \tag{8}$$

On a aussi combiné des schémas différents de la mise à jour d'ensemble avec deux régimes de localisation: la covariance localisée et l'ensemble local. Le premier est couplé avec la méthode d'observation perturbée, et la seconde est associé à la méthode de transformation directe. En termes de l'approche de la mise à jour directe, on a exploité les liens entre la matrice de covariance de erreur d'analyse et l'inverse de Hessian de la fonction

coût en 4D, et on a introduit une procédure de minimisation quasi-Newton s'appuyant sur une approximation de l'inverse de Hessian.

$$\mathbf{A}_b'^{k+1} = \mathbf{A}_b'^{k} \mathcal{H}_{\mathbb{I}}^{-\frac{1}{2}}. \tag{9}$$

L'objectif principal de cette thèse également survient dans l'étude des techniques d'assimilation efficaces pour les observations de données d'images. A cette fin, les méthodes d'ensemble proposées ont été évalués sur des données synthétiques et réelles. Leurs performances ont été comparées à une méthode de 4DVar standard et plusieurs méthodes d'ensemble proposés dans la littérature. On a constaté que la méthode d'ensemble constitue une solution efficace pour gérer les données incomplètes, ce qui constitue la situation standard associé à des observations d'image. Avec les données observées partielles, les méthodes d'ensemble surpassent la 4DVar standard en terms de reconstruction de composant non observée. On a également observé que l'ensemble généré par une perturbation de paramètre fournit une dispersion de l'ensemble plus pertinent et permet de mieux rapprocher les statistiques d'erreur d'ébauche lorsque le paramètre d'intérêt est lié à des effets physiques. Le coût du calcul (temps CPU et demandes de mémoire) de méthodes d'ensembles sont considérablement inférieur que la méthode de 4DVar standard si une technique de calcul parallèle approprié est déployée.

Les méthodes proposées ont été également évalué dans le cadre des données d'image expérimentale bruyants d'un écoulement à surface libre fourni par un capteur Kinect (chapitre 7). Ces observations montrent grande région de données manquantes. Nos méthodes donnent de meilleurs résultats pour suivre la hauteur de la surface libre et présenter des avantages dans le traitement des discontinuités. Un opérateur d'image nonlinéaire directe basé sur l'erreur de reconstruction d'image a été évaluée pour ces techniques d'ensemble sur un modèle quasi-géostrophique de surface d'écoulement océaniques. Les méthodes d'ensemble proposées ont montré à constituer techniques intéressantes dans le contexte général d'un opérateur d'observation d'image nonlinéaire.

Approche du Modèle Stochastique

Afin de traiter une forte différence d'échelle entre les données d'image et la résolution de la grille du modèle dynamique, on a exploré la performance d'une représentation du modèle de Shallow Water sous l'incertitude de location. Les équations de la dynamique des écoulements stochastique sont construits sur l'hypothèse que la quantité d'intérêts (masse, quantité de mouvement ou énergie) est transporté par la particule fluide stochastique. La motivation principale est l'assimilation des images à haute résolution dans les modèles dynamiques à grande échelle. Cela constitue une situation standard en géophysique. Ce modèle introduit un modèle de sous-maille encodage des effets des processus physiques observés sur l'observation à haute résolution. Ce modèle permet aussi un coût de calcul moins cher par rapport à RANS ou DNS. On a montré comment estimer le paramètre de modèle sous-maille à partir des données directement et à partir de notre méthode variationnelle basée sur d'ensemble. Les évaluations ont été réalisées avec des données d'images synthétique 1D, synthétique 2D et réelles 2D, respectivement. Les résultats sont encourageants et montrent un grand potentiel pour traiter des données d'image en haute résolution.

Introduction

Thesis topic:

In experimental fluid mechanics, extracting information from observations measured in laboratories has always been a great tool to complement the knowledge from theoretical fluid dynamics, as the latter relies on studying and solving the governing equations (Navier-Stokes equations) derived from the conservation laws. One fruitful result for such information extraction process are numerous empirical formulas inducted from various applications. Those measurements can also be used to validate theories and to reveal the flow phenomenon that the dynamic models fail to predict. With the development of computational fluid dynamics (CFD), especially in the field of Numerical Weather Prediction (NWP) applications, and in order to meet the demand of better predicting atmospheric states at the lowest possible computational cost, an approach to integrate simultaneously the observations into the prediction models has been gradually formulated through the years. This approach bears the name of 'data assimilation' (DA).

Data assimilation (DA) can have various definitions from different perspectives. From the perspective of meteorology or oceanography, Talagrand (1997) states that the assimilation process is "using all the available information, determine as accurately as possible the state of the atmospheric or oceanic flow", or more specifically, Blum et al. (2009) suggest that "the ensemble of techniques which, starting from heterogeneous information, permit to retrieve the initial state of the flow." Wikle and Berliner (2007) interprets this term from a statistical view point as "an approach for fusing data with prior knowledge (*e.g.*, mathematical representations of physical laws; model output) to obtain an estimate of the distribution of the true state of a process."

Although the principle of DA can indeed be applied to many disciplines, it is not a surprise that DA techniques originated from atmospheric science. The reason mainly hinges on the fact that the atmospheric system is highly sensitive to initial conditions (Lorenz, 1963).

A typical DA system is composed of three aspects: the model, the observations, and the assimilation method. The models can be very different depending on the applications field. In the case of the regional ocean modeling system (ROMS), it is a primitive equation model in the field of oceanography. The main variables include but are not limited to velocity, pressure, temperature, salinity, etc. To govern the evolution of these variables, there are the equations of mass balance, momentum balance, energy balance and advective-diffusive. In addition to these equations, we also need the knowledge of air-sea interaction, boundary conditions, horizontal and vertical mixing parameterization and other sources of effects which eventually interfere with the course of flow. The model is rather complex, even though it has been already subject to many simplifications and hypotheses. Within this thesis, I focus on 'toy' models which process the similar characteristic as operational

models used to describe the geophysical flows. Those toy models are computationally cheaper, but remain quite general.

Another aspect of the DA system lay in the available observations. Historically, a large portion of the data corresponds to *in situ* measurements. These observations are heterogeneous and are much scarcer than the number of corresponding model variables (especially in oceanography). This type of data would result in a classical "ill-posed problem" if observations are directly used to estimate the actual state, and thus justifies the introduction of a 'background' state into the DA system. Along with the development of the remote-sensing technology since 1970s, more and more image data is captured by satellite sensors. These image sequences have a much higher spatial and temporal density than other observations. One may think that the assimilation of high-resolution image data would lead to a drastic improvement compared to sparse data. However, this is not the case since there is another aspect to the story. Firstly, we need to face the balance problem between the state variables, since the variables inferred from remote-sensing approaches generally do not respect the equilibrium rules. Besides, more observations lead to a high conditional number for the assimilation cost function, which will eventually devastate the data assimilation process. Furthermore, the image data is only indirectly linked to the state variables, and usually the luminance function can hardly reflect all the state variables present in the dynamic model. These new problems posed by image data imply that the DA method is to be re-examined in this context. Note that the image data assimilation has been a topic drawing more and more attention since the last decades and several strategies have been proposed to deal with these difficulties, (Papadakis and Mémin, 2008; Corpetti et al., 2009; Souopgui, 2010; Titaud et al., 2010; Beyou et al., 2013b).

To sum up, the assimilation technique can be viewed as a learning process of a dynamic model on the basis of the observations available by adjusting the model's parameters (the initial conditions can be viewed as model parameters in this sense).

This thesis focuses on the investigation of assimilation methods and their applications related to data assimilation. A particular interest of this thesis is to explore the approaches dedicated to assimilate image data. To achieve this goal, modifications to the method may be made either to the model, to the extraction of observations, or to the assimilation algorithm itself.

Thesis Outline

Part I Context

Chapter 1: Data assimilation In the first chapter, we introduce briefly the data assimilation problem dedicated to NWP.

Chapter 2 & 3: Variational methods, Sequential methods The second and third chapter contain the basic mathematical aspects of variational methods and sequential methods, respectively. As the central method in this thesis is a hybrid method relying on both variational and Kalman filter methods, it is crucial to give a basic outline of both approaches. We leave a description of the state of the art of the hybrid method to the second part.

Part II Hybrid method

Chapter 4: Hybrid methods review This part starts with a review of the hybrid methods proposed and tested by other researchers so far. The motivation and needs for hybrid methods are thoroughly presented. We also highlight the key differences between these methods provided in the existed literature.

Chapter 5: Ensemble-based 4DVar The algorithms of the ensemble-based 4DVar that we have been working on are detailed in this section. We focus on the enhancements made to the method and highlight the advantages in comparison to other methods. There are three important aspects on which we concentrate:

- The first one concerns the extra outer loop, the background update and the associated background error covariance matrix. In the literature, the extra outer loop is usually considered unnecessary for an ensemble-based method because of the relatively low conditional number of the cost function. However, as some results of this thesis show it, the extra outer loops can significantly improve the results.
- The second point is the update methods of ensemble associated with two consecutive outer loops or two assimilation windows, which can be either stochastic (with perturbed observations) or deterministic. The method of Liu et al. (2008) adopts the former. But the latter is also possible and is used in 4DEnKF proposed by Hunt et al. (2007).
- The third aspect is the localization technique, which can either localize the covariance or use local ensembles.

Part III Application

Chapter 6: Shallow water model verification This chapter is mainly based on the article elaborated with my coworkers C. Robinson, D. Heitz and E. Mémin. Instead of emphasizing on the comparative result with 4DVar, this chapter highlights the properties of the ensemble-based 4DVar. Moreover, different approaches are numerically compared and discussed. Two cases are mainly treated: the Gaussian error case and the slope error case. The Gaussian error case is studied thoroughly with a comparison between various configurations including different ensemble update or localization techniques. The slope error case contains three scenarios: height observations, velocity observations, and complete observations.

Chapter 7: Application with image data This chapter begins as a continuity of the previous chapter. Firstly, we introduce the experimental setup and the parameter configuration. The method showing how the surface height is extracted from the image sequence captured by Kinect sensor is presented. We focus here on the observation treatment, including bad points elimination, blind spot interpolation and potential filtering of the observations. Real world results on experiments raw data are then provided.

This second part of this chapter gives a succinct but complete presentation of the SQG model and of its coupling with the image data. We also introduce the principles of image motion estimation and image observation operators. Finally, the results are provided both in the case of synthetic image data and real world SST image data.

Part IV Stochastic model approach

Chapter 8: Shallow water model under location uncertainty The objective of this chapter consists in assimilating high-resolution image data by processing an analysis on a coarser grid through a dynamics defined from an uncertainty principle where subgrid data are defined as a stochastic process. The model tested in this chapter is introduced in the article of Mémin (2014). We present the complete formulation of the conservative shallow water model under location uncertainty and explain the associated terms, e.g, diffusion terms and stochastic terms.

Chapter 9: Ensemble-based parameter estimation: scheme and results This chapter is devoted to the presentation of an ensemble estimation of uncertainty parameters from the high-resolution image data. This method is then tested with the model described in chapter 8. Both synthetic data and real image data captured from camera Kinect have been assessed.

A general conclusion, in which some perspectives are given, closes this manuscript.

Part I

Context

Chapter 1

Data assimilation

1.1 Introduction

The idea of data assimilation can be traced back to the setup of interpolation algorithms at the inception of numerical weather prediction. Those interpolation algorithms were based on the simple ideas of obtaining the analysis by interpolating the observation to the dynamic numerical grid points. Pioneering works by Bergthorsson and Döös (1955) and Cressman (1959), later combined with a statistical interpretation by Gandin (1965) together give birth to a methodology called optimal interpolation (OI). OI is able to deal with more complex observations and is relatively easy to implement. The optimal control theory shed light on another approach, referred to as the Variational method (Var) which imposed itself as probably the most used method nowadays. Variational assimilation can be roughly categorized into 3D variational (3DVar) and 4D variational (4DVar) methods depending on whether or not a temporal window and the system dynamics are considered in the DA system. It has long been known that the 3DVar approach is equivalent to the OI approach (Lorenc, 1986). Details regarding this method will be elaborated in section 1.4.2. Another prominent group of methods, which includes the Kalman filter as a representative element, is referred to as sequential methods. We will discuss the related topics in section 1.4.1. All these approaches aim to correct the background state, if any, given the observations.

1.2 The assimilation process

This section aims at providing a general mathematic description of the data assimilation problem. As recalled in the introduction, the data assimilation principle depicts a method for estimating the true initial state $\mathbf{x}^t(t_0)$ (with superscript 't' standing for 'true', to be distinguished from the time index) of system $\mathbf{x}(t,x) \in \mathbb{R}^n$ from an *a priori* initial state $\mathbf{x}_0^b$ (the subscript '0' stands for the time instant t_0), a dynamic model $\mathbb{M}$ and the measurements $\mathcal{Y} \in \mathbb{R}^m$ associated with the system. The *a priori* initial state is often called the background state ($\mathbf{x}_0^b$ with superscript 'b') or forecast state ($\mathbf{x}_0^f$ with superscript 'f') depending on the context. The estimated initial state is called the analysis ($\mathbf{x}_0^a$ with superscript 'a'). The

assimilation system is given by the following three equations,

$$\partial_t \mathbf{x}(t, \mathrm{x}) + \mathbb{M}(\mathbf{x}(t, \mathrm{x}), u) = \mathbf{q}(t, \mathrm{x}), \tag{1.1}$$

$$\mathbf{x}_0(\mathrm{x}) = \mathbf{x}_0^b(\mathrm{x}) + \boldsymbol{\eta}(\mathrm{x}), \tag{1.2}$$

$$\mathcal{Y}(t, \mathrm{x}) = \mathbb{H}(\mathbf{x}(t, \mathrm{x})) + \boldsymbol{\epsilon}(t, \mathrm{x}). \tag{1.3}$$

The first equation is the dynamic model. It is simply the numerical form of the physical laws prescribing the system evolution. The integration of the initial state from time t_0 to time t_f provides a trajectory of the state variables. The dynamic operator $\mathbb{M}$ is in the general case a nonlinear differential operator that could also depend on some unknown parameter u. In certain cases, the unknown parameters can be estimated in the same way as the initial state. The dynamic model is called imperfect, if we consider a model error term $\mathbf{q}$.

The second equation is usually necessary when we have *a priori* knowledge of the initial state $\mathbf{x}_0^b$. It is usually referred to as the background state. We can model the initial state, $\mathbf{x}(t_0, x)$, as the sum of the *a priori* initial state and some noise $\boldsymbol{\eta} \in \mathbb{R}^n$.

The last equation links the observations, $\mathcal{Y}$, and the state variable $\mathbf{x}$ through an observation operator $\mathbb{H} \in \mathbb{R}^{m \times n}$, plus some noise term $\boldsymbol{\epsilon} \in \mathbb{R}^m$. The observation noise $\boldsymbol{\epsilon}$ is a random, additive observation error. Note that $\boldsymbol{\epsilon}$ is often related to the instrumental error and indeed unknown, however, it can be assumed to be known *a priori*.

For the sake of simplicity we will restrict the discussion here to linear models, that is to say, the observation operator $\mathbb{H}$ and the dynamic model operator $\mathbb{M}$ are bounded as linear operators. The estimation for nonlinear systems is described in Chapter 2 and Chapter 3.

1.3 Linear optimal estimation

Here we consider a linear observation operator $\mathbf{H}$ instead of $\mathbb{H}$, so Eq.(1.3) reads,

$$\mathcal{Y} = \mathbf{H}\mathbf{x} + \boldsymbol{\epsilon}. \tag{1.4}$$

We aim to find an estimate $\hat{\mathbf{x}}$ of the true state $\mathbf{x}$ based on the knowledge of observation $\mathcal{Y}$. The error term $\boldsymbol{\epsilon}$ is assumed to be unbiased and has the covariance matrix $\mathbf{R}$ written as

$$\mathbf{R} = E(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T).$$

The operator $E(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T)$ stands for the mathematical expectation of the tensor product of two random error terms. This section is mainly based on Gelb (1974).

1.3.1 Linear least squares estimation

The problem described above corresponds to the classic linear least squares estimation. In least square estimation, we choose $\hat{\mathbf{x}}$ so that its value minimizes the discrepancy between the observation and the corresponding projection value of the model into the observation space. In a least-square sense, we seek the argument that minimizes the scalar cost function J:

$$J = (\mathcal{Y} - \mathbf{H}\hat{\mathbf{x}})^T(\mathcal{Y} - \mathbf{H}\hat{\mathbf{x}}). \tag{1.5}$$

The minimization of J is obtained by canceling its gradient with respect to $\hat{\mathbf{x}}$:

$$\frac{\partial J}{\partial \hat{\mathbf{x}}} = 0 = \mathbf{H}^T\mathbf{H}\hat{\mathbf{x}} - \mathbf{H}^T\mathcal{Y}. \tag{1.6}$$

The optimal estimator $\hat{\mathbf{x}}$ is given as:

$$\hat{\mathbf{x}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\mathcal{Y}, \tag{1.7}$$

Taking account of the error covariance term $\mathbf{R}$, we modify slightly the previous expressions by "weighted least-square" estimation which minimizes the weighted square discrepancy:

$$J = (\mathcal{Y} - \mathbf{H}\hat{\mathbf{x}})^T\mathbf{R}^{-1}(\mathcal{Y} - \mathbf{H}\hat{\mathbf{x}}). \tag{1.8}$$

The estimator in this case reads,

$$\hat{\mathbf{x}} = (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}^T\mathbf{R}^{-1}\mathcal{Y}. \tag{1.9}$$

Note this approach requires no specific knowledge of the error terms of the problem.

1.3.2 Maximum likelihood estimation

The principle of maximum likelihood estimation consists in estimating the argument that maximizes the conditional probability of the observations given the true state $p(\mathcal{Y}|\mathbf{x})$. We could easily observe from Eq.(1.4) that the conditional probability density function of $\mathcal{Y}$ conditioned upon $\mathbf{x}$ actually equals the density function of error $\boldsymbol{\epsilon}$ centered at $\mathbf{Hx}$. If the observation error $\boldsymbol{\epsilon}$ is assumed to be Gaussian distributed with mean zero and covariance matrix $\mathbf{R}$, the conditional probability density function can be expressed as,

$$p(\mathcal{Y}|\mathbf{x}) = \frac{1}{(2\pi)^{m/2}\det(\mathbf{R})^{1/2}}\exp[-\frac{1}{2}(\mathcal{Y} - \mathbf{Hx})^T\mathbf{R}^{-1}(\mathcal{Y} - \mathbf{Hx})]. \tag{1.10}$$

Maximizing this probability density function corresponds exactly to the minimization of the cost function (1.8).

1.3.3 Bayesian minimum variance estimation

From a Bayesian inference point of view, the optimal estimation to problem (1.4) consists of nothing but finding the *a posteriori* conditional density function, $p(\mathbf{x}|\mathcal{Y})$, of the state vector given the observation. According to Bayes' theorem,

$$p(\mathbf{x}|\mathcal{Y}) = \frac{p(\mathcal{Y}|\mathbf{x})p(\mathbf{x})}{p(\mathcal{Y})}, \tag{1.11}$$

where $p(\mathbf{x})$ is the *a priori* probability density function with regard to the state vector $\mathbf{x}$. This distribution indicates the *a priori* knowledge of the state variable. Eq.(1.2) and (1.1) give us a way of modeling the statistical model of $\mathbf{x}$ based on the background state $\mathbf{x}_0^b$ and the error term $\boldsymbol{\eta}$. The observation marginal probability density function $p(\mathcal{Y})$ is given by

$$p(\mathcal{Y}) = \int p(\mathcal{Y}|\mathbf{x})p(\mathbf{x})d\mathbf{x}.$$

This marginal remains hard to define. It corresponds to a normalization constant and its knowledge is in general not needed. $p(\mathcal{Y}|\mathbf{x})$ follows the same definition as in the maximum likelihood estimation.

Like the observation error term, the error term $\boldsymbol{\eta}$ is also assumed to be Gaussian distributed with mean zero and covariance matrix $\mathbf{B}$,

$$\mathbf{B} = E(\boldsymbol{\eta}\boldsymbol{\eta}^T),$$

so $\mathbf{x}$ given by

$$\mathbf{x} = \mathbf{x}^b + \boldsymbol{\eta}, \tag{1.12}$$

follows also a Gaussian distribution $\mathcal{N}(\mathbf{x}^b, \mathbf{B})$ with mean $\mathbf{x}^b$ and covariance matrix $\mathbf{B}$. Finally, the *a posteriori* conditional density function reads (up to a constant):

$$p(\mathbf{x}|\mathcal{Y}) \propto \exp[-\frac{1}{2}(\mathcal{Y} - \mathbf{H}\mathbf{x})^T\mathbf{R}^{-1}(\mathcal{Y} - \mathbf{H}\mathbf{x}) - \frac{1}{2}(\mathbf{x} - \mathbf{x}^b)^T\mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}^b)]. \tag{1.13}$$

Matrix $\mathbf{R}$ and $\mathbf{B}$ are called the observation error covariance matrix and the background error covariance, respectively. We note that the distribution of $\mathbf{x}$ conditioned on $\mathcal{Y}$ is still Gaussian.

With $p(\mathbf{x}|\mathcal{Y})$, we can easily build an estimator based on the **minimum variance estimation** principle. The minimum variance (unbiased) estimator $\hat{\mathbf{x}}$ provides, as its name indicates, the estimation of minimum variance. It can be proven that the minimum variance estimator corresponds to the *a posteriori* conditional mean $E(\mathbf{x}|\mathcal{Y})$. By definition of the variance, we have the following cost function,

$$J = \mathrm{tr}[\int (\hat{\mathbf{x}} - \mathbf{x})(\hat{\mathbf{x}} - \mathbf{x})^T d\mathbf{x}], \tag{1.14}$$

This form is equivalent to minimizing the norm of the error terms, which can be written as,

$$J = \int (\hat{\mathbf{x}} - \mathbf{x})^T(\hat{\mathbf{x}} - \mathbf{x})p(\mathbf{x}|\mathcal{Y})d\mathbf{x}, \tag{1.15}$$

The cost function is minimized for a null gradient.

$$\frac{\partial J}{\partial \hat{\mathbf{x}}} = \int 2(\hat{\mathbf{x}} - \mathbf{x})p(\mathbf{x}|\mathcal{Y})d\mathbf{x} = 0, \tag{1.16}$$

$$\hat{\mathbf{x}} = \int \mathbf{x}p(\mathbf{x}|\mathcal{Y})d\mathbf{x} = E(\mathbf{x}|\mathcal{Y}). \tag{1.17}$$

By introducing the Gaussian distributed statistical model of $\mathbf{x}$ and $\mathcal{Y}$, we intend to find the estimator which is a linear combination of $\mathcal{Y}$ and of the background state $\mathbf{x}^b$,

$$\hat{\mathbf{x}} = K(\mathcal{Y} - \mathbf{H}\mathbf{x}^b) + \mathbf{x}^b, \tag{1.18}$$

so we can rewrite the cost function in terms of K,

$$J = \int \big(K(\mathcal{Y} - \mathbf{H}\mathbf{x}^b) + \mathbf{x}^b - \mathbf{x}\big)^T\big(K(\mathcal{Y} - \mathbf{H}\mathbf{x}^b) + \mathbf{x}^b - \mathbf{x}\big)p(\mathbf{x}|\mathcal{Y})d\mathbf{x}, \tag{1.19}$$

$$\frac{\partial J}{\partial K} = \int \big(K(\mathcal{Y} - \mathbf{H}\mathbf{x}^b) + \mathbf{x}^b - \mathbf{x}\big)(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)^T p(\mathbf{x}|\mathcal{Y})d\mathbf{x} = 0, \tag{1.20}$$

$$\iff E\Big(\big(K(\mathcal{Y} - \mathbf{H}\mathbf{x}^b) + \mathbf{x}^b - \mathbf{x}\big)(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)^T\Big) = 0, \tag{1.21}$$

$$E\big(K(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)^T\big) = E\big((\mathbf{x} - \mathbf{x}^b)(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)^T\big), \tag{1.22}$$

$$K = E\big((\mathbf{x} - \mathbf{x}^b)(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)^T\big)\Big(E\big((\mathcal{Y} - \mathbf{H}\mathbf{x}^b)(\mathcal{Y} - \mathbf{H}\mathbf{x}^b)^T\big)\Big)^{-1}, \tag{1.23}$$

$$K = E\big(\boldsymbol{\eta}(\boldsymbol{\epsilon} + \mathbf{H}\boldsymbol{\eta})^T\big)\Big(E\big((\boldsymbol{\epsilon} + \mathbf{H}\boldsymbol{\eta})(\boldsymbol{\epsilon} + \mathbf{H}\boldsymbol{\eta})^T\big)\Big)^{-1}, \tag{1.24}$$

$$K = E\big(\boldsymbol{\eta}\boldsymbol{\eta}^T\mathbf{H}\big)\Big(E\big(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T + \mathbf{H}\boldsymbol{\eta}\boldsymbol{\eta}^T\mathbf{H}^T\big)\Big)^{-1}, \tag{1.25}$$

To simplify the expression K, we account for the relationships in Eq.(1.4) and Eq.(1.12). Besides, if $\boldsymbol{\eta}$ and $\boldsymbol{\epsilon}$ are assumed to be uncorrelated, then expression K reads:

$$K = \mathbf{B}\mathbf{H}^T\left(\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T\right)^{-1}. \tag{1.26}$$

The estimator is thus given by

$$\hat{\mathbf{x}} = \mathbf{x}^b + \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1}(\mathcal{Y} - \mathbf{H}\mathbf{x}^b), \tag{1.27}$$

or alternatively through the Sherman-Morrison-Woodbury formula[1] as

$$\hat{\mathbf{x}} = \mathbf{x}^b + (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + \mathbf{B}^{-1})^{-1}\mathbf{H}^T\mathbf{R}^{-1}(\mathcal{Y} - \mathbf{H}\mathbf{x}^b) \tag{1.28}$$

The second expression collapses to Eq.(1.9) when neither *a piori* knowledge of the state nor the error is available.

From Eq.(1.27), the *a posteriori* error covariance matrix $\hat{\mathbf{P}}$ is obtained as:

$$\hat{\mathbf{P}} = E((\hat{\mathbf{x}} - \mathbf{x}^t)(\hat{\mathbf{x}} - \mathbf{x}^t)^T), \tag{1.29}$$

which reads with form (1.28),

$$\hat{\mathbf{P}} = \mathbf{B} - \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{B}, \tag{1.30}$$

Note that an empirical (unbiased) expression of (1.29) as an average value calculated from a finite set of samples $\mathbf{x}^{(i)}$ is given by,

$$\hat{\mathbf{P}} = \frac{1}{N-1}\sum_{i=1}^{N}(\hat{\mathbf{x}} - \mathbf{x}^{(i)})(\hat{\mathbf{x}} - \mathbf{x}^{(i)})^T, \quad \hat{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}^{(i)}. \tag{1.31}$$

1.4 Available methods

In section 1.3, optimal estimation (in a linear sense) has been written as a static problem considering only Eq.(1.2) and Eq.(1.3) of the DA problem. Here we additionally take into account the state variables dynamics (1.1).

1.4.1 Sequential methods

The sequential methods denote many different methods including the particle filter, the Kalman filter, etc. This type of methods is mainly based on the principle of Bayesian minimum variance estimation. The term "sequential" refers to the way in which the observations are assimilated: the state is propagated by the dynamic model forward in time; at a certain time when the observation is available, the state forecast $\mathbf{x}^f$ is corrected, yielding the analysis state $\mathbf{x}^a$.

[1]The Sherman-Morrison-Woodbury formula is $(A+UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}$, where A, U, C and V are matrix (Higham, 2002).

Kalman filter. The Kalman filter is derived from minimum variance estimation in a Gaussian linear context. To comply with traditional Kalman filter notations, we rewrite Eq.(1.1) in a discrete linear form,

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k, \tag{1.32}$$

where Φ is the linear dynamic discrete operator. Suppose we have some *a priori* information about the state and its error covariance matrix at time instant t_k, and the observation is available at time instant t_{k+1}. Then the propagation of error can be described by,

$$\mathbf{P}^f_{k+1} = \Phi_k \mathbf{P}^a_k \Phi^T_k + \mathbf{Q}_k, \tag{1.33}$$

where $\mathbf{P}^f$ and $\mathbf{P}^a$ bear the name of background or analysis error covariance matrix, respectively.

$$\mathbf{P}^f = E((\mathbf{x}^f - \mathbf{x}^t)(\mathbf{x}^f - \mathbf{x}^t)^T), \tag{1.34}$$

$$\mathbf{P}^a = E((\mathbf{x}^a - \mathbf{x}^t)(\mathbf{x}^a - \mathbf{x}^t)^T), \tag{1.35}$$

and $\mathbf{Q} \in \mathbb{R}^n$ is the model error covariance matrix defined by $E(\mathbf{q}\mathbf{q}^T)$. The full scheme of the Kalman filter can be expressed as algorithm 1.

Algorithm 1 Kalman Filter Algorithm

1: **procedure** ANALYSIS

2: From time instant k, compute the forecast state $\mathbf{x}^f_{k+1}$ with the forward integration of relation (1.32)

3: Compute the forecast error covariance matrix $\mathbf{P}^f_{k+1}$ based on the error propagation equation (1.33)

4: The analysis state $\mathbf{x}^a_{k+1}$ can be obtained directly from

$$\mathbf{x}^a_{k+1} = \mathbf{x}^f_{k+1} + \mathbf{K}_{k+1}(\mathcal{Y}_{k+1} - \mathbf{H}\mathbf{x}^f_{k+1}), \tag{1.36}$$

where $\mathbf{K}_{k+1}$ is the Kalman gain matrix. It reads from Eq.(1.27) by substituting $\mathbf{B}$ with $\mathbf{P}^f$:

$$\mathbf{K}_{k+1} = \mathbf{P}^f_{k+1}\mathbf{H}^T(\mathbf{H}\mathbf{P}^f_{k+1}\mathbf{H}^T + \mathbf{R})^{-1}, \tag{1.37}$$

or from Eq.(1.28)

$$\mathbf{K}_{k+1} = (\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} + {\mathbf{P}^f_{k+1}}^{-1})^{-1}\mathbf{H}^T\mathbf{R}^{-1}. \tag{1.38}$$

The Eq.(1.37) also corresponds to the Sherman-Morrison-Woodbury formula, this expression of the Kalman gain is computationally advantageous as the inversion involved is performed in the observation space, which is usually of a lower dimension than the state space.

5: The analysis error covariance matrix $\mathbf{P}^a_{k+1}$ is updated by Eq.(1.30):

$$\mathbf{P}^a_{k+1} = \mathbf{P}^f_{k+1} - \mathbf{P}^f_{k+1}\mathbf{H}^T(\mathbf{H}\mathbf{P}^f_{k+1}\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{P}^f_{k+1}. \tag{1.39}$$

6: Propagate the analysis state $\mathbf{x}_{k+1}$ and associated error covariance matrix $\mathbf{P}^a_{k+1}$ to the next instant when the observations are available

7: **end procedure**

The detailed description of the Kalman filter and related techniques will be given in chapter 3.

Particle filter. The particle filter is directly related the Monte-Carlo methods in which a set of particles are formulated to represent the model's probability density function. These particles are driven by the dynamic model in time and are weighted according to an importance sampling ratio. Eventually one can have all the information needed from the weighted ensemble (mean, variance, etc.). The procedure consists in approximating the posterior probability density function as a weighted combination of the particles' prior probability density function:

$$p(\mathbf{x}|\mathcal{Y}) = \sum_{i=1}^{N} q_i \delta(\mathbf{x}_i), \tag{1.40}$$

the weight q_i for each particle is calculated by:

$$q_i = \frac{p(\mathcal{Y}|\mathbf{x}_i)}{\sum_{j=1}^{N} p(\mathcal{Y}|\mathbf{x}_j)}. \tag{1.41}$$

where the conditional probability density function $p(\mathcal{Y}|\mathbf{x}_j)$ usually assumes the Gaussian form described in equation (1.10). In practice, this weight is attached to the corresponding particle. This procedure is called importance sampling. It is interesting to point out that each particle remains unchanged during this course.

Note also that the particle filter is still not yet developed for meteorological or oceanographic applications.Van Leeuwen (2009) gives a comprehensive review of particle filtering in geophysical systems.

1.4.2 Variational methods

The variational methods originated from the optimal control theory and variation calculus (Lions, 1971). The principle of calculus of variation simply states that the extrema of certain functional is obtained by setting the first variation of the functional to zero. Note that this is only a necessary condition. Suppose we have a functional defined as

$$J = \int_{t_0}^{t_f} F(t, x, x')dt, \tag{1.42}$$

where x is a function of t and F is twice differentiable in all variables. The necessary condition for J to be an extremum is

$$\delta J = 0, \tag{1.43}$$

where δ is the variation operator. This also relates to the Euler-Lagrange equation,

$$\frac{d}{dt}\frac{\partial F}{\partial x'} - \frac{\partial F}{\partial x} = 0. \tag{1.44}$$

The variational methods have been first introduced into the DA problem by Sasaki (1958) and later extended to the "four dimensional analysis" in Sasaki (1970). In this approach, the cost function is defined as the sum of squares of the discrepancy between observation and "objectively modified values". These "objectively modified values" are denoted by $Y_s \equiv \mathbf{H}\mathbf{x}(t)$ and correspond to the projection value of the model space in the observation space. The cost function takes the form:

$$J \equiv \int_{t_0}^{t_f} \big(Y_s(t) - \mathcal{Y}(t)\big)^T \mathbf{R}^{-1} \big(Y_s(t) - \mathcal{Y}(t)\big) dt. \tag{1.45}$$

The first variation can be expressed as

$$\begin{aligned}\delta J &= \delta \int_{t_0}^{t_f} \big(Y_s(t) - \mathcal{Y}(t)\big)^T \mathbf{R}^{-1} \big(Y_s(t) - \mathcal{Y}(t)\big) dt, \\ &= 2 \int_{t_0}^{t_f} \{\frac{\partial J}{\partial Y_s}, \delta Y_s(t)\} dt, \end{aligned} \tag{1.46}$$

where $\{\bullet, \bullet\}$ denotes the associated inner product. From this definition, we can conclude the form of the gradient of the cost function $\boldsymbol{\nabla} J$.

$$\delta J = \{\boldsymbol{\nabla}_{\mathbf{x}} J, \delta \mathbf{x}\}, \tag{1.47}$$

Assuming the dynamic model is linear with $\mathbf{x}_t = \Phi_t \mathbf{x}_0$, the cost function depending on the initial condition $\mathbf{x}_0$ can be written as:

$$J(\mathbf{x}_0) = \int_{t_0}^{t_f} \big(\mathbf{H}\Phi_t \mathbf{x}_0 - \mathcal{Y}(t)\big)^T \mathbf{R}^{-1} \big(\mathbf{H}\Phi_t \mathbf{x}_0 - \mathcal{Y}(t)\big) dt, \tag{1.48}$$

and the first variation reads,

$$\begin{aligned}\delta J &= \int_{t_0}^{t_f} (\mathbf{H}\Phi_t \delta \mathbf{x}_0)^T \mathbf{R}^{-1} \big(\mathbf{H}\Phi_t \mathbf{x}_0 - \mathcal{Y}(t)\big) dt \\ &= 2 \int_{t_0}^{t_f} \{\boldsymbol{\nabla}_{\mathbf{x}_0} J, \delta \mathbf{x}_0\} dt \end{aligned} \tag{1.49}$$

By setting the first variation of the cost function to zero, we have the solution $\hat{\mathbf{x}}_0$ to the variational problem.

$$\hat{\mathbf{x}}_0 = (\Phi_t^T \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \Phi_t)^{-1} \Phi_t^T \mathbf{H}^T \mathbf{R}^{-1} \mathcal{Y}. \tag{1.50}$$

We could easily verify that in a linear scenario, the optimizer of Eq.(1.50) is indeed equivalent to the optimizer (1.9) as can be concluded from linear least square estimation with a flow matrix Φ_t defined as the identity (no dynamics); it is also equivalent to the optimizer (1.27) deduced from minimum variance estimation with an *a priori* Gaussian distributed background error term. Under the same condition, the equivalence between the variational methods and the Kalman filter can also be demonstrated (Li and Navon, 2001). The final analysis state at t_f based on an integration of the initial analysis Eq.(1.50) and the analysis state at t_f based on equation (1.36) are equivalent. However, when applied within the assimilation interval, the variational assimilation and the Kalman filter differ. The former corresponds to a smoothing procedure as the whole set of measurements are taken into account, whereas the latter relies only on the past data.

Although the variational methods show a great potential, their applications are limited for high dimensional spaces. The problem is still more intractable for nonlinear operators. Many important works have been done in the 1980s that are designed to deal with this difficulties. Le Dimet and Talagrand (1986), as one of the pioneers in this field, introduced the adjoint minimization technique that transforms the constrained problem into a sequence of unconstrained problem. Since then, the efficient implementation of variational assimilation techniques have been made possible.

To sum up, the data assimilation problem in the framework of variational methods can be rephrased as: we intend to estimate the state vector, $\mathbf{x}$, from the Eq.(1.3) subject to the *constraint* in the form of the dynamic evolution model equation (1.1) and the initial state modeling equation (1.2). The further development of variational methods is shown in Chapter 2.

1.4.3 Other methods

Several other data assimilation methods have been proposed in the literature. Some are designed as a combination of several aforementioned methods. For example, the Weighted Ensemble Kalman Filter (WEnKF) (Papadakis et al., 2010; Beyou et al., 2013b) is a combination of the particle filter and the ensemble Kalman filter. It has proved to be promising in dealing with high uncertainty image data. Some are based on simplification of the previous frameworks. For instance, the optimal nudging method is realized by adding an extra forcing relaxation term into the dynamic model. Such forcing term is usually of diffusive-type constructed by the product of a tunable nudging coefficient and the discrepancy between the observation and the corresponding projection of the model space in the observation space (Lorenc et al., 1991; Zou et al., 1992).

1.5 Summary

Starting from a general mathematical description of the system of equations involved in the DA, this chapter described in detail the basic principles of optimal estimation. A brief preview of the popular methods used in data assimilation has been as well presented.

Chapter 2

Variational Methods

In this chapter, we aim at providing a more complete presentation of variational assimilation methods. In particular, the potential nonlinearity of the observation and of the dynamics will be taken into consideration.

2.1 The problem of variational data assimilation

We recall the cost function (1.8),

$$J \equiv \int_{t_0}^{t_f} \big(Y_s(t) - \mathcal{Y}(t)\big)^T \mathbf{R}^{-1} \big(Y_s(t) - \mathcal{Y}(t)\big) dt,$$

where now the observation equations

$$Y_s(t) = \mathbb{H}(\mathbf{x}_t)$$

involved a nonlinear operator. And the state variable, $\mathbf{x}_t$, is driven by the nonlinear evolution equation dynamics.

$$\mathbf{x}_t = \varphi_t(\mathbf{x}_0) = \mathbf{x}_0 + \int_0^t \mathbb{M}(\mathbf{x}(s))ds, \tag{2.1}$$

The flow map (viewed here as a function of a random initial condition) is denoted $\varphi_t(\mathbf{x}_0)$.

We can also define $\partial_{\mathbf{x}}\varphi_t(\mathbf{x})$ as the tangent linear operator of flow map φ_t,

$$\delta\mathbf{x}_t = \partial_{\mathbf{x}}\varphi_t(\mathbf{x})\delta\mathbf{x}_0, \tag{2.2}$$

The tangent linear model of the differential expression of dynamics (1.1) is defined as,

$$\partial_t \delta\mathbf{x}(t,x) + \partial_{\mathbf{x}}\mathbb{M}(\mathbf{x}(t,x),u)\delta\mathbf{x}(t,x) = 0, \tag{2.3}$$

where $\partial_{\mathbf{x}}\mathbb{M}(\mathbf{x})$ denotes the tangent linear operator of $\mathbb{M}$. It is defined as

$$\lim_{\beta\to 0} \frac{\mathbb{M}(\mathbf{x}+\beta d\mathbf{x}) - \mathbb{M}(\mathbf{x})}{\beta} = \partial_{\mathbf{x}}\mathbb{M}(\mathbf{x})d\mathbf{x}. \tag{2.4}$$

Therefore the tangent flow map corresponds to the integration of the tangent linear model,

$$\delta\mathbf{x}_t = \partial_{\mathbf{x}}\varphi_t(\mathbf{x})\delta\mathbf{x}_0 = \delta\mathbf{x}_0 + \int_0^t \partial_{\mathbf{x}}\mathbb{M}(\mathbf{x}(s))\delta\mathbf{x}(s)ds. \tag{2.5}$$

2.1.1 Adjoint equation technique

Recall the first variation equation (1.49),

$$\begin{aligned}\delta J(Y_s(t)) &= 2\int_{t_0}^{t_f}\{\frac{\partial J}{\partial Y_s},\delta Y_s(t)\}dt,\\ &= 2\int_{t_0}^{t_f}\{\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big),\delta Y_s(t)\}dt.\end{aligned} \tag{2.6}$$

Lagrange multipliers

The general principle of dealing with constrained optimization problem aims at incorporating the constrain in the cost function. One way to do this consists in modifying the cost function with an additional term weighted by Lagrange multipliers. The new cost function reads,

$$\mathcal{L}(\mathbf{x},\Lambda)=\int_{t_0}^{t_f}\big(Y_s(t)-\mathcal{Y}(t)\big)^T\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big)dt+\int_{t_0}^{t_f}\{\Lambda,\partial_t\mathbf{x}(t,x)+\mathbb{M}(\mathbf{x}(t,x)))\}dt. \tag{2.7}$$

The minimization problem of cost function (1.8) subject to dynamic constrain (1.1) is now transformed in an unconstrained minimization problem of Lagrange function $\mathcal{L}$, The minimization of $\mathcal{L}$ is obtained by canceling its derivative:

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\mathbf{x}} &= \int_{t_0}^{t_f}2\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(\mathbb{H}(\mathbf{x}_t)-\mathcal{Y}(t)\big)dt\\ &\quad+\frac{\partial}{\partial\mathbf{x}}(\Lambda_{tf}\mathbf{x}_{tf}-\Lambda_0\mathbf{x}_0-\int_{t_0}^{t_f}\frac{\partial\Lambda}{\partial t}\mathbf{x}dt)+\int_{t_0}^{t_f}\{\Lambda,\partial_{\mathbf{x}}\mathbb{M}\}dt,\\ &= \int_{t_0}^{t_f}\Big(2\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(\mathbb{H}(\mathbf{x}_t)-\mathcal{Y}(t)\big)-\frac{\partial\Lambda}{\partial t}+\partial_{\mathbf{x}}\mathbb{M}^*\Lambda\Big)dt=0,\end{aligned} \tag{2.8}$$

$$\frac{\partial\mathcal{L}}{\partial\Lambda}=\text{Eq.(1.1)}. \tag{2.9}$$

In equation (2.8), as defined in Eq.(2.5), the operator $\partial_{\mathbf{x}}\mathbb{H}$ is the tangent observation operator defined by,

$$\lim_{\beta\to0}\frac{\mathbb{H}(\mathbf{x}_t+\beta d\mathbf{x}_t)-\mathbb{H}(\mathbf{x}_t)}{\beta}=\partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)d\mathbf{x}_t, \tag{2.10}$$

and the operators $\partial_{\mathbf{x}}\mathbb{H}^*$ and $\partial_{\mathbf{x}}\mathbb{M}^*$ are adjoint operator associated with the tangent linear operators $\partial_{\mathbf{x}}\mathbb{H}$ and $\partial_{\mathbf{x}}\mathbb{M}$ respectively: $\{\mathcal{H}g,f\}=\{g,\mathcal{H}^*f\}$. Eq.(2.8) yields indeed the adjoint equation associated with Lagrange multipliers vector. And we can show that the gradient of cost function with respect the initial condition $\mathbf{x}_0$ is:

$$\boldsymbol{\nabla}_{\mathbf{x}_0}J=-\Lambda(t_0),$$

Note that we used the rule of derivative of an inner product given any vectors $\mathbf{u}$ and $\mathbf{v}$,

$$\frac{\partial}{\partial\mathbf{x}}\{\mathbf{u},\mathbf{v}\}=\{\frac{\partial\mathbf{u}}{\partial\mathbf{x}},\mathbf{v}\}+\{\mathbf{u},\frac{\partial\mathbf{v}}{\partial\mathbf{x}}\},$$

and the definition of adjoint,

$$\{\mathcal{H}\mathbf{u},\mathbf{v}\}=\{\mathbf{u},\mathcal{H}^*\mathbf{v}\}.$$

The arguments of tangent linear operator and its adjoint operator indicate, when clarity is needed, the point at which this derivative is computed. It is straightforward to check that the linear tangent of a linear operator is the operator itself.

Reducing the constraint

Another method named originally "reducing the constraint" from Le Dimet and Talagrand (1986); Talagrand (1997) consists to express the constraint by transforming the cost function to one which only depends on the initial condition. In order to evaluate the term $\delta Y_s(t)$, we must consider that the tangent linear observation operator $\partial_{\mathbf{x}}\mathbb{H}$ with $\delta\mathbf{x}_t$ is defined through Eq.(2.5), so we have,

$$\delta Y_s(t) = \partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_{\mathbf{x}}\varphi_t(\mathbf{x})\delta\mathbf{x}_0, \tag{2.11}$$

then the first variation equals to,

$$\delta J = 2\int_{t_0}^{t_f} \{\mathbf{R}^{-1}\big(Y_s(t) - \mathcal{Y}(t)\big), \partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_{\mathbf{x}}\varphi_t(\mathbf{x})\delta\mathbf{x}_0\}dt \tag{2.12}$$

An elegant solution to this problem consists in relying on an adjoint formulation Lions (1971). Within this formalism, considering the adjoint operator $\partial_{\mathbf{x}}\mathbb{H}^*, \partial_{\mathbf{x}}\varphi_t^*$ of linear operators $\partial_{\mathbf{x}}\mathbb{H}, \partial_{\mathbf{x}}\varphi_t$ respectively allows us writing the relationship,

$$\{\mathbf{R}^{-1}\big(Y_s(t) - \mathcal{Y}(t)\big), \partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_{\mathbf{x}}\varphi_t(\mathbf{x})\delta\mathbf{x}_0\} = \{\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(Y_s(t) - \mathcal{Y}(t)\big), \delta\mathbf{x}_0\}. \tag{2.13}$$

Note that $\partial_{\mathbf{x}}\varphi_t^*$ is associated with the tangent linear flow map operator and is related to the adjoint model operator $\partial_{\mathbf{x}}\mathbb{M}^*$. According to the definition of the gradient (1.47), the gradient is given by,

$$\boldsymbol{\nabla}_{\mathbf{x}_0} J = 2\int_{t_0}^{t_f} \partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(Y_s(t) - \mathcal{Y}(t)\big)dt. \tag{2.14}$$

In order to evaluate the gradient, we define the adjoint variable $\delta^*\mathbf{x}$ driven by the adjoint model operator $\partial_{\mathbf{x}}\mathbb{M}^*$,

$$\begin{cases} -\partial_t\delta^*\mathbf{x}_t + \partial_{\mathbf{x}}\mathbb{M}^*\delta^*\mathbf{x}_t = -2\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}(\mathbb{H}(\mathbf{x}_t) - \mathcal{Y}_t), \\ \delta^*\mathbf{x}(t_f) = 0. \end{cases} \tag{2.15}$$

Because the adjoint model operator $\partial_{\mathbf{x}}\mathbb{M}^*$ is also linear, the solution to the above equation can be put as a linear combination of the $\delta^*\mathbf{x}(t_f)$, and such a linear operator is indeed the adjoint operator aforementioned $\partial_{\mathbf{x}}\varphi_t^*$.

$$\delta^*\mathbf{x}_t = \partial_{\mathbf{x}}\varphi_t^*\delta^*\mathbf{x}_{t_f}. \tag{2.16}$$

The gradient functional is thus obtained by a backward integration of equation (2.15) corresponding to the adjoint variable, $\delta^*\mathbf{x}$, with its final value sets to zero. The solution of the backward integration supplies the initial value of adjoint variable $\delta^*\mathbf{x}_0$, which provides directly the gradient as:

$$\boldsymbol{\nabla}_{\mathbf{x}_0} J = -\delta^*\mathbf{x}_0.$$

The backward integration of Eq.(2.15) requires the knowledge of the innovation term $\mathbb{H}(\mathbf{x}_t) - \mathcal{Y}_t$, therefore a forward integration of the nonlinear dynamical system must be

done prior to the adjoint calculation. The model trajectory calculated from the nonlinear dynamic system is used to construct the tangent linear model operators.

It is clear that the Lagrange multipliers vector mentioned in the previous method and the adjoint variable are indeed equivalent. For the sake of simplicity, we will multiply the cost function (1.8) with $\frac{1}{2}$ so that the factor of the RHS of adjoint equation (2.15) reduces to one. No matter which approach we use, by canceling the gradient with respect to the initial condition,

$$\boldsymbol{\nabla}_{\mathbf{x}_0} J(\hat{\mathbf{x}}_0) = 0,$$

we can deduce in theory the minimizer,

$$\hat{\mathbf{x}}_0 = -\mathcal{H}^{-1}\partial_{\mathbf{x}} J(t_0), \tag{2.17}$$

where $\mathcal{H}$ is the Hessian matrix gathering the cost function second derivatives:

$$\mathcal{H} = \int_{t_0}^{t_f} \partial_{\mathbf{x}}\varphi_t^* \partial_{\mathbf{x}}\mathbb{H}^* \mathbf{R}^{-1} \partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t dt. \tag{2.18}$$

2.1.2 Parameter estimation with variational data assimilation

In this section, we will describe the model parameter estimation with variational assimilation technique. Recall that (1.1), takes the parameter u into consideration. Thus the flow map operator reads,

$$\varphi_t(\mathbf{x}_0, u) = \mathbf{x}_t = \mathbf{x}_0 + \int_0^t \mathbb{M}(\mathbf{x}(s), u) ds, \tag{2.19}$$

The modified cost function of Eq. (1.8) reads,

$$J(\mathbf{x}_0, u) \equiv \int_{t_0}^{t_f} \big(Y_s(t)) - \mathcal{Y}(t)\big)^T \mathbf{R}^{-1} \big(Y_s(t) - \mathcal{Y}(t)\big) dt,$$

where

$$Y_s(t) = \mathbb{H}(\mathbf{x}_t)(\varphi_t(\mathbf{x}_0, u)). \tag{2.20}$$

We intend to find the first variation of J in the form,

$$\delta J(\mathbf{x}_0, u) = 2\int_{t_0}^{t_f} (\{\frac{\partial J}{\partial \mathbf{x}_0}, \delta \mathbf{x}_0\} + \{\frac{\partial J}{\partial u}, \delta u\}) dt. \tag{2.21}$$

The partial derivative $\frac{\partial J}{\partial \mathbf{x}_0}$ is obtained through Eq.(2.14), we still need to determine the partial derivative of the cost function with respect to the parameter u. The tangent linear model corresponding to Eq.(1.1) is defined as,

$$\partial_t \delta\mathbf{x}(t,x) + \partial_{\mathbf{x}}\mathbb{M}(\mathbf{x}(t,x),u)\delta\mathbf{x}(t,x) + \partial_u \mathbb{M}(\mathbf{x}(t,x),u)\delta u = 0, \tag{2.22}$$

where $\partial_u \mathbb{M}(\mathbf{x}, u)$ denotes the tangent linear operator of $\mathbb{M}$, defined as

$$\lim_{\beta\to 0} \frac{\mathbb{M}(\mathbf{x}, u+\beta du) - \mathbb{M}(\mathbf{x}, u)}{\beta} = \partial_u \mathbb{M}(\mathbf{x}, u) du. \tag{2.23}$$

The linearity of $\partial_{\mathbf{x}}\mathbb{M}(\mathbf{x})$ and $\partial_u\mathbb{M}(\mathbf{x})$ allow us to express the solution of Eq.(2.22), $\delta\mathbf{x}_t$ as a linear combination of $\delta\mathbf{x}_0$ and δu with the tangent linear operator of $\varphi_t(\mathbf{x},u)$ with respect to u:

$$\begin{aligned}\delta\mathbf{x}_t &= \partial_{\mathbf{x}}\varphi_t(\mathbf{x},u)\delta\mathbf{x}_0 + \partial_u\varphi_t(\mathbf{x},u)\delta u \\ &= \delta\mathbf{x}_0 + \int_0^t \partial_{\mathbf{x}}\mathbb{M}(\mathbf{x}(s),u)\delta\mathbf{x}(s)ds + \int_0^t \partial_u\mathbb{M}(\mathbf{x}(s),u)\delta u ds,\end{aligned} \tag{2.24}$$

so the variation of Y_s reads,

$$\delta Y_s = \partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_{\mathbf{x}}\varphi_t(\mathbf{x},u)\delta\mathbf{x}_0 + \partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_u\varphi_t(\mathbf{x},u)\delta u, \tag{2.25}$$

and the first variation of J is given by,

$$\begin{aligned}\delta J = 2\int_{t_0}^{t_f}\{\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big),\partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_{\mathbf{x}}\varphi_t(\mathbf{x},u)\delta\mathbf{x}_0\}dt+ \\ 2\int_{t_0}^{t_f}\{\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big),\partial_{\mathbf{x}}\mathbb{H}(\mathbf{x}_t)\partial_u\varphi_t(\mathbf{x},u)\delta u\}dt.\end{aligned} \tag{2.26}$$

By introducing the adjoint linear model $\partial_u\varphi_t^*(\mathbf{x},u)$ of the tangent linear model $\partial_u\varphi_t(\mathbf{x},u)$ to the second term in the RHS of (2.26), considering Eq.(2.13) we obtain the following relationship:

$$\begin{aligned}\delta J = 2\int_{t_0}^{t_f}\{\partial_{\mathbf{x}}\varphi_l^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big),\delta\mathbf{x}_0\}+ \\ 2\int_{t_0}^{t_f}\{\partial_u\varphi_l^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big),\delta u\}.\end{aligned} \tag{2.27}$$

Hence the partial derivative of the cost function with respect to u reads,

$$\frac{\partial J}{\partial u} = 2\int_{t_0}^{t_f}\partial_u\varphi_l^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\big(Y_s(t)-\mathcal{Y}(t)\big)dt. \tag{2.28}$$

This equation is obtained here by the approach "reduce the constraint" aforementioned. This partial derivative can also be derived from the Lagrange Multiplier technique. In Eq.(2.7), $\mathbb{M}(\mathbf{x}(t,x))$ is replaced by $\mathbb{M}(\mathbf{x}(t,x),u)$. This modification does not change the expression of $\frac{\partial\mathcal{L}}{\partial\mathbf{x}}$, and leads to an extra partial derivative with respect to parameter u,

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial u} =& \frac{\partial}{\partial u}\int_{t_0}^{t_f}\{\Lambda,\frac{\partial\mathbf{x}}{\partial t}\}dt + \frac{\partial}{\partial u}\int_{t_0}^{t_f}\{\Lambda,\mathbb{M}(\mathbf{x}_t,u)\}dt \\ &= \frac{\partial}{\partial u}(\Lambda_{tf}\mathbf{x}_{tf} - \Lambda_0\mathbf{x}_0 - \int_{t_0}^{t_f}\frac{\partial\Lambda}{\partial t}\mathbf{x}dt) + \int_{t_0}^{t_f}\{\Lambda,\partial_u\mathbb{M}\}dt, \\ &= \int_{t_0}^{t_f}\partial_u\mathbb{M}^*\Lambda dt = 0,\end{aligned} \tag{2.29}$$

and

$$\frac{\partial\mathcal{L}}{\partial u} = \frac{\partial J}{\partial u},$$

at stationary point. It is easy to see that Eqs.(2.28) and (2.29) are indeed equivalent with Λ, the Lagrange multiplier vector or the adjoint variable associated with $\mathbf{x}$, driven by backward integration of adjoint equation (2.15). The partial derivative is calculated based on an integration over time of the product of adjoint model $\partial_u\mathbb{M}^*$ and Λ_t.

2.2 Four dimensional variational data assimilation

In practice, we usually have some *a priori* knowledge on the unknown initial condition besides the observation discrepancy defined in Eq.(1.8). The initial condition and the *a priori* background state is linked by Eq. (1.2). The background state guaranties the well posedness of the state vector inference from the observation, since the observation is usually of low resolution compared to the state vector; moreover, it provides a balanced relationship amid the state vectors, for instance it enables enforcing the geostrophic relationship between the pressure gradient and the velocity.

A standard four dimensional variational data assimilation problem is formulated as the minimization of the following objective function from Lorenc (1986):

$$J(\mathbf{x}_0) = \frac{1}{2}\|\mathbf{x}(t_0,\mathrm{x}) - \mathbf{x}_0^b(\mathrm{x})\|^2_{\mathbf{B}} + \frac{1}{2}\int_{t_0}^{t_f} \|\mathbb{H}(\mathbf{x}(t,\mathrm{x})) - \mathcal{Y}(t,\mathrm{x})\|^2_{\mathbf{R}} dt. \tag{2.30}$$

This objective function involves the L_2 norm with respect to the inverse covariance tensor $\|f\|^2_A = \int_\Omega f(x)A^{-1}(x,y)f(y)dxdy$. The associated minimization problem is referred to in the literature as the strong constraint variational assimilation formulation. This constitutes an optimal control problem where one seeks the value of the initial condition, $\mathbf{x}_0$, that yields the lowest error between the measurements and the state variable trajectory. Note that such an energy function can be interpreted as the log likelihood function associated to the *a posteriori* distribution of the state given the past history of measurements and the background.

2.2.1 Probability point of view of 4DVar

Assuming the flow map is a diffeomorphism (e.g. a differentiable map whose inverse exists and which is differentiable as well), we get

$$\begin{aligned}
p(\mathbf{x}_t|Y_{t_f},\dots,Y_{t_0},\mathbf{x}^b_{t_0}) \propto & p(Y_{t_f},\dots,Y_{t_0}|\varphi_t(\mathbf{x}_0),\mathbf{x}^b_0)p(\varphi_t(\mathbf{x}_0)|\mathbf{x}^b_0),\\
\propto & \prod_{t_i=0}^{t_f} p(Y_{t_i}|\varphi_{t_i}(\mathbf{x}_0))p(\varphi_t(\mathbf{x}_0)|\mathbf{x}^b_0),\\
\propto & \prod_{t_i=0}^{t_f} p(Y_{t_i}|\varphi_{t_i}(\mathbf{x}_0))p(\varphi_t^{-1}(\mathbf{x}_t)|\mathbf{x}^b_0)|\mathrm{det}\partial_{\mathbf{x}}\varphi_{\mathrm{t}}^{-1}(\mathbf{x}_{\mathrm{t}})|,\\
\propto & \prod_{t_i=0}^{t_f} p(Y_{t_i}|\varphi_{t_i}(\mathbf{x}_0))p(\mathbf{x}_0|\mathbf{x}^b_0)|\mathrm{det}\partial_{\mathbf{x}}\varphi_{\mathrm{t}}^{-1}(\mathbf{x}_{\mathrm{t}})|.
\end{aligned} \tag{2.31}$$

We assumed here that the observations at a given time depend only on the state at the same time, and that they are conditionally independent with respect to the state variables. Now, replacing the probability distribution by their expressions Eqs (1.3) and (1.2), we get the objective function Eq.(2.30) up to a time dependent factor. If the flow map is volume preserving and invertible, we have $|\mathrm{det}\partial_{\mathbf{x}}\varphi_{\mathrm{t}}^{-1}(\mathbf{x}_{\mathrm{t}})| = |\mathrm{det}\partial_{\mathbf{x}}\varphi_{\mathrm{t}}(\mathbf{x}_{\mathrm{t}_0})| = 1$ and we get exactly the sought energy function. In that case, the maximum *a posteriori* estimate hence corresponds to the objective function minima. In the Gaussian case, the maximum *a posteriori* estimate and the conditional mean with respect to the whole measurements[1]

[1] It is a well-known fact that this estimate constitutes the minimum variance estimate, Anderson and Moore (1979).

trajectory are identical. Let us note, nevertheless, that the *a posteriori* pdf is Gaussian only if the dynamical model and the observation operator are both linear (as in that case $E\varphi_t(\mathbf{x}_0) = \varphi(E(\mathbf{x}_0))$). We also point out that the tangent linear expression of the dynamical models, as defined in geophysical applications, are even not invertible in general. For a time and space discrete approximation of the tangent linear dynamical operator (*i.e.* a matrix), a pseudo inverse may be defined from a Singular Value Decomposition (SVD). Noting $\partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0)$ this matrix operator, which depends on time and on a given initial condition, we can write

$$\partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0) = U(t,\mathbf{x}_0)\Sigma(t,\mathbf{x}_0)V^T(t,\mathbf{x}_0), \tag{2.32}$$

where $\Sigma(t,\mathbf{x}(0)) = \mathrm{diag}(\sigma_1,\cdots,\sigma_{\mathrm{p}},0\cdots,0)$ is a real diagonal matrix gathering the square-root of the eigenvalues of the matrix $\partial_{\mathbf{x}}\varphi_t^T\partial_{\mathbf{x}}\varphi_t$, and both U and V are orthonormal matrices, corresponding respectively to eigenvectors of matrix $\partial_{\mathbf{x}}\varphi\partial_{\mathbf{x}}\varphi^T$ and $\partial_{\mathbf{x}}\varphi^T\partial_{\mathbf{x}}\varphi$ respectively. The three matrices are $n\times n$ and may depend on time and on the initial condition. The pseudo inverse is

$$\partial_{\mathbf{x}}\varphi_t^{-1}(\mathbf{x}_0) = V(t,\mathbf{x}_0)\Sigma^{-1}(t,\mathbf{x}_0)U^T(t,\mathbf{x}_0), \tag{2.33}$$

where $\Sigma^{-1}(t,\mathbf{x}_0') = \mathrm{diag}(1/\sigma_1,\cdots,1/\sigma_{\mathrm{p}},0\cdots,0)$ and for divergence free volume preserving dynamics we get $|\mathrm{det}\varphi_{\mathrm{t}}^{-1}(\mathbf{x}_{\mathrm{t}})| = |\prod_{\mathrm{i}=1}^{\mathrm{p}}\sigma_{\mathrm{i}}^{-1}| = 1$ and the logarithm of the posterior corresponds to objective function Eq.(2.30).

2.2.2 Byproducts of 4DVar data assimilation

We recall in this subsection some well-known properties of the standard variational functional that will be useful in the following (see Li and Navon (2001); Lorenc (1986); Rabier and Courtier (1992)). A (local) minimizer of functional Eq.(2.30) is provided for $\partial_{\mathbf{x}_0}J(\mathbf{x}_0) = 0$. The gradient with respect to the initial condition is given by Eq.(2.14):

$$\partial_{\mathbf{x}_0}J(\mathbf{x}_0) = \mathbf{B}^{-1}(\mathbf{x}_0-\mathbf{x}_0^b) + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}(\mathbb{H}(\mathbf{x}_t)-\mathcal{Y}_t))dt, \tag{2.34}$$

and the minimizer reads

$$\hat{\mathbf{x}}_0 = -\mathcal{H}^{-1}\partial_{\mathbf{x}}J(0), \tag{2.35}$$

where $\mathcal{H}$ is the Hessian matrix gathering the cost function second derivatives:

$$\mathcal{H} = \mathbf{B}^{-1} + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t dt. \tag{2.36}$$

We have shown through Eq.(1.50) that when the measurement and the dynamics operators are both linear, the objective function is convex. The estimate, if it is accurately computed, corresponds thus to the unique global minimum. Denoting the true state $\mathbf{x}_0^t$, we have $\nabla J(\hat{\mathbf{x}}_0 - \mathbf{x}_0^t + \mathbf{x}_0^t) = 0 \Leftrightarrow$

$$(\mathbf{B}^{-1} + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t dt)(\mathbf{x}_0^t-\hat{\mathbf{x}}_0) \approx$$
$$\mathbf{B}^{-1}(\mathbf{x}_0^t-\mathbf{x}_0^b) + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*(\mathbf{x}_0)\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}(\mathbb{H}(\varphi_t(\mathbf{x}_0^t))-\mathcal{Y}(t))dt, \tag{2.37}$$

where a linearization around the true state, $\mathbf{x}^t$, of the cost function gradient at the minimum point has been performed. Note that a strict equality only applies for linear operator. Multiplying the left hand and right hand terms by their transpose expression, taking the expectation and assuming the background errors $\mathbf{x}_0^b - \mathbf{x}_0^t$ and the innovation $\mathcal{Y}(t) - \mathbb{H}(\mathbf{x}^t(t))$ are uncorrelated, we get, the classical result:

$$\mathbf{P}_0 \approx \left(\mathbf{B}^{-1} + \int_{t_0}^{t_f} \partial_{\mathbf{x}} \varphi_t^* \partial_{\mathbf{x}} \mathbb{H}^* \mathbf{R}^{-1} \partial_{\mathbf{x}} \mathbb{H} \partial_{\mathbf{x}} \varphi_t dt \right)^{-1} = \mathcal{H}^{-1}, \tag{2.38}$$

which states that the initial error covariance matrix $\mathbf{P}_0 = \mathbb{E}((\mathbf{x}_0^b - \mathbf{x}_0^t)(\mathbf{x}_0^b - \mathbf{x}_0^t)^T)$ is given by the inverse Hessian matrix of the functional. This matrix is usually called the analysis covariance matrix, and is specifically denoted by a superscript "a". From distribution Eq.(2.31), for a volume preserving transformation, we observe immediately that an optimal initial value supplies an optimal trajectory that maximizes the *posterior* distribution. Thus, at a given time, the analysis error covariance reads (assuming without loss of generality a zero mean):

$$\mathbf{P}_t^a = \int (\varphi_t(\mathbf{x}_0^b - \mathbf{x}_0^t)\varphi_t^T(\mathbf{x}_0^b - \mathbf{x}_0^t)) p(\mathbf{x}_t | \mathcal{Y}_{t_f}, \ldots, \mathcal{Y}_{t_0}, \mathbf{x}_{t_0}^b) d\mathbf{x}_t. \tag{2.39}$$

If the dynamical model is linear, $\varphi_t(\mathbf{x}_0) = \Phi_t \mathbf{x}_0$, we then get:

$$\mathbf{P}_t^a = \Phi_t \mathbf{P}_0^a \Phi_t^T, \tag{2.40}$$

which describes the way the errors are propagated along time in variationnal data assimilation. This formula also unveils a recursion, involving a forecast state $\bar{\mathbf{x}}_t^f = \Phi_t \hat{\mathbf{x}}_{t-dt}$ and a forecast covariance matrix $\mathbf{P}_t^f = \Phi_t \mathbf{P}_{t-dt}^a \Phi_t^T$. In the linear case, with an additive stationary noise, the update expression can be written as Eq.(1.36), (1.37) and (1.39) respectively. They indeed correspond to the recursive Kalman filter formulation (associated to a noise free dynamics) given in algorithm 1.

2.2.3 Functional minimization

Due to the dimension of the state space, the minimization of the function (2.30) requires implementing an iterative optimization strategy. The most efficient optimization procedure needs to evaluate the functional gradient at several points. This adjoint backward dynamics in Eq.(2.15) provides the gradient functional at the initial time:

$$\partial_{\mathbf{x}_0} J(\mathbf{x}_0) = -\Lambda(t_0) + \mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_0^b). \tag{2.41}$$

Many iterative optimization schemes can be used to serve this purpose. This includes as a non limitative list: conjugate gradient methods and quasi-Newton methods (Nocedal and Wright, 2004). We will present here a simple quasi-Newton optimization procedure where the estimate at iteration $\mathbf{x}_0^{k+1}$ is updated as:

$$\mathbf{x}_0^{k+1} = \mathbf{x}_0^k - \alpha_n \tilde{\mathcal{H}}_{\mathbf{x}_0^k}^{-1} \partial_{\mathbf{x}_0} J(\mathbf{x}_0^k), \tag{2.42}$$

where $\tilde{\mathcal{H}}_{\mathbf{x}_0^n}^{-1}$ denotes an approximation of the Hessian inverse computed from the functional gradient with respect to $\mathbf{x}_0^k$; the constant α_k is chosen so that the Wolfe conditions are respected. This constitutes the standard 4DVar data assimilation procedure. An algorithmic synopsis of the 4DVar is described in algorithm 2.

Note that the construction of the background error covariance matrix $\mathbf{B}$ and the observation error covariance matrix $\mathbf{R}$ is not a trivial task. The detailed illustration of related techniques are presented in chapter 5.

Algorithm 2 Variational data assimilation algorithm

procedure ANALYSIS
  Set an initial condition: $\mathbf{x}_0^0 = \mathbf{x}_0^b$, given convergence toleration ϵ.
  Compute $\mathbf{x}_t$ with the forward integration of relation (1.1)
  Solve for the initial condition $\mathbf{x}_0$ which minimize the problem (2.30) with $\mathbf{x}_0^b, \mathcal{Y}, \mathbf{B}, \mathbf{R}$ known.
  for $k = 1 : k_{max}$ **do**
    Compute the gradient by Eq.(2.41) based on the prior evaluation of adjoint variables driven by Eq.(2.15),
    Iteratively searching for optimizer based on (2.42),
    Check convergence condition: $\|\mathbf{x}_0^{k+1} - \mathbf{x}_0^k\| < \epsilon$,
  end for
end procedure

2.3 Incremental 4D variational data assimilation

2.3.1 The problem

When the operators involved are nonlinear, the variational assimilation procedure can be improved by introducing a nonlinear least squares procedure in the same spirit as a Gauss-Newton incremental strategy Courtier et al. (1994). This optimization strategy consists in performing a linearization of the dynamics around the current trajectory and operating the optimization with respect to an incremental solution. Instead of correcting directly the initial state $\mathbf{x}(t_0)$ as in the previous subsection, the incremental approach consists in correcting an increment, $\delta\mathbf{x}_0$, that evolves according to the tangent dynamics equation computed around a given state $\mathbf{x}$:

$$\begin{cases} \partial_t \delta\mathbf{x}(\mathrm{x}, t) + \partial_{\mathrm{x}} \mathbb{M}(\mathbf{x})\delta\mathbf{x}(x, t) = 0, \\ \delta\mathbf{x}(\mathrm{x}, 0) = \mathbf{x}_0^b - \mathbf{x}_0 + \boldsymbol{\eta}. \end{cases} \tag{2.43}$$

The cost function in terms of the increment $\delta\mathbf{x}_0$ is consequently defined as:

$$J(\delta\mathbf{x}_0) = \frac{1}{2}\|\delta\mathbf{x}_0\|^2_{\mathbf{B}} + \frac{1}{2}\int_{t_0}^{t_f} \|\mathbb{H}(\mathbf{x}(\mathrm{x}, t)) - \mathcal{Y}(\mathrm{x}, t)\|^2_{\mathbf{R}} dt, \tag{2.44}$$

where

$$\mathbf{x}_t = \varphi_t(\mathbf{x}_b) + \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_b)\delta\mathbf{x}_0, \text{ and } \delta\mathbf{x}_0 = \delta\mathbf{x}(\mathrm{x}, 0) - (\mathbf{x}_0^b - \mathbf{x}_0),$$

or alternatively,

$$J(\delta\mathbf{x}_0) = \frac{1}{2}\|\delta\mathbf{x}_0\|^2_{\mathbf{B}} + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \delta\mathbf{x}(\mathrm{x}, t) - \mathbf{D}(\mathrm{x}, t)\|^2_{\mathbf{R}} dt, \tag{2.45}$$

where

$$\delta\mathbf{x}_t = \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_b)\delta\mathbf{x}_0, \text{ and } \delta\mathbf{x}_0 = \delta\mathbf{x}(\mathrm{x}, 0) - (\mathbf{x}_0^b - \mathbf{x}_0),$$

and where the innovation vector $\mathbf{D}(\mathrm{x}, t)$ is defined as:

$$\mathbf{D}(\mathrm{x}, t) = \mathcal{Y}(\mathrm{x}, t) - \mathbb{H}(\varphi_t(\mathbf{x}_b)). \tag{2.46}$$

2.3.2 Adjoint equation

The adjoint system associated to this functional is directly inferred from the previous one and reads:

$$\begin{cases} -\partial_t \Lambda(t) + (\partial_{\mathbf{x}} \mathbb{M})^* \, \Lambda(t) = (\partial_{\mathbf{x}} \mathbb{H})^* \, \mathbf{R}^{-1} (\partial_{\mathbf{x}} \mathbb{H} \, \partial_{\mathbf{x}} \varphi_t(\mathbf{x}_0) \delta \mathbf{x}_0 - \mathbf{D}(\mathrm{x}, t)) \\ \Lambda(t_f) = 0, \end{cases} \tag{2.47}$$

with the gradient functional at the initial time given by,

$$\partial_{\delta \mathbf{x}_0} J(\delta \mathbf{x}_0) = -\Lambda(t_0) + \mathbf{B}^{-1} \delta \mathbf{x}_0. \tag{2.48}$$

2.3.3 Outer loop and inner loop

We note that the incremental functionals (2.44) and (2.45) correspond to a linearization around a given background solution of the innovation term. It is now quadratic and thus, has a unique minimum. This tangent linear dynamic model and the linear tangent observation operator have some advantages. For example, the product between the operator $\partial_{\mathbf{x}} \mathbb{M}$ and the increment is simply a matrix multiplication. However, the linear tangent model is only an approximation of the non-linear model, we must always be cautious on the validity conditions of the linear tangent model. It is obvious that if the $\mathbf{x}_t$ that we are looking for differs greatly from the background $\mathbf{x}_b$, the linear hypothesis may not hold anymore. Thus Courtier et al. (1994) proposed to use two interleaved loops to incorporate the evolution of the background solution through the nonlinear dynamics, while keeping the simplicity of the internal loop to recover an optimal increment driven by the tangent linear dynamics. The idea simply states: in each outer loop, we redefine the initial condition, and the innovation vector is computed based on the new trajectory integrated from the modified initial condition with the complete non-linear model; in the inner loop, the tangent dynamic model is linearized around the new model trajectory. The computed initial increment allows us to update the current background initial solution. Let us note that in practice, due to the high computational cost associated with the dynamical models in geophysical applications, only a low number of external iteration are performed.

This procedure considers a correction term $\mathbf{x}_0^{k-1} - \mathbf{x}_0^b$ to the energy function's background error term. This correction term constrains the whole initial condition and makes the background discrepancy term consistent with the background error term denoted by $\boldsymbol{\eta}$. (see algorithm 3).

Different variants of the incremental technique can be defined depending on how the increment within each outer loop is defined. In the previous case, $\delta \mathbf{x}_0^k$ is defined as the difference of the initial condition between two consecutive outer loops. Another possibility to define the increment $\delta \mathbf{x}_0^k$ consists in keeping the difference between the initial condition and the background state. This way of proceeding requires to add a correction term to the innovation term, as shown in Weaver et al. (2003) (algorithm 4). It can be easily shown that these two choices are indeed equivalent.

In our case, we advocate for a slightly different approach. As a matter of fact, in this thesis we aim at focusing on cases for which only a background of bad quality is available, along with a badly known background error covariance matrix. Such case typically occurs in image based assimilation problems in which the background state is often given by a noisy image with possibly large areas of missing data. Hence, here the background error covariance matrix $\mathbf{B}$ is supposed to be a function of the outer loop index k:

$$\mathbf{B}^k = \overline{(\mathbf{x}_0^{k-1} - \mathbf{x}^{ref})(\mathbf{x}_0^{k-1} - \mathbf{x}^{ref})^T},$$

Algorithm 3 Incremental 4DVar: Courtier et al. (1994) Nested Loops Algorithm

procedure ANALYSIS
  Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, given convergence toleration ξ and ϵ.
Outer loop:
  for $k = 1 : k_{max}$ **do**
    Compute the innovation vector, $\mathbf{D}_t^k = \mathcal{Y}_t - \mathbb{H}(\varphi_t(\mathbf{x}_0^{k-1}))$,
    Inner loop: Solve for the initial condition increment $\delta\mathbf{x}_0$ which minimizes the problem

$$J(\delta\mathbf{x}_0^k) = \frac{1}{2}\|\delta\mathbf{x}_0^k + \mathbf{x}_0^{k-1} - \mathbf{x}_0^b\|_{\mathbf{B}}^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \delta\mathbf{x}_t^k - \mathbf{D}_t^k\|_{\mathbf{R}}^2 dt,$$
$$\delta\mathbf{x}_t^k = \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0^{k-1})\delta\mathbf{x}_0^k,$$

    while $\|\partial_{\delta\mathbf{x}_0^{k,n}} J(\delta\mathbf{x}_0^{k,n})\| > \epsilon$ **do**
      an Iterative gradient descent based on (2.42) or other descent method (LBFGS, algorithm 5), where the gradient is computed by

$$\partial_{\delta\mathbf{x}_0^{k,n}} J(\delta\mathbf{x}_0^{k,n}) = -\Lambda(t_0) + (\mathbf{x}_0^{k-1,n} - \mathbf{x}_0^b)^T\mathbf{B}^{-1}\delta\mathbf{x}_0, \tag{2.49}$$

      and from the prior evaluation of the adjoint variables driven by Eq.(2.47),
    end while
    $\mathbf{x}_0^k = \mathbf{x}_0^{k-1} + \delta\mathbf{x}_0^k$,
    Check convergence condition $\|\delta\mathbf{x}_0^k - \delta\mathbf{x}_0^{k-1}\| > \xi$,
  end for
end procedure

where the operator $\overline{f(t)}$ takes the mean value of $f(t)$ and for the first outer loop $\mathbf{x}_0^0 = \mathbf{x}_0^b$. This approach will be elaborated in the next section combined with the preconditioning technique.

Algorithm 4 Incremental 4DVar: Weaver et al. (2003) Nested Loops Algorithm

procedure ANALYSIS
  Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, given convergence toleration ξ and ϵ.
Outer loop:
  for $k = 1 : k_{max}$ **do**
    $\mathbf{D}_t^k = \mathcal{Y}_t - \mathbb{H}(\varphi_t(\mathbf{x}_0^{k-1})) + \partial_{\mathbf{x}}\mathbb{H}\ \delta\mathbf{x}_t^{k-1}$,
    Inner loop: Solve for the initial condition increment $\delta\mathbf{x}_0$ which minimize the problem

$$J(\delta\mathbf{x}_0^k) = \frac{1}{2}\|\delta\mathbf{x}_0^k\|_{\mathbf{B}}^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \delta\mathbf{x}_t^k - \mathbf{D}_t^k\|_{\mathbf{R}}^2 dt,$$
$$\delta\mathbf{x}_t^k = \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0^{k-1})\delta\mathbf{x}_0,$$

    while $\|\partial_{\delta\mathbf{x}_0^{k,n}} J(\delta\mathbf{x}_0^{k,n})\| > \epsilon$ **do**
      an Iterative gradient descent based on (2.42) or other descent method (LBFGS, algorithm 5), where the gradient is computed by

$$\partial_{\delta\mathbf{x}_0^{k,n}} J(\delta\mathbf{x}_0^{k,n}) = -\Lambda(t_0) + \mathbf{B}^{-1}\delta\mathbf{x}_0. \tag{2.50}$$

    and from the prior evaluation of the adjoint variables driven by Eq.(2.47),
    end while
    $\mathbf{x}_0^k = \mathbf{x}_0^b + \delta\mathbf{x}_0^k$,
    Check convergence condition $\|\delta\mathbf{x}_0^k - \delta\mathbf{x}_0^{k-1}\| > \xi$,
  end for
end procedure

2.3.4 Functional minimization

We briefly describe in this section the minimization procedure on which we rely in the inner loop of algorithm 3, 4 and 6. The LBFGS (Limited memory BFGS) method (Nocedal and Wright, 2004) is a limited-memory quasi-Newton method based on the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method. Unlike its parent method, BFGS, which needs to update the full inverse Hessian matrix at each iteration based on the current cost function curvature and the inverse Hessian matrix constructed from the previous iteration, the LBFGS method only stores few vectors of the size of the full Hessian to approximate the Hessian matrix. Therefore LBFGS is an ideal optimization strategy to deal with high dimensional problem. Another advantage of LBFGS method is related to the ensemble-based method in which the approximated inverse Hessian matrix can be used to update the analysis error covariance matrix. We will discuss more thoroughly this point in chapter 5. A schematic representation of the LBFGS method is described in algorithm 5. A more detailed presentation is given in chapter 9 of Nocedal and Wright (2004).

Algorithm 5 Limited Memory BFGS Algorithm

1: **procedure** LBFGS
2: Prescribe initial increment $\delta\mathbf{x}_0^0$ and convergence tolerance ϵ
3: Prescribe initial inverse Hessian approximation $\mathcal{H}_{\mathbf{x}_0^0}^{-1}$, usually diagonal.
4: **while** $\|\partial_{\delta\mathbf{x}_0^n} J(\delta\mathbf{x}_0^n)\| > \epsilon$ **do**
5: Searching for new point based on the step α^n and the direction $-\alpha^n\mathcal{H}^n\partial_{\delta\mathbf{x}_0^n}J(\delta\mathbf{x}_0^n)$

$$\delta\mathbf{x}^{n+1} = \delta\mathbf{x}^n - \alpha^n\mathcal{H}^n\partial_{\delta\mathbf{x}_0^n}J(\delta\mathbf{x}_0^n), \tag{2.51}$$

6: Update the inverse Hessian matrix,

$$\mathcal{H}_{\delta\mathbf{x}_0^n}^{-1} = (V^n)^T\mathcal{H}_{\delta\mathbf{x}_0^n}^{-1}V^n + \rho^n s^n(s^n)^T, \tag{2.52}$$

where

$$\begin{aligned}
\rho^n &= \frac{1}{y^{nT}s^n},\\
V^n &= I - \rho^n y^n(s^n)^T,\\
s^n &= \delta\mathbf{x}^{n+1} - \delta\mathbf{x}^n,\\
y^n &= \partial_{\delta\mathbf{x}_0^{n+1}}J(\delta\mathbf{x}_0^{n+1}) - \partial_{\delta\mathbf{x}_0^n}J(\delta\mathbf{x}_0^n)
\end{aligned}$$

7: **end while**
8: **end procedure**

2.3.5 Preconditioning and conditioning of the incremental assimilation system

In this section we will show the condition number of the Hessian of the cost function (2.45) plays an important role in formulating the increment solution.

The minimization of the cost function (2.45) comes to solve a linear system

$$\mathcal{H}\delta\mathbf{x}_0 = b. \tag{2.53}$$

Such an expression is obtained by canceling the gradient of (3.31) with respect to the initial increment $\mathbf{x}_0$,

$$\partial_{\delta\mathbf{x}_0}J(\delta\mathbf{x}_0) = \mathbf{B}^{-1}\delta\mathbf{x}_0 + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}(\partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t\delta\mathbf{x}_0 - \mathbf{D}(t))dt = 0, \tag{2.54}$$

$$\Longleftrightarrow\ (\mathbf{B}^{-1} + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t dt)\delta\mathbf{x}_0 = \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\mathbf{D}(t)dt \tag{2.55}$$

and letting

$$\mathcal{H} = \mathbf{B}^{-1} + \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t dt, \tag{2.56}$$

$$b = \int_{t_0}^{t_f}\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\mathbf{D}(t)dt. \tag{2.57}$$

The possibly ill-conditioned nature of such a system depends on the condition number of the Hessian matrix. The condition number is given by the ratio of the largest eigenvalue over the minimum eigenvalue. The larger the condition number is, the more sensitive the system is to errors both in the b vector and in the estimate. In practice, the system 2.53 is never solved directly. An iterative method is always used. However, large condition number also slows down the inner loop convergence rate (Haben et al., 2011a).

A known solution to improve a badly-conditioned system, consists in solving an equivalent system with a lower condition number. Preconditioning constitutes a practical procedure to reach that goal. A common procedure relied on applying a change of variable consists in setting:

$$\delta \mathbf{x}_t = \mathbf{U} \delta \mathbf{z}_t, \tag{2.58}$$

such that the new Hessian matrix, $\widetilde{\mathcal{H}} = \mathbf{U}^{\frac{1}{2}^T} \mathcal{H} \mathbf{U}^{\frac{1}{2}}$, possesses a lower condition number.

Note that the original Hessian is here symmetric positive definite,

$$\mathcal{H} = \mathcal{H}^{\frac{1}{2}^T} \mathcal{H}^{\frac{1}{2}}, \;\; \mathcal{H}^{\frac{1}{2}} = \mathbf{B}^{-\frac{1}{2}} + \int_{t_0}^{t_f} \mathbf{R}^{-\frac{1}{2}} \partial_X \mathbb{H} \partial_{\mathbf{x}} \varphi_t dt. \tag{2.59}$$

We see immediately that if $\mathbf{U}$ is set to the inverse of square root of the Hessian matrix, then the original linear system is solved in a single step. This approach, which leads to the lowest unity condition number, is obviously unpractical since it requires solving the system. In variational assimilation, a common procedure consists to operate a preconditioning through the background correlation matrix. This is called the control variable transform(CVT):

$$\delta \mathbf{x}_t = \mathbf{B}^{\frac{1}{2}} \delta \mathbf{z}_t. \tag{2.60}$$

Note that the Hessian has a first component that is the background information matrix. This background covariance matrix can be thus considered as a rough approximation of the Hessian inverse. To highlight its role on the Hessian conditioning, the condition number of the Hessian matrix can be computed in a simple case. Let us assume in the following that the measurements noise are independent and spatially identically distributed with a constant covariance $\mathbf{R} = \sigma \mathbb{I}$ (and σ constant). We assume also that the measurements and the state variable live in the same space ($\mathbb{H} = \mathbb{I}$). The Hessian condition number can be bounded in the matrix 2-norm by:

$$\begin{aligned} \|\mathcal{H}\|_2 &\leq \lambda_{\max}(\mathbf{B}^{-1}) + \sigma^{-1} \int_{t_0}^{t_f} \|C_t\|_2, & (2.61) \\ &\leq \lambda_{\max}(\mathbf{B}^{-1}) + \sigma^{-1} \max_{t \in [t_0 t_f]} (\lambda_{\max}(C_t))(t_f - t_0), & (2.62) \end{aligned}$$

where $\lambda_{\max}(A)$ is the largest real eigenvalue of a symmetric matrix A and $C_t = \partial_{\mathbf{x}} \varphi_t^T \partial_{\mathbf{x}} \varphi_t$. As matrix C_t is in general rank deficient, its lower eigenvalue is zero and we obtain immediately a lower bound of the Hessian 2-norm:

$$\|\mathcal{H}\|_2 \geq \lambda_{\min}(\mathbf{B}^{-1}). \tag{2.63}$$

These two bounds provide a bound on the Hessian condition number:

$$\begin{aligned} \kappa(\mathcal{H}) &\leq \frac{\lambda_{\max}(\mathbf{B}^{-1})}{\lambda_{\min}(\mathbf{B}^{-1})} (1 + \frac{\max_{t \in [t_0 t_f]} (\lambda_{\max}(C_t))(t_f - t_0)}{\sigma \lambda_{\max}(\mathbf{B}^{-1})}), \\ \kappa(\mathcal{H}) &\leq \kappa(\mathbf{B})(1 + \sigma^{-1} \lambda_{\min}(\mathbf{B}) \max_{t \in [t_0 t_f]} (\lambda_{\max}(C_t))(t_f - t_0)). \end{aligned} \tag{2.64}$$

We observe that this bound depends directly on the background matrix. An ill-conditioned background matrix is likely to lead to an ill-conditioned Hessian. The Hessian conditioning bound depends also on the larger eigenvalue of matrix C_t, which can be related to the maximum finite time Lyapunov exponent on the flow domain and over the assimilation interval. Dynamics exhibiting high stretching integrated on large temporal window leads potentially to an ill-conditioned Hessian. Non noisy perfect measurements are also a potential source of bad conditioning. This counter intuitive situation can be well understood through the conditional pdf given in Eq.(2.31). As a matter of fact, considering a perfect measurement with a variance tending to zero, gives a likelihood that tends to a Dirac function. The log posterior pdf (2.31) is:

$$\int_{t_0}^{t_f} \delta(\mathcal{Y}_t - \varphi_t(\mathbf{x}_{t_0}))P(\varphi_t(\mathbf{x}_{t_0})|\mathbf{x}^b_{t_0})dt. \tag{2.65}$$

This pdf is maximal for the optimal solution and zero otherwise. If the measurements corresponds exactly to the model dynamics – which is never the case in practice – the optimal trajectory is given by $\varphi_t(\mathcal{Y}_{t_0})$. However, a tiny perturbation of the initial condition cancels the pdf and makes the estimation highly unstable.

Considering now a preconditioning with the background covariance, we get a modified Hessian:

$$\widetilde{\mathcal{H}} = \mathbb{I} + \int_{t_0}^{t_f} \mathbf{B}^{\frac{1}{2}T}\partial_X\varphi_t^*\partial_X\mathbb{H}^*\mathbf{R}^{-1}\partial_X\mathbb{H}\partial_{\mathbf{x}}\varphi_t\mathbf{B}^{\frac{1}{2}}dt. \tag{2.66}$$

With exactly the same assumption as previously, we obtain a tighter bound for the modified Hessian:

$$\kappa(\widetilde{\mathcal{H}}) \leq (1 + \sigma^{-1}\lambda_{\max}(\mathbf{B}) \max_{t\in[t_0t_f]}(\lambda_{\max}(C_t))(t_f - t_0)). \tag{2.67}$$

The background conditioning forms a potentially good candidate for the system preconditioning. We used here simplified assumptions for the observation operator and for the measurement noise. However, similar bounds can be established for a more general model assuming the observation space is of lower dimension than the state space (Haben et al., 2011b). As shown in this section, the background covariance matrix plays a central role. In the chapter 5, we describe a technique in which this covariance matrix is empirically determined based on preconditioning form of cost function.

2.4 Preconditioned incremental form of cost function

In section 2.3.5 we discussed the general idea of Hessian conditioning. The preconditioning of variational system is generally performed through a change of variable with the matrix square-root of the background error covariance matrix $\mathbf{B}$, which leads to a new Hessian matrix, $\widetilde{\mathcal{H}} = \mathbf{B}^{\frac{1}{2}T}\mathcal{H}\mathbf{B}^{\frac{1}{2}}$ with a lower condition number. Applying the CVT (2.60) to the function (2.45), we get a modified objective function,

$$J(\delta\mathbf{z}_0) = \frac{1}{2}\|\delta\mathbf{z}_0\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{B}^{\frac{1}{2}}\delta\mathbf{z}_0 - \mathbf{D}(t,x)\|^2_{\mathbf{R}}dt. \tag{2.68}$$

This modified cost function removes $\mathbf{B}^{-1}$ from the background term. Hence no cross-variable correlation is anymore involved in the control vector $\delta\mathbf{z}_0$. Note this change of variable can also be viewed as a whitening filtering of the background error. Although better conditioned, the resulting system remains in general difficult to solve and requires

the use of the adjoint minimization setups. The associated adjoint equation is similar to Eq.(2.47),

$$\begin{cases} -\partial_t \Lambda(t) + (\partial_{\mathbf{x}} \mathbb{M})^* \, \Lambda(t) = (\partial_{\mathbf{x}} \mathbb{H})^* \, \mathbf{R}^{-1} (\partial_{\mathbf{x}} \mathbb{H} \, \partial_{\mathbf{x}} \varphi_t(\mathbf{x}_0) \mathbf{B}^{\frac{1}{2}} \delta \mathbf{z}_0 - \mathbf{D}(t, \mathrm{x})) \\ \Lambda(t_f) = 0, \end{cases} \tag{2.69}$$

and the gradient of the cost function Eq.(2.68) with respect to $\delta \mathbf{z}_0$ is,

$$\partial_{\delta \mathbf{z}_0} J(\delta \mathbf{z}_0) = -\mathbf{B}^{\frac{1}{2}T} \Lambda(t_0) + \delta \mathbf{z}_0, \tag{2.70}$$

We have mentioned in previous section that we advocate for an update the background error covariance matrix along with the outer loops. Generally this update of the background error covariance between consecutive outer loops in terms of standard 4DVar method is not performed due to the high computational cost associated to such updating. However, as in this thesis, we will focus on an empirical, ensemble, low rank approximation of the background error covariance, an efficient technique for such a reactualization will be plausible. Especially since we are dealing with the preconditioned form of cost function, the update of the background error covariance term $\mathbf{B}$ actually constitutes a change of the preconditioning matrix $\mathbf{B}^{\frac{1}{2}}$ revealed by Eq.(2.60). Similar to the original algorithm proposed in Courtier et al. (1994), the convergence of the increment sequence $\delta \mathbf{x}_0^k$ is not guaranteed.

A schematic representation of the proposed strategy is shown in algorithm 6. We will see in the next section how the update is done by using the traditional 'NMC' method. A more relevant update scheme associated with the ensemble approach will be introduced in chapter 5.

We will come back on the necessities of updating the background error covariance matrix in chapter 6.

2.5 The background error covariance B

The background error covariance $\mathbf{B}$ plays a central role in formulating the analysis. In this section we present several tracks to model this covariance matrix. In order to make the following chapters self-explanatory, we hereby concentrate on three aspects of the $\mathbf{B}$ matrix: first of all, the interpretation of the role of the $\mathbf{B}$ matrix in DA process; secondly, the evaluation of the $\mathbf{B}$ matrix in typical operational environment; finally, the implicit evolvement of $\mathbf{B}$. This section is mainly based on the presentation provided in Bannister (2008a,b).

2.5.1 Definition of the B matrix

The concept of the background error matrix $\mathbf{B}$ has been introduced in section 1.3.3. The definition of $\mathbf{B}$ is given by,

$$\mathbf{B} = \overline{(\mathbf{x}^b - \mathbf{x}^t)(\mathbf{x}^b - \mathbf{x}^t)^T}, \tag{2.71}$$

where $\mathbf{x}^b$ is the background state, $\mathbf{x}^t$ is the true state and overbar operator suggests the mathematical expectation. Ideally this matrix should be equivalent to the forecast error covariance matrix defined in the method of Kalman filter, and the propagation of error covariance matrix should be given Eq. (1.33) for linear models as,

$$\mathbf{B} = \mathbf{M}\mathbf{P}^{\mathbf{a}}\mathbf{M}^{\mathbf{T}} + \mathbf{Q}, \tag{2.72}$$

Algorithm 6 Preconditioned incremental 4DVar: Precondition Updating Nested Loops Algorithm

procedure ANALYSIS
  Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, given convergence toleration ξ and ϵ.
Outer loop:
  for $k = 1 : k_{max}$ **do**
    $\mathbf{D}_t^k = \mathcal{Y}_t - \mathbb{H}(\varphi_t(\mathbf{x}_0^{k-1}))$,
    Update the square root of the background error covariance matrix $\mathbf{B}^{\frac{1}{2},k}$.
    Initialize the increment vector $\delta\mathbf{x}_0^k$ and do an inverse control variable transformation $\delta\mathbf{z}_0^k = (\mathbf{B}^{\frac{1}{2},k})^{-1}\delta\mathbf{x}_0^k$.
    Inner loop: Solve for the initial condition increment $\delta\mathbf{z}_0$ which minimize the problem

$$J(\delta\mathbf{z}_0^k) = \frac{1}{2}\|\delta\mathbf{z}_0^k\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0^{k-1})\mathbf{B}^{\frac{1}{2},k}\delta\mathbf{z}_0 - \mathbf{D}_t^k\|_{\mathbf{R}}^2 dt,$$

    while $\|\partial_{\delta\mathbf{z}_0^{k,n}} J(\delta\mathbf{z}_0^{k,n})\| > \epsilon$ **do**
      an Iterative gradient descent based on (2.42) or other descent method (LBFGS, algorithm 5).
    end while
    $\mathbf{x}_0^k = \mathbf{x}_0^{k-1} + \mathbf{B}^{\frac{1}{2},k}\delta\mathbf{z}_0^k$,
    Check convergence condition $\|\delta\mathbf{x}_0^k - \delta\mathbf{x}_0^{k-1}\| > \xi$,
  end for
end procedure

where $\mathbf{P^a}$ is the analysis error covariance matrix and $\mathbf{Q}$ is the model error covariance matrix. Note that in Kalman filter, the matrix $\mathbf{B}$ is associated to Gaussian distributed background errors, whereas in variational methods, a much more broader type of error statistical distribution law could be applied. However, in most cases, we tend to have a Gaussian distributed random error fields due to its simplicity. Besides, having the same structure of error covariance as in the Kalman filtering offers us the possibility to compare performance of both methods.

2.5.2 Interpretation of the role of the B matrix

The relation between the analysis and the $\mathbf{B}$ matrix can be revealed by Eq.(1.27):

$$\mathbf{x}^a = \mathbf{x}^b + \mathbf{B}\mathbf{H}^T(\mathbf{H}\mathbf{B}\mathbf{H}^T + \mathbf{R})^{-1}(\mathcal{Y} - \mathbf{H}\mathbf{x}^b), \tag{1.27}$$

This expression is valid for a static case (dynamic operator fixed to identity). By assuming a case with a single observation with an identity observation operator as well, a simplified version can be formulated as Bannister (2008a):

$$\mathbf{x}^a_l = \mathbf{x}^b_l + B_{lk}\frac{\mathbf{x}_{obs} - \mathbf{x}^b_k}{B_{kk} + \sigma^2_{obs}}, \tag{2.73}$$

where the observation is available as $\mathbf{x}_{obs}$ positioned at the kth element of the state vector with error σ_{obs}. The coefficient B_{kk} is the variance of the kth element while B_{lk} contains the value of the covariance between the kth and the lth elements. A special case of Eq.(2.73) when $l = k$ reads:

$$\mathbf{x}^a_k = \mathbf{x}^b_k + B_{kk}\frac{\mathbf{x}_{obs} - \mathbf{x}^b_k}{B_{kk} + \sigma^2_{obs}}, \tag{2.74}$$

Both equations 2.73 and 2.74 show how the observation information pass to the analysis through the $\mathbf{B}$ matrix. In the latter case, we can directly infer that: if $B_{kk} \ll \sigma^2_{obs}$, then $\mathbf{x}^a_k \approx \mathbf{x}^b_k$; if $B_{kk} \gg \sigma^2_{obs}$, then $\mathbf{x}^a_k \approx \mathbf{x}_{obs}$. This conclusion can be interpreted in an intuitive way: when $B_{kk} \ll \sigma^2_{obs}$, we have a much smaller background error than the observation error which suggests we should have more confidence in the background state and *vice versa*. In the former case when $l \neq k$, the equation (2.73) shows how the innovation vector at the kth element propagates itself to the lth element. This 'non-local' effect of $\mathbf{B}$ is usually used to justify the importance of the off-diagonal terms of $\mathbf{B}$. In the 'NMC' method listed below, the off-diagonal terms are represented by a spatial correlation matrix.

2.5.3 Evaluation of the B matrix: 'NMC' method

To calculate precisely the covariance matrix, one needs to know the true state $\mathbf{x}^t$ and the *a priori* probability density function of $\mathbf{x}_b$. However, both of these terms are usually unknown, so the $\mathbf{B}$ matrix must be approximated in some way. Bannister (2008a) pointed out that in operational implementation, the $\mathbf{B}$ matrix is a rough approximation of the $\mathbf{P}^f$, and it is quite immediate to think that any improvement in the approximation of matrix $\mathbf{B}$ should lead to a better analysis performance. A traditional way of measuring the background error covariance is the 'NMC' method initially proposed by Parrish and Derber (1992) and later polished by several articles (Derber and Bouttier, 1999).

In this method, the structure function of the $\mathbf{B}$ matrix is defined in a generic way as,

$$\mathbf{B} = KD^{\frac{1}{2}}FF^TD^{\frac{1}{2}}K^T, \tag{2.75}$$

where K is the linear balance operator, which defines the multivariate covariance based on a diagonal covariance matrix; D gathering the variance of the background error. Matrix FF^T is a spatial correlation operator. The square root of the background error covariance matrix $\mathbf{B}^{\frac{1}{2}}$ is directly given by:

$$\mathbf{B}^{\frac{1}{2}} = KD^{\frac{1}{2}}F, \tag{2.76}$$

$\mathbf{B}^{\frac{1}{2}}$ can be employed efficiently in the method of CVT introduced in section 2.3.5. By doing so, the full rank $\mathbf{B}$ matrix no longer needs to be stored. Instead, we only need to store the $\mathbf{B}^{\frac{1}{2}}$ that can be constructed in a compact way as we shall see it.

The employment of the balance operator matrix K is based on the assumption that the state variables can be splited into two parts: one balanced part with respect to a given equilibrium relation and one imbalanced part. Such balance are essential in geophysics. For example, considering for example a simple 1D shallow water case in which the geostrophic relationship holds between the free surface height and the velocity,

$$\begin{cases} h = h, \\ v = v_b + v_u = K_{vh}h + v_u, \end{cases} \tag{2.77}$$

where we take h as the reference, and split u into two parts: the balanced part v_b could be recovered by the height field with the geostrophic relationship, plus an unbalanced part v_u that needs to be evaluated. In this case, $K_{vh} = -\frac{g}{f}$.

We can define the error covariance matrix of the unbalanced parts as B_u. It is a block diagonal matrix which has no cross-covariance between off diagonal blocks variables.

$$B_u = \begin{pmatrix} B_{hh} & 0 \\ 0 & B_{vv} \end{pmatrix}, \tag{2.78}$$

Matrix K reads,

$$K = \begin{pmatrix} I & 0 \\ K_{uh} & I \end{pmatrix}, \tag{2.79}$$

and B_{ii} can be expressed as,

$$B_{ii} = D_{ii}^{1/2} F F^T D_{ii}^{1/2}. \tag{2.80}$$

This matrix contains only the univariate variance and covariance, note that the covariance here are only spatially-related and do not depend on other variables. So here the question remains to find the auto covariance matrix B_{xx} composed of the variance matrix D and the correlation matrix $C = FF^T$.

The variance matrix D of background error is calculated from the mean of differences of several samples. Each sample is provided by different forecast states at the same time but integrated from a more or less distant past (usually 24/48 hrs forecasts).

The correlation matrix C concentrates all the information on spatial correlation associated with a single variable. In operational applications, the horizontal correlation and the vertical correlation are usually separated. The vertical correlation is usually easier to treat compared to the horizontal correlation as there are much fewer vertical levels than the horizontal nodes. Here we center on the formulation of the horizontal correlation.

In principle such spatial correlation matrix rely on the construction of compactly supported correlation functions $\mathcal{C}(\mathbf{x}, \mathbf{y})$ $(\mathbf{x}, \mathbf{y} \in \mathbb{R}^3)$. These functions are called compactly supported since their values reach zero if the distance between $\mathbf{x}$ and $\mathbf{y}$ exceeds the cutoff distance. Gaspari and Cohn (1999) proposed an efficient way of evaluating such functions.

The whole idea consists in looking for a homogeneous and isotropic function $\mathcal{C}$ as a self-convolution of compactly supported, radially symmetric functions B on $\mathbb{R}^3$. This is proved by theorems 3.a.1 and 3.a.3 in Gaspari and Cohn (1999). The construction of the radially symmetric functions $B(\mathbf{x}-\mathbf{y})$ is trivial task. In fact, B can be traced back to a even function $B_0(\|\mathbf{x}-\mathbf{y}\|)$ where $\|\bullet\|$ denotes the Euclidean distance and obviously $\|\mathbf{x}-\mathbf{y}\| \in \mathbb{R}$. A popular choice of B_0 as a continuous, piecewise linear function can produce a 5-th order function taking the form of equation (4.10) in Gaspari and Cohn (1999). The same correlation function is widely employed in the localization technique for ensemble Kalman filter method. See section 5.1.4 for its application in the ensemble-based variational method.

Apart from the method proposed by Gaspari and Cohn (1999), an alternative consists to rely on the diffusion equation to preclude the calculation associated with the convolution process. Weaver and Courtier (2001) use an explicit form to solve the diffusion equation, Mirouze and Weaver (2010) finds a bridge between the implicit form of the diffusion equation and the convolution of an auto regression function, while the integration of the implicit-solved diffusion equation is simpler than the convolution method in evaluating the correlation matrix. The diffusion equation has the capacity to adjust the length scale as well. The idea behind the implicit (or explicit Weaver and Courtier (2001)) resolution of the diffusion equation is that by doing a time discretization of the diffusion equation. The spatial correlation is provided through a pseudo-time interval. The physical meaning of the time quantity of the diffusion equation therefore mimics the length scale of the correlation function. The spatial derivation part of the diffusion equation is retained.

Now that we have all the components of the $\mathbf{B}^{\frac{1}{2}}$ matrix in Eq.(2.76), the analysis process can be adapted from the principles of algorithm 6 to include the preconditioned form Eq.(2.68). The proposed algorithm is listed in algorithm 7.

2.5.4 The implicit evolution of the B matrix in preconditioned incremental 4DVar

We take advantage of the nested loops scheme in the incremental 4DVar method to update the square root of the background error covariance matrix $\mathbf{B}^{\frac{1}{2},k}$ for each outer loop shown in step 5 of algorithm 7. One motivation for such an proposition is that the variational methods is incapable of producing the so-called *a posteriori* error covariance matrix, unlike Kalman filter methods. But this do not mean that the background error covariance matrix remains static, on the contrary, after having fixed the $\mathbf{B}$ matrix at the beginning of one data assimilation window, the background error covariance indeed evolves implicitly according to the tangent linear and adjoint models within the inner loop process. This is also outlined in Buehner et al. (2010a).

2.6 Summary

In this chapter we tried to present a thorough overview of variational methods. The adjoint equation technique which constitutes the keystone of variational systems and the estimation of parameter within a variational context have been first presented. The 4DVar, followed by the incremental 4DVar, have been also described. Two important facets of those techniques have been thoroughly described: the preconditioning technique and the background error covariance matrix.

Algorithm 7 Preconditioned incremental 4DVar data assimilation algorithm

procedure ANALYSIS

Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, given convergence toleration ξ and ϵ

Outer loop:

for $k = 1 : k_{max}$ **do**

Update the square root of the background error covariance matrix:

$$\mathbf{B}^{\frac{1}{2},k} = K(D^{\frac{1}{2}})^k F,$$

where

$$(D^{\frac{1}{2}})^k = \sqrt{diag((\mathbf{x}_0^{k-1} - \mathbf{x}^{ref})(\mathbf{x}_0^{k-1} - \mathbf{x}^{ref})^T)}$$

$\mathbf{D}_t^k = \mathcal{Y}_t - \mathbb{H}(\varphi_t(\mathbf{x}_0^{k-1}))$,

Do an inverse control variable transformation $\delta\mathbf{z_0} = \mathbf{B}^{-\frac{1}{2},k}\delta\mathbf{x}_0$,

Inner loop: Solve for the initial condition increment $\delta\mathbf{z}_0$ which minimizes the problem 2.68

while $\|\partial_{\delta\mathbf{z}_0^{k,n}} J(\delta\mathbf{z}_0^{k,n})\| > \epsilon$ **do**

an Iterative gradient descent based on (2.42) or other descent method (LBFGS, algorithm 5), where the gradient is calculated by Eq.(2.69) in which the adjoint variable Λ is driven by Eq.(2.70).

end while

Update control initial space $\mathbf{z}_0^k$,

$$\mathbf{z}_0^k = \mathbf{z}_0^{k-1} + \delta\mathbf{z}_0^k,$$

Do a control variable transformation

$$\delta\mathbf{x}_0^k = \mathbf{B}^{\frac{1}{2},k}\delta\mathbf{z}_0^k$$

Update the initial condition $\mathbf{x}_0^{b,k}$,

$$\mathbf{x}_0^k = \mathbf{x}_0^{k-1} + \delta\mathbf{x}_0^k,$$

Check convergence condition:

if $\|\delta\mathbf{x}_0^k - \delta\mathbf{x}_0^{k-1}\| < \xi$, **then**

$$\mathbf{x}_0^a = \mathbf{x}_0^k,$$

Break

end if

end for

Evolve the analysis state $\mathbf{x}_0^a$ to the beginning of the next cycle through the nonlinear dynamics (1.1). The forecast state is used to initialize the next assimilation cycle.

end procedure

Chapter 3

Sequential Methods: The Ensemble Kalman Filter

The Kalman filter constitutes a powerful tool for tracking and data assimilation. However, this technique is only well defined for linear dynamic and observation operators. In order to incorporate a nonlinear operator defined in high dimensional spaces, Kalman filtering must be significantly adjusted.

3.1 Extended Kalman filter

With a nonlinear discrete model operator corresponding to Eq.(1.1),

$$\mathbf{x}_{k+1} = \varphi_k(\mathbf{x}_k), \tag{3.1}$$

and a nonlinear observation operator equation (1.3), the first idea has been to rely on a Taylor series expansion of nonlinear models. This constitutes the well-known extended version of the Kalman filter (EKF). This technique works well only for slightly nonlinear models. Its general expression is given in algorithm 8.

3.2 Ensemble Kalman filter

In nonlinear, high dimensional applications, the EKF cannot be implemented due to the high cost associated with the construction of the evolved covariance matrix. Besides, the implementation of the EKF relies on the local linear tangent and adjoint models, which leads to neglect the nonlinear effects. Efficient ensemble techniques have been devised specifically for that purpose following the work initiated by Evensen (1994). They are mainly defined through replacing the forecast mean and covariance matrix by an empirical expression of the ensemble mean and covariance matrix. This low-rank approximation is accompanied by an intensive use of the incomplete SVD representation to implement efficient matrix vector multiplications or to define a pseudo-inverse.

The Ensemble Kalman filter (EnKF) is a two-phase filter: in the forecast phase, the ensemble is propagated by the nonlinear dynamics. A cloud of possible states is generated from a randomization of the dynamics or of its parameters. This ensemble of samples allows computing a low-rank approximation of the error covariance matrix. Two main kinds of methods have been proposed for the second step, the analysis stage. The first one relies on a direct Monte Carlo approach, which introduces measurement noise samples

Algorithm 8 Extended Kalman Filter Algorithm

1: **procedure** ANALYSIS
2: From time instant k, compute the forecast state $\mathbf{x}^f_{k+1}$ with the forward integration of relation (3.1)
3: Compute the forecast error covariance matrix $\mathbf{P}^f_{k+1}$ based on the error propagation equation driven by the tangent linear dynamic model

$$\mathbf{P}^f_{k+1} = \partial_{\mathbf{x}}\varphi_k \mathbf{P}^a_k \partial_{\mathbf{x}}\varphi^*_k + \mathbf{Q}_k. \tag{3.2}$$

4: The analysis state $\mathbf{x}^a_{k+1}$ can be obtained directly from

$$\mathbf{x}^a_{k+1} = \mathbf{x}^f_{k+1} + \mathbf{K}_{k+1}(\mathcal{Y}_{k+1} - \mathbb{H}(\mathbf{x}^f_{k+1})), \tag{3.3}$$

where $\mathbf{K}_{k+1}$ is the Kalman gain matrix. It reads

$$\mathbf{K}_{k+1} = \mathbf{P}^f_{k+1}\partial_{\mathbf{x}}\mathbb{H}^*(\partial_{\mathbf{x}}\mathbb{H}\mathbf{P}^f_{k+1}\partial_{\mathbf{x}}\mathbb{H}^* + \mathbf{R})^{-1}. \tag{3.4}$$

5: The analysis error covariance matrix $\mathbf{P}^a_{k+1}$ is updated by:

$$\mathbf{P}^a_{k+1} = \mathbf{P}^f_{k+1} - \mathbf{K}_{k+1}\partial_{\mathbf{x}}\mathbb{H}\mathbf{P}^f_{k+1}. \tag{3.5}$$

6: Propagate the analysis state $\mathbf{x}^a_{k+1}$ and associated error covariance matrix $\mathbf{P}^a_{k+1}$ to the next instant when observation is available
7: **end procedure**

(Burgers et al., 1998; van Leeuwen and Evensen, 1996; Houtekamer and Mitchell, 1998). The second one, corresponds to the square-root filter technique (Anderson, 2003; Bishop et al., 2001; Ott et al., 2004; Tippett et al., 2003; Whitaker and Hamill, 2002). Those latter schemes avoid sampling issues associated with small-size ensembles by confining the analysis in the space spanned by the forecast ensemble centered perturbations.

3.2.1 Ensemble forecast

The term ensemble in this thesis refers to a set of samples of a system. It usually indicates the state ensemble in the context of data assimilation. Each realization constitutes a member of the ensemble. The more ensemble members are employed, the more accurate are the approximations. The concept of ensemble is widely used in the Monte Carlo method in which the evolution of the model statistics is represented by the propagation of ensemble states at different time levels.

The ensemble can be generated by different initial conditions or parameters driven by system (1.1). This scheme is created by perturbing the initial condition. However, more and more ensemble systems are generated by perturbing the model parameters. Such a procedure seems to be able to generate an ensemble with a more accurate ensemble spread. The parameter perturbation has been used in ensemble forecast (Wei et al., 2013) and in EnKF (Wu et al., 2008).

The ensemble generation works as follows. In the initialization stage, a multiple perturbation vector is added to the background field in order to generate the initial ensemble,

$$\mathbf{x}^{f,(i)}_0 = \mathbf{x}^f_0 + \delta^{(i)}_0, \quad i = 1, \ldots, N. \tag{3.6}$$

Each member of the ensemble is independently integrated through the nonlinear model operator, (3.1),

$$\mathbf{x}_t^{f,(i)} = \varphi_t(\mathbf{x}_0^{f,(i)}), \quad i = 1, \ldots, N. \tag{3.7}$$

Thereafter we can define matrix $\mathbf{X}_t^f = (\mathbf{x}_t^{f,(1)}, \ldots, \mathbf{x}_t^{f,(i)}, \ldots, \mathbf{x}_t^{f,(N)}) \in \mathbb{R}^{n \times N}$ gathering the ensemble state samples, and $\langle \mathbf{X}_t \rangle$ as the ensemble mean. The operator $\langle f(t) \rangle = N^{-1} \sum_1^N f^{(i)}(t)$ takes an empirical ensemble mean of a quantity $f(t)$ through N samples. The ensemble covariance matrix $\mathbf{P}^e$ can be used to approximate the forecast error covariance matrix defined in Eq.(1.34)

$$\mathbf{P}^f \approx \mathbf{P}^e = \frac{1}{N-1}(\mathbf{X}_t^f - \langle \mathbf{X}_t \rangle)(\mathbf{X}_t^f - \langle \mathbf{X}_t \rangle)^T, \tag{3.8}$$

or, alternatively,

$$\mathbf{P}^f \approx \mathbf{P}^e = \frac{1}{N-1}(\mathbf{X}_t'^f)(\mathbf{X}_t'^f)^T, \tag{3.9}$$

where $\mathbf{X}_t'^f = \mathbf{X}_t^f - \langle \mathbf{X}_t \rangle$ denotes the anomaly ensemble matrix.
Remark 1: The initial ensemble field has to be integrated with the dynamic model for some time to ensure the dynamic balance of the ensemble (Evensen, 2003).
Remark 2: The noise term δ_0 added to the background field should represent the same statistic as the background error $\boldsymbol{\eta}$.
Remark 3: We assume here there is no model error. If the model is imperfect, the integration scheme needs to be adjusted to account for the stochastic noise (Evensen, 2003).

3.2.2 Ensemble analysis scheme

The perturbed observation approach

In the pioneering work on the EnKF by Evensen (1994), the analysis scheme includes the error statistics given by the ensemble forecast fields, but does not employ the perturbed observation data. Burgers et al. (1998) point out that the latter must be corrected in order to maintain a consistent error covariance of the analysis ensemble. This idea is realized by introducing an ensemble observation.

$$\mathcal{Y}_t^{(i)} = \mathcal{Y}_t + \epsilon_t^{(i)}, \quad i = 1, \ldots, N, \tag{3.10}$$

then each member of the ensemble must be updated using the same formalism as the Kalman filter update equation.

$$\mathbf{x}_t^{a,(i)} = \mathbf{x}_t^{f,(i)} + \mathbf{K}_t(\mathcal{Y}_t^{(i)} - \mathbb{H}(\mathbf{x}_t^{f,(i)})), \quad i = 1, \ldots, N \tag{3.11}$$

The Kalman gain matrix, $\mathbf{K}$, takes the form:

$$\mathbf{K}_t = \mathbf{P}^f \mathbb{H}^T (\mathbb{H}\mathbf{P}^f \mathbb{H}^T + \mathbf{R})^{-1}, \tag{3.12}$$

Note that the nonlinear observation operator is used in Eq.(3.12). In practice, the same empirical approximation is used to calculate the terms $\mathbf{P}^f \mathbb{H}^T$ and $\mathbb{H}\mathbf{P}^f \mathbb{H}^T$:

$$\mathbf{P}^f \mathbb{H}^T = \frac{1}{N-1}(\mathbf{X}_t^f - \langle \mathbf{X}_t^f \rangle)(\mathbb{H}(\mathbf{X}_t^f) - \langle \mathbb{H}(\mathbf{X}_t^f) \rangle)^T, \tag{3.13}$$

$$\mathbb{H}\mathbf{P}^f \mathbb{H}^T = \frac{1}{N-1}(\mathbb{H}(\mathbf{X}_t^f) - \langle \mathbb{H}(\mathbf{X}_t^f) \rangle)(\mathbb{H}(\mathbf{X}_t^f) - \langle \mathbb{H}(\mathbf{X}_t^f) \rangle)^T, \tag{3.14}$$

The analysis covariance matrix is updated empirically from the ensemble through

$$\mathbf{P}^a \approx \frac{1}{N-1}(\mathbf{X}_t^a - \langle \mathbf{X}_t^a \rangle)(\mathbf{X}_t^a - \langle \mathbf{X}_t^a \rangle)^T, \tag{3.15}$$

instead of a direct transformation from the forecast error covariance matrix $\mathbf{P}^f$. At the end of each analysis step, the analysis ensemble is then integrated through the dynamical model until the next observation. This provides a new forecast ensemble together with a new forecast error covariance matrix. In practice, $\mathbf{P}^f$ and $\mathbf{P}^a$ are never explicitly calculated nor stored as they do not appear explicitly in the analysis scheme.

The direct transformation approach

Tippett et al. (2003) argued that the use of perturbed observations can result in sampling issues with small number ensembles. To overcome this potential drawback, a proper matrix transformation must be realized to directly link the forecast ensemble error characteristic to the analysis ensemble error characteristic. This transformation is made through the introduction of the square-root of the analysis error covariance matrix. As the error covariance matrix is a symmetric positive-definite matrix, it can be expressed as,

$$\mathbf{P}^f = \mathbf{A}'^f \mathbf{A}'^{f^T}, \tag{3.16}$$

$$\mathbf{P}^a = \mathbf{A}'^a \mathbf{A}'^{a^T}, \tag{3.17}$$

where the matrix $\mathbf{A}'^f$ denotes the square root matrix of $\mathbf{P}^f$. In the case of an ensemble approximation of the error covariance, $\mathbf{A}'^f$ is given by,

$$\mathbf{A}'^f = \frac{1}{\sqrt{N-1}}\mathbf{X}_t'^f. \tag{3.18}$$

Several methods following this path have been proposed (Anderson, 2001; Whitaker and Hamill, 2002; Bishop et al., 2001). Here we detail the example the Ensemble Transform Kalman Filter (ETKF), proposed initially by Bishop et al. (2001), which consists in assuming that $\mathbf{A}'^a$ equals to the product of $\mathbf{A}'^f$ right-multiplied by a transformation matrix $\mathbf{T}$:

$$\mathbf{A}'^a = \mathbf{A}'^f \mathbf{T}, \tag{3.19}$$

Let us note that if $\mathbf{T}$ is left-multiplied by $\mathbf{A}'^f$, we obtain a different method referred to as the EAKF (Ensemble Adjustment Kalman Filter) proposed in Anderson (2001).

Now let us start from the updating equation of the analysis covariance matrix in the Kalman filter:

$$\mathbf{P}^a = \mathbf{P}^f - \mathbf{P}^f\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{P}^f. \tag{1.39}$$

Considering the square root matrix, we have,

$$\mathbf{A}'^a = \mathbf{A}'^f[\mathbb{I} - \mathbf{A}'^{f^T}\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{A}'^f]^{\frac{1}{2}}. \tag{3.20}$$

This equation can be transformed through the Sherman-Morrison-Woodbury identity as,

$$\mathbf{A}'^a = \mathbf{A}'^f[\mathbb{I} + \mathbf{A}'^{f^T}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{A}'^f]^{-\frac{1}{2}}. \tag{3.21}$$

For sake of simplicity, we will assume a linear operator $\mathbf{H}$ for the observation operator.

In ETKF, the *a posterior*, estimate lives in the subspace generated by N samples: $\langle \mathbf{X}_t^a \rangle = \langle \mathbf{X}_t^f \rangle + \mathbf{X}_t'^f w$, the vector, w corresponds to the coordinates in the subspace spanned by the sampled ensemble members. The weight, w, is fixed in order to respect the Kalman equations.

Finally, we have:

$$\mathbf{T}_t = [\mathbb{I}_N + (\mathbf{H}\mathbf{A}_t'^f)^T \mathbf{R}^{-1} \mathbf{H}\mathbf{A}_t'^f]^{-\frac{1}{2}}, \tag{3.22}$$

$$\mathbf{A}_t'^a = \mathbf{A}_t'^f \mathbf{T}_t, \tag{3.23}$$

$$\langle \mathbf{X}_t^a \rangle = \langle \mathbf{X}_t^f \rangle + \mathbf{A}_t'^f (\mathbf{H}\mathbf{A}_t'^f)^T (\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1} (\mathcal{Y}_t - \mathbb{H}(\langle \mathbf{X}_t^f \rangle)), \tag{3.24}$$

$$\mathbf{X}_t^a = \langle \mathbf{X}_t^a \rangle + \sqrt{N-1}\mathbf{A}_t'^a. \tag{3.25}$$

Note that from an implementation point of view, a singular value decomposition of matrix $[\mathbb{I}_N + (\mathbf{H}\mathbf{A}_t'^f)^T \mathbf{R}^{-1} \mathbf{H}\mathbf{A}_t'^f]$ is usually performed. We have thus:

$$\mathbf{T} = \mathbf{C}(\Gamma + \mathbb{I}_N)^{-1/2}, \tag{3.26}$$

where the matrix $\mathbf{C}$ and the diagonal matrix Γ contain the orthonormal eigenvalues and the eigenvector of $(\mathbf{H}\mathbf{A}_t'^f)^T \mathbf{R}^{-1} \mathbf{H}\mathbf{A}_t'^f$, respectively, and $\mathbb{I}_N$ is the $N \times N$ identity matrix.

The equation (3.24) can also be approximated in the same way since the residual covariance matrix $(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R}) \in \mathbb{R}^{m \times m}$ remains hard to invert.

$$\langle \mathbf{X}_t^a \rangle = \langle \mathbf{X}_t^f \rangle + \mathbf{A}_t'^f (\mathbb{I}_N - \mathbf{C}(\Gamma + \mathbb{I}_N)^{-1}\mathbf{C}^T)(\mathbf{H}\mathbf{A}_t'^f)^{-1}(\mathcal{Y}_t - \mathbb{H}(\bar{\mathbf{X}}_t^f)). \tag{3.27}$$

To obtain this expression, we define an expression as $\mathbb{I}_N - [\mathbb{I}_N + \mathbf{A}'^{f^T}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{A}'^f]^{-1}$ and apply the Sherman-Morrison-Woodbury identity to this expression:

$$\mathbb{I}_N - [\mathbb{I}_N + \mathbf{A}'^{f^T}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{A}'^f]^{-1} \iff \mathbf{A}'^{f^T}\mathbf{H}^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{A}'^f. \tag{3.28}$$

The above equation is left-multiplied by $\mathbf{A}_t'^f$ and right-multiplied by $(\mathbf{H}\mathbf{A}_t'^f)^{-1}$ in order to respect the form in (3.24),

$$\mathbf{A}_t'^f \Big(\mathbb{I}_N - [\mathbb{I}_N + \mathbf{A}'^{f^T}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{A}'^f]^{-1}\Big)(\mathbf{H}\mathbf{A}_t'^f)^{-1} \iff \mathbf{A}_t'^f(\mathbf{H}\mathbf{A}_t'^f)^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}. \tag{3.29}$$

The inverse of the term $[\mathbb{I}_N + \mathbf{A}'^{f^T}\mathbf{H}^T\mathbf{R}^{-1}\mathbf{H}\mathbf{A}'^f]$ is already solved, thus we have:

$$\mathbf{A}_t'^f(\mathbb{I}_N - \mathbf{C}(\Gamma + \mathbb{I}_N)^{-1}\mathbf{C}^T)(\mathbf{H}\mathbf{A}_t'^f)^{-1} \approx \mathbf{A}_t'^f(\mathbf{H}\mathbf{A}_t'^f)^T(\mathbf{H}\mathbf{P}^f\mathbf{H}^T + \mathbf{R})^{-1}, \tag{3.30}$$

and the term $(\mathbf{H}\mathbf{A}_t'^f)^{-1}$ can be evaluated as a pseudo-inverse through SVD as well.

In addition, the square root matrix $\mathbf{A}'$ is not unique and can be defined up to an orthogonal matrix $\mathbf{U}$. Judicious choices allow preserving in the analysis matrix the zero mean of the forecast anomaly matrix (*i.e.* $\mathbf{X}_t'^a\mathbf{U} = 0 \Rightarrow \mathbf{X}_t'^f\mathbf{T}_t\mathbf{U} = 0$) (Wang et al., 2004; Sakov and Oke, 2007).

Let us note that other ensemble filters, defined from the particle filtering concept (Gordon et al., 1993), use also these recursive equations to build efficient filtering procedures (Beyou et al., 2013a,b; Papadakis et al., 2010; van Leeuwen, 2010).

3.2.3 Relationship with 4DVar

The corresponding function (2.45) can be rewritten in a simpler form as:

$$J(\delta\mathbf{x}_0) = \frac{1}{2}\{\delta\mathbf{x}_0, \mathcal{H}\delta\mathbf{x}_0\} - \int_{t_0}^{t_f} \{\delta\mathbf{x}_0, b\} + C, \tag{3.31}$$

where the Hessian, the linear term are defined in (2.56) and (2.57) respectively. The constant term C gathers all the rest terms,

$$C = \int_{t_0}^{t_f} \mathbf{D}^T(t)\partial_{\mathbf{x}}\varphi_t^*\partial_{\mathbf{x}}\mathbb{H}^*\mathbf{R}^{-1}\partial_{\mathbf{x}}\mathbb{H}\partial_{\mathbf{x}}\varphi_t\mathbf{D}(t)dt. \tag{3.32}$$

We highlight that for a linear dynamics and linear observation operator, fixing the background covariance matrix to an empirical matrix $\mathbf{X}_t\mathbf{X}_t^T/(N-1)$ with $\mathbf{X}_t = \phi_t(\mathbf{X}_0)$, the tangent linear model is the linear dynamical model itself: $\partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0) = \Phi_t$. Consequently we recover the Kalman filter equivalence at the end of the assimilation window. Defining furthermore the Hessian (or the inverse of the error covariance) from the square root of Eq.(2.56) we get exactly the ETKF solution. This equivalence holds only for a fixed single time assimilation window (the 3DVar technique) or at the end of the assimilation window for a linear dynamics. The ETKF technique may be useful to provide an approximate cost function's Hessian.

3.3 Summary

In this chapter we briefly reviewed the ensemble Kalman filter methods. The recursive filter procedure consists of two steps: the ensemble forecast and the ensemble analysis. The ensemble analysis scheme contains two distinctive variants: the perturbed observation approach and the direct transformation approach. Both variants have been briefly presented.

Part II

Hybrid Methods

Chapter 4

Hybrid Methods Review

In the previous part we firstly introduced the data assimilation problem from a optimal estimation point of view, then the variational method has been presented from the perspective of optimal control of partial differentiation equations. At last, we focused on the ensemble Kalman filter that relies on a Monte Carlo implementation of Kalman filter.

Recently, several schemes aiming at coupling the advantages of ensemble methods and variational assimilation strategies have been proposed. Generally speaking, they differ mainly on the framework on which they are based. One group of hybrid methods retains the formalism of an iterative gradient descent procedures of the cost function derived in the context of variational methods. In this group, the scheme is usually built through the incorporation within the variational cost function of an ensemble-based background covariance, thus it has more rigorous basis as they rely on exact descent procedures. In the continuity of the ideas proposed for fixed time data assimilation cost function (Hessian-incr3DVar) in Hamill and Snyder (2000) and extended later to temporal variational data assimilation in Lorenc (2003), several authors have proposed methods that express explicitly either the background error covariance as a linear combination of the ensemble covariance and the static covariance or the solution as a linear combination of the square root of the ensemble-based covariance (Buehner, 2005; Zhang et al., 2009; Buehner et al., 2010a,b; Clayton et al., 2012; Fairbairn et al., 2013; Buehner et al., 2013; Desroziers et al., 2014). The optimization is often coupled with an adjoint scheme (Zhang et al., 2009; Buehner et al., 2010a,b; Krysta et al., 2011; Clayton et al., 2012).

The other group of hybrid methods retains the formalism of ensemble Kalman filter, and in principle aims at assimilating asynchronous data Hunt et al. (2004); Zupanski (2005); Fertig et al. (2007); Hunt et al. (2007); Sakov et al. (2010). Note that the ability to assimilate asynchronous data is indeed a topic covered by the Kalman smoother initially proposed in van Leeuwen and Evensen (1996), then revisited as ensemble Kalman smoother (EnKS) in Evensen and van Leeuwen (2000). (Nevertheless, the EnKS method is, by definition, differs from the EnKF, and therefore outside of the scope of our interest in terms of 'hybridization'.) Another notable attribute of this group is that when facing high dimension problems, the explicit analysis or update step in terms of the Kalman filter tends to be replaced by an iterative gradient descent procedures of a cost function. The minimization procedure is similar to the variational methods, but the form of the cost function is arbitrary. A popular choice consists in keeping the form of variational methods (Zupanski, 2005; Solonen et al., 2012; Sakov et al., 2012; Bocquet and Sakov, 2013a). It is important to mention at this point that although the iterative ensemble Kalman filter (IEnKS) method, proposed in Bocquet and Sakov (2013a), is not a hybrid method by its

definition, it actually addresses the same problem as standard 4DVar. So this method will be introduced as well.

We will present above approaches in the following sections.

4.1 Incorporate the ensemble-based error covariance into variational system

In the inception of hybrid method, the idea has been applied to static variational assimilation context referred usually to as 3DVar.

4.1.1 EnKF-3DVar

In this way, Hamill and Snyder (2000) proposed a hybrid method combining a 3DVar cost function and the EnKF update process. Their technique consists basically to implement in parallel several 3DVar data assimilation processes. In each process, the corresponding ensemble forecast is used as the background condition and the background error covariance is fixed as a weighted combination of the ensemble based covariance $\mathbf{P}^f$ with a static covariance matrix $\mathbf{B}^c$ formulated by 'NMC' method.

$$\mathbf{B} = \alpha_1 \mathbf{P}^e + \alpha_2 \mathbf{B}^c. \tag{4.1}$$

The analysis error covariance matrix is calculated from the analysis ensemble fields.

4.1.2 EnKF-3DVar with CVT

Lorenc (2003) rephrased the idea of using ensemble covariance in variational methods in defining the unknown increment as a linear combination of the sum of weighted ensemble perturbations plus the traditional control variable transformation.

$$\delta\mathbf{x} = \beta_1 \mathbf{P}^{e\frac{1}{2}} \delta\mathbf{z} + \beta_2 \mathbf{B}^{c\frac{1}{2}} \delta\mathbf{v}. \tag{4.2}$$

He also argues the possibility of extending this technique to the standard 4D-Var method by considering a flow-dependent covariance modeled by EnKF at the beginning of the assimilation window. This covariance matrix then implicitly evolves within the assimilation window through the tangent and adjoint models.

Localization: inverse weighting matrix

Lorenc especially highlights the crucial importance of a variance localization procedure for the covariance produced by a limited number ensemble. He proposed a localization procedure by using a spatial correlation function $\mathcal{C}$ as the weighting matrix exerted to the background part of the cost function J_b:

$$J_b = \frac{1}{2} \sum_{i=1}^{N} \|\delta\boldsymbol{\alpha}_i\|^2_{\mathcal{C}^{-1}} + \frac{1}{2} \|\delta\mathbf{v}\|^2. \tag{4.3}$$

It can be shown that this formulation implies the following control variable transformation,

$$\delta\mathbf{x} = \beta_1 \sum_{i=1}^{N} (\mathbf{A}'_e)_i \odot \delta\boldsymbol{\alpha}_i + \beta_2 \mathbf{B}^{c\frac{1}{2}} \delta\mathbf{v}. \tag{4.4}$$

Note the Schur product between the ensemble perturbation and the associated control vector.

Localization: transform ensemble perturbation matrix

Buehner (2005) examined the different strategies of constituting the background error covariance (perturbed 3D-Var, EnKF and the hybrid), and argued that these ensemble-based covariance constitute good alternatives to arbitrary covariance used in the context of operational data assimilation. He also proposed a localization scheme which differs from the one previously proposed. In this scheme, the weighting correlation function is not introduced into the cost function but operates a localization directly on the perturbation matrix $\mathbf{A}'_e$. This new perturbation matrix $\mathbf{P}'_e$ is defined as,

$$\mathbf{P}'_e = (diag(\mathbf{A}'^{(1)}_e)\mathcal{C}^{\frac{1}{2}}, \ldots, diag(\mathbf{A}'^{(N)}_e)\mathcal{C}^{\frac{1}{2}}) \approx \mathbf{P}^{e\,\frac{1}{2}}, \tag{4.5}$$

The control variable transformation (4.2) is then applied on this localized ensemble perturbation matrix.

In Wang et al. (2007), both the hybrid schemes proposed by Hamill and Snyder (2000) and Lorenc (2003); Buehner (2005) are proved to be equivalent. The two localization schemes proposed by Lorenc (2003) and Buehner (2005) are shown to be indeed equivalent as well.

4.1.3 ETKF-3DVar

A hybrid method relying on the Ensemble Transform Kalman Filter (ETKF) has been also proposed in Wang et al. (2008a) to generate the analysis ensemble perturbation. Compared to the previous hybrid EnKF-3DVar, the observation perturbation is not needed for ETKF-3DVar. The hybrid cost function takes the form of Eq.(4.3). The ensemble perturbations are updated following the ETKF deterministic update (refer to section 3.2.2 for more details),

$$\mathbf{A}'^{a}_e = \mathbf{A}'^{f}_e \mathbf{C}(\Gamma + \mathbb{I}_N)^{-1/2}\mathbf{C}^T, \tag{4.6}$$

This scheme has been tested with simulated observation system and real observations experiments in Wang et al. (2008a,b). The authors found that the hybrid method could recover unobserved moisture fields from other state variables due to the cross-variable covariance derived directly from ensemble. Note that the ETKF scheme used to update the ensemble perturbation does not include any localization procedures.

4.1.4 En4DVar

Previous methods could be extended to 4DVar with little effort if the evolution of error covariance is implemented through the use of a standard adjoint minimization technique. We will call this extended method En4DVar.

An example of En4DVar denoted by Buehner et al. (2010a,b) as 4DVar-Benkf is implemented in a quasi-operational model and compared with other variational methods. In 4DVar-Benkf, the error covariance matrix employed at the beginning of the 4DVar system is inherited from an EnKF system's ensemble. Note that in this scheme, the EnKF system run is independently realized prior to the 4DVar system run. It is suggested by the authors that the integration of the 3D ensemble-based error covariance led to better analysis especially in those regions where the observation is only indirectly linked to the state variable or where the observation are scarcer.

An operational implementation of En4DVar is studied in Clayton et al. (2012). The authors used the ensemble-based covariance from an ensemble prediction system (EPS)

based on ETKF. The results are encouraging. They also propose an interesting way to preserve the variable balance of ensemble errors due to localization. The procedure can be expressed as the following equation in accordance with the notations used in section 2.5.

$$\delta \mathbf{x} = K\{\beta_1 \sum_{i=1}^{N} [K^{-1}(\mathbf{A}'_e)_i] \odot \delta \mathbf{z} + \beta_2 D^{\frac{1}{2}} F \delta \mathbf{v}\}, \tag{4.7}$$

where the operator K, D, F are the balance operator matrix, the diagonal variance matrix and the square root of a spatial correlation operator respectively.

A similar coupling strategy of EnKF and 4DVar is proposed in Zhang et al. (2009) to benefits from the advantages of each method. Particularly, the adjoint operator is considered as beneficial in the assimilation and therefore it is retained. In addition to the use of background error covariance matrix derived from EnKF, the analysis ensemble mean of EnKF is replaced by the 4DVar analysis. This so-called E4DVar method has been applied to a limited-area weather prediction model in Zhang and Zhang (2012), and the authors found that the coupled scheme yield better results than the 4DVar or the EnKF systems. Note that in E4DVar, the 4DVar system not only absorbs the flow-dependent error structures from the EnKF system, but also provides its analysis as the analysis ensemble mean used in EnKF. But in 4DVar-Benkf, only a one way information exchange is conducted.

This method is promising, however, as the computation of the tangent-linear and adjoint model operators in 4DVar can be quite involved, some methods have been designed to avoid the conduction of such procedures.

4.1.5 4DEnVar

Liu et al. (2008) proposed a hybrid method in which the evolution of error covariance matrix is computed explicitly form the 4D ensemble driven by the non-linear model. The idea of this method is to expand the increment solely on the basis of ensemble perturbation, which comes setting $\beta_1 = 1, \beta_2 = 0$ in Eq.(4.2),

$$\delta \mathbf{x} = \mathbf{P}^{e\,\frac{1}{2}} \delta \mathbf{z}, \tag{4.8}$$

By doing so, the gradient of the cost function can be computed explicitly from the error covariance derived from a 4D ensemble, therefore avoiding the usage of the tangent linear or adjoint forecast models. The advantage of this method lies in its simplicity and computational efficiency, however, the low dimension subspace spanned by the ensemble members may suffer from the same degeneracy problems as EnKF in terms of errors sampling.

A corresponding covariance localization form of 4DEnVar is proposed in Liu et al. (2009) where the authors adopt the idea of Buehner (2005) to the temporal evolution of the ensemble perturbation matrix. Another possible covariance localization approach based on Lorenc (2003) is implemented in Desroziers et al. (2014) along with other localization schemes. Desroziers et al. (2014) also highlights the importance of the dimension of the control vector which is directly related to the cost of the minimization algorithm.

A similar approach is also tested in Buehner et al. (2010a,b), followed by Buehner et al. (2013). A key difference should be noted: The 4DEnVar implemented by Buehner used the ensemble fields generated by an EnKF system, whereas in Liu et al. (2009), no EnKF is explicitly required, instead, the error covariance is generated by an ensemble of variational systems. Each member of the ensemble is subject to the optimization process

with perturbed observations. This could be interpreted as a Monte-Carlo formulation of a variational system. This important attribute is pointed out by Fairbairn et al. (2013) where the authors proposed the EDA method as an ensemble of 4DEnVars. Two ways of updating the error covariance have been introduced in Fairbairn et al. (2013). The first one, called EDA-S, updates the ensemble perturbation using perturbed observation. It is indeed identical to the method of Liu et al. (2009). The other method, named EDA-D, used the deterministic Ensemble Kalman filter proposed by Sakov and Oke (2008) to avoid the use of perturbed observations.

Despite in 4DEnVar, an ensemble of states needs to be processed under the variational system, this method is still potentially computationally advantageous due to the extremely low cost of the minimization procedure ($\mathcal{O}(10^2)$ against $\mathcal{O}(10^6)$). Covariance localization acts as a double-edged sword in terms of quality improvement and efficiency increase of 4DEnVar. Increasing the distance beyond which two points decorrelate (the decorrelation length), enlarges reduced subspaces spanned by modified ensemble perturbation; but it also leads to an enormous increasing number of control vector. More details of this approach will be discussed in chapter 5.

4.1.6 Explicit 4DVar

A variant method called ensemble-based explicit 4DVar is proposed by several articles Qiu and Chou (2005); Tian et al. (2008). Unlike previous ensemble-based 4DVar methods where the ensemble information is injected into the error covariance terms, the authors introduced the ensemble information into the analysis increment through either SVD or POD technique. In this method, N members of 4D snapshots of state variables are constructed by perturbing randomly the initial condition. The 4D analysis state $\mathbf{x}_t^a$ is then expressed as a linear combination of the base vector extracted from the N members of 4D snapshots. By inserting the new expression into the cost function, it is possible to exhibit explicit form of the gradient with respect to the weighting vector. As a consequence, the evaluation of the tangent linear operator or its adjoint are not required. The SVD and POD techniques differ slightly in terms of the base vectors extracted from the 4D snapshots.

4.2 Assimilating asynchronous data with EnKF

The standard EnKF is a method allowing assimilating sequentially the observation whenever they are available (Evensen, 2003). Thus the observation only affects the current and future analysis state. The variational method, on the other hand, finds the best fit of the model states to all the observations, both past and future. One motivation of this modified EnKF method consists in remaining within the EnKF system while trying to assimilate the data at different times spontaneously as the 4DVar method.

4.2.1 4DEnKF

The four-dimensional EnKF (Hunt et al., 2004) makes the EnKF method capable of assimilating asynchronous data. The essence of this method is to find an observation operator which could link the state vector at this moment to future or past observations, once this goal is achieved, the extension to four dimension of EnKF is straightforward. In 4DEnKF, if we are at time t_k and we want to assimilate the observation $\mathcal{Y}_0$ at time t_0, the asyn-

chronous observation operator reads (Eq.(8) in Hunt et al. (2004)),

$$\mathbf{H}_0\mathbf{x}_0 = \mathbf{h}_0(\mathbf{X}_0(\mathbf{X}_k^T\mathbf{X}_k)^{-1}\mathbf{X}_k^T\mathbf{x}_k), \tag{4.9}$$

where $\mathbf{X}_k$ is the ensemble fields at time t_k. The operator $\mathbf{h}_0$ links directly the set of states to the observation $\mathcal{Y}_0$. If we want to assimilate multiple observations distributed along assimilation window (t_0, t_f), the observation operator is defined as (Eq.(9) in Hunt et al. (2004)),

$$\mathbf{H} = \begin{pmatrix} \mathbf{h}_0\mathbf{X}_0 \\ \mathbf{h}_1\mathbf{X}_1 \\ \vdots \\ \mathbf{h}_f\mathbf{X}_f \end{pmatrix} (\mathbf{X}_f^T\mathbf{X}_f)^{-1}\mathbf{X}_f^T, \tag{4.10}$$

in order to relate observations $\mathcal{Y}_k$ to the whole state variables' trajectory. This operator can be applied to the analysis scheme discussed in section 3.2.2. Either the perturbed observation approach or the direct transform approach are applicable.

4.2.2 4D-LETKF

The 4D-LETKF, the successor of 4DEnKF, introduced by Hunt et al. (2007); Fertig et al. (2007) (4D local ensemble transform Kalman filter) adopts the same idea as 4DEnKF in the context of a variational optimization system. The idea takes its roots on the equivalence between 4DVar and Kalman filter (KF) (Li and Navon, 2001) and is elaborated by Hunt et al. (2007). The authors rely on the fact that the optimal state estimation in KF is indeed identical to minimize a cost function of the form,

$$J_k = \|\mathbf{x}_k - \mathbf{x}_k^b\|^2_{\mathbf{P}_k^f} + \|\mathbb{H}(\mathbf{x}_k) - \mathcal{Y}_k\|^2_{\mathbf{R}_k}, \tag{4.11}$$

where $\mathbf{x}_k$ is the optimal estimation provided by the KF at time step t_k. A simple way to deduce this cost function is to develop Eq.(1.19) under the assumption of absence of correlation between the background errors and the observation errors. Note that although Eq.(4.11) has an identical form as a 3DVar cost function by replacing $\mathbf{P}_k^f$ with $\mathbf{B}$, their implications are completely different. The cost function form (4.11) ensues from Gaussian hypothesis on the distribution of the background and the observation errors; the cost function defined in variational system, on the other hand, do not rely specifically on such hypothesis and can be more freely defined. The minimization of the cost function of the form Eq.(4.11) corresponds to an estimator of the *a posteriori* conditional mean.

The 4D-LETKF is formulated on the basis of ensemble similar to EnKF where the forecast error covariance matrix $\mathbf{P}_k^f$ is approximated by the ensemble covariance as in Eq.(3.8). If we assume that the analysis increment belongs to the space spanned by $\mathbf{X}'^f$, the analysis is necessarily of the form $\mathbf{x}_k = \mathbf{x}_k^b + \mathbf{X}'^f\mathbf{w}$. By inserting this form into Eq.(4.11) and dropping subscript k, we deduce the following modified cost function:

$$J(\mathbf{w}) = (N-1)\|\mathbf{w}\|^2 + \|\mathbb{H}(\mathbf{x}^b + \mathbf{X}'^f\mathbf{w}) - \mathcal{Y}\|^2_{\mathbf{R}}, \tag{4.12}$$

The minimization and cycling of such a system rely on ETKF principles described through Eqs. (3.22) to (3.25). The expansion to assimilate 4D observations of such transformation in the analysis step hinges on the idea of considering the analysis trajectories as linear combination of the ensemble background trajectories. This idea is used in the explicit 4DVar framework as well. Considering a state variable trajectory from t_0 to t_f, with a set

of observations distributed on the same interval and denoting $\mathcal{X}$ the 4D ensemble fields concatenation,

$$\mathcal{X}^f = (\mathbf{X}_0, \ldots, \mathbf{X}_{t_f}), \tag{4.13}$$

the analysis state can be expressed as,

$$(\mathbf{x}_0^a, \ldots, \mathbf{x}_{t_f}^a) = (\mathbf{x}_0^b, \ldots, \mathbf{x}_{t_f}^b) + \mathcal{X}'^f \mathbf{w}, \tag{4.14}$$

where the $\mathcal{X}'^f = \mathcal{X}^f - \langle \mathcal{X}^f \rangle$ is the 4D trajectories' anomaly ensemble matrix. By such a transformation, any asynchronous data can be assimilated and the gradient of cost function Eq.(4.12) with respect to weighting vector $\mathbf{w}$ can be easily calculated. From this point of view, the cost functions involved in 4D-LETKF and in the aforementioned 4DEnVar are indeed equivalent. The algorithm has been further improved by considering localization (Hunt et al., 2007). The covariance localization is done following the local ensemble approach introduce by Ott et al. (2004). We will present this formalism in section 5. The only thing that we need to mention now is that with a local support, the explicit calculation of the analysis state and the error covariance through Kalman filter equations is actually feasible. Those correspond exactly to the analysis steps done in Hunt et al. (2007).

The 4D-LETKF has been studied in Harlim and Hunt (2007) with a primitive equation dynamics model. It showed good consistency when dealing with asynchronous observations. Yang et al. (2009) compared the 4D-LETKF with 3DVar, 4DVar and an ensemble-based 3DVar schemes and found that the 4D-LETKF could be a competitive choice against 4DVar especially with shorter assimilation windows.

4.2.3 AEnKF

The asynchronous EnKF (AEnKF) proposed by Sakov et al. (2010) aims at finding the explicit form of temporal evolution of the increment $\delta\mathbf{x}$ defined by the difference of analysis and forecast ensemble mean,

$$\delta\mathbf{x} = \langle \mathbf{X}^a \rangle - \langle \mathbf{X}^f \rangle = \mathbf{A}'^f \mathbf{G}\mathbf{s},$$

and the increment of ensemble perturbation defined by,

$$\delta\mathbf{A}' = \mathbf{A}'^a - \mathbf{A}'^f = \mathbf{A}'^f \mathbf{T}'.$$

The form of $\mathbf{G}, \mathbf{s}$ and $\mathbf{T}'$ can be directly inferred from the analysis forms of ETKF Eqs.(3.22) to (3.24); a less obvious form can be inferred from the analysis equations of the EnKF with the perturbed observations. See Sakov et al. (2010) for more details. In both cases, following the same scenario as 4DEnKF, with the observation available at t_0, the increments at t_1 can be computed by the linear tangent operator Φ,

$$\delta\mathbf{x}_1 = \Phi_0 \delta\mathbf{x}_0 = \Phi_0 \mathbf{A}_0'^f \mathbf{G}_0 \mathbf{s}_0, \tag{4.15}$$

$$\delta\mathbf{A}_1' = \Phi_0 \delta\mathbf{A}_0' = \Phi_0 \mathbf{A}_0'^f \mathbf{T}_\mathbf{0}', \tag{4.16}$$

These equations show the propagation of the information from the observation time to the analysis time. The idea can be extended to multiple observations very easily. It is argued in Sakov et al. (2010) that the AEnKF and the 4D-LETKF are indeed equivalent.

4.3 Incorporate an iterative procedures into EnKF system

The explicit form of the update equations in EnKF (or EKF) are difficult to calculate when dealing with high dimensional problem. On the other hand, in 4DVar, extensive studies have been done on the methods to minimize efficiently the large dimension associated to the quadratic cost function. Besides, if strong non-linear effects are associated with the dynamical model or the observation operator, it is in general beneficial to resort to an incremental 4DVar framework (section 2.3.3) and its unavoidable outer loop associated to successive linearization.

4.3.1 MLEF

One method that can be classified under this category is the maximum likelihood ensemble data assimilation (MLEF) proposed by Zupanski (2005). The forecast step is similar to EnKF, but the analysis step, unlike EnKF, minimizes a quadratic form cost function like Eq.(4.11) and adopts an iterative minimization algorithm (conjugate-gradient) to solve the unconstrained minimization problem. This method essentially relies on the preconditioning technique by CVT that is similar to Eq.(4.8) in the 4DEnVar method, and replaces the ensemble perturbation matrix $\mathbf{P}^e$ by the square root of the cost function inverse Hessian of the cost function. Since the inverse Hessian belongs to the subspace spanned by the ensemble members, it remains easy to evaluate. The author highlights also an interesting method for evaluating the analysis error covariance matrix: when a quasi-Newton minimization procedure is applied to solve the cost function, the inverse Hessian approximation obtained at the minimum provides a good approximation of the analysis ensemble perturbation matrix. Note that the same relationship also exists in variational system but the calculation of the Hessian is hindered by its dimension. The exact equivalence between the two matrix (analysis error covariance matrix and the inverse Hessian) only holds in a linear case.

4.3.2 IEnKF&IEnKS

Another approach introduced by Sakov et al. (2012) combines the idea of MLEF with further propagation of ensemble fields. The iterative update of analysis state can be explicitly described by a Newton approach in the case of minimizing $\langle \mathbf{X}_0 \rangle$ for future observations $\mathcal{Y}_t$,

$$\langle \mathbf{X}_0^{k+1} \rangle = \langle \mathbf{X}_0^k \rangle + \mathbf{K}_t^k (\mathcal{Y}_t - \mathbb{H}(\langle \mathbf{X}_t^k \rangle)) + \mathbf{P}_0^{f,k} (\mathbf{P}_0^{f,0})^{-1} (\langle \mathbf{X}_0^0 \rangle - \langle \mathbf{X}_0^k \rangle), \tag{4.17}$$

It is clear that in order to evaluate the analysis $\langle \mathbf{X}_0^{k+1} \rangle$ at the $(k+1)$th iteration, the $\mathbf{P}_0^{f,k}$ and $\mathbf{K}_t^k$ must be derived from the updated ensemble fields generated around the analysis state $\langle \mathbf{X}_0^k \rangle$ with the updated ensemble perturbation matrix $\mathbf{A}_0'^k$ at the kth iteration. But in MLEF, only one iteration is realized. In a later modification made by Bocquet and Sakov (2012), the same assumption that the analysis increment lives in the ensemble space is made as in the 4D-LETKF method. Besides, an inflation free scheme is employed.

Its extension, named IEnKS, is proposed in Bocquet and Sakov (2013a). This method relies on an Gauss-Newton iterative minimization of a cost function built on ensemble space. Such a cost function takes a similar form as Eq.(4.12):

$$J(\mathbf{w}) = (N-1)\|\mathbf{w}\|^2 + \|\mathbb{H}_t \mathbb{M}_{t \leftarrow 0} (\mathbf{x}_0^{(0)} + \mathbf{X}'^f \mathbf{w}) - \mathcal{Y}\|_{\mathbf{R}_t}^2, \tag{4.18}$$

It is claimed by the authors that the IEnKS can outperform the EnKF and the EnKS methods regarding both to the smoothing performance and the filtering performance.

4.3.3 VEnKF

A slightly different approach, relied on the low-storage approximation of the large dimension matrices associated with the KF update equations by quasi-Newton minimization methods, is proposed by Solonen et al. (2012). This method shares the same cost function Eq.(4.11) related to Kalman filter as MLEF and IEnKF. Instead of solving it iteratively or by CVT technique, the VEnKF consists in finding the minimum of an auxiliary problem using LBFGS algorithm,

$$\arg\min_{u} u^T(\mathbf{A}'\mathbf{A}'^T + \mathbf{Q})u, \tag{4.19}$$

The inverse forecast error covariance matrix $\mathbf{P}^{f\,-1}$ can be inferred as a byproduct of LBFGS procedure applied to such problem. Once $\mathbf{P}^{f\,-1}$ is known, the minimization of problem Eq.(4.11) is straightforward. The LBFGS algorithm is applied as well because the inverse Hessian matrix formulated at the minimum constitutes a good approximation of the inverse analysis error covariance matrix $\mathbf{P}^{a\,-1}$. A resampling procedure is required to generate new ensemble subject to Gaussian distribution.

4.4 Summary

In the next chapter we will describe an ensemble-based variational method closely related to the strategy proposed by Buehner (2005) and Liu et al. (2008). It will be referred to as 4DEnVar in the following. This technique introduces in its objective function an empirical ensemble-based background-error covariance which avoids the use of tangent linear and adjoint model. The associated optimization is conducted as a gradient descent procedure and does not rely on iterative ensemble filtering updates exploiting linear equivalences between Kalman smoothers and the *a posteriori* energy minimization. Although the standard Kalman filter or the extended Kalman filter employ explicit formulation to calculate the *a posteriori* error covariance matrix, in practice, as we discussed it in section 3.2, the evaluation through Eqs.(1.39) or (3.5) is not possible. An efficient Monte Carlo method is used in ensemble Kalman filter to approximate the *a posteriori* error covariance matrix. In the next section, we will show how the forecast error covariance $\mathbf{P}^f$ and the analysis error covariance matrix $\mathbf{P}^a$ can be formulated when integrated into a preconditioned incremental 4DVar DA algorithm 7.

Beyond this practical implementation aspect, it is nevertheless difficult to state precisely the limitations of such a technique and to assess its comparative performances with respect to the standard 4DVar implementation. As a mater of fact, instead of considering the whole state space, ensemble 4D-Var methods provide, on the one hand, solutions defined on a subspace spanned by the ensemble members, and, on the other hand, they introduce a flow dependent time varying covariance instead of a static full-rank background covariance. As the background error covariance matrix plays a key role in the variational process, our study particularly focuses on the generation of the analysis ensemble state with localization techniques. An experimental evaluation between an incremental 4DVar and an ensemble based 4DVar will be carried out in this study. This evaluation is performed using a Shallow Water model.

Chapter 5

Ensemble-based 4DVar

5.1 Ensemble-based 4DVar scheme

In this section we will describe in detail an enhanced ensemble-based 4D variational method dedicated to image data assimilation. This hybrid method is designed in such a way that it combines the advantage of two distinctive approaches while eliminating their weakness in each side. Comparative summary of the characteristic between 4DVar and EnKF (Lorenc, 2003; Kalnay et al., 2007; Buehner et al., 2010a) is shown in table 5.1. For example, the flexibility of the covariance specification without restrict to particular statistical model is considered as an advantage of 4DVar against EnKF that generally needs the prior pdf of error covariance; being capable of updating the forecast(background) error covariance in EnKF combined with the explicit flow dependent error covariance structures constitute clear advantages; although the tangent linear and adjoint models provide accurate gradient values for the minimization process, the constitution of such adjoint model can often be tedious and cumbersome in practice; in terms of analysis methods, simultaneously assimilating all observations within the assimilation window obviously outperforms its sequential counterpart. By doing so, not only we obtain the optimal initial condition, but also the system trajectory that best fit each observation; we are in favor of iteratively minimizing quadratic function for the estimation of the optimal state, as there are many mature iterative minimizing algorithm dedicated to large dimension problem. Explicit sequential solver as the EnKF, on the other hand, have difficulties to handle large dimension problem. Finally, lost of balance with respect to the state variables is a weakness of the localization techniques used in EnKF. Some adjustment may be made to alleviate this problem.

Our ensemble-based 4DVar scheme is composed of three phases: first the ensemble generation and forecast; then the minimization of the cost function expressed on the basis of a low rank approximation of the $\mathbf{B}$ matrix; finally, the background error covariance update.

Table 5.1 – Summary of characteristic between incremental 4DVar and EnKF

	Incremental 4DVar	EnKF
Covariance Statistical Model	Not needed, Non-Gaussian errors allowed	Prior pdf of background error and observation error needed usually Gaussian
Background Errors	Fix at the beginning, Implicit evolution	Derived from ensemble, Flow dependent, can be quite noisy
Ability to Update the Background Error Covariance	No	Yes, the posterior background error covariance can be estimated directly
Localization	Not needed	Needed
Tangent Linear Adjoint Models	Needed	Not needed
Analysis Methods	Simultaneously assimilate all observations	Sequentially assimilate batches of observations
Estimation Methods	Iteratively minimizing a quadratic cost function: quasi-Newton methods	Explicitly calculate the analysis by Kalman gain matrix
Balance Constraints	Yes	Slightly lost in localization

5.1.1 The ensemble generation and forecast

We have already discussed in section 3.2.1 certain aspects of ensemble generation used in EnKF method. Identical approaches can be applied to our ensemble-based 4DVar system without any modification. Due to the way the ensemble forecast is explicitly used in the minimization process of the cost function (5.5), it is imperative to have an initial ensemble that approximate well the initial background error statistics. Besides, the error growth according to the forecast model must also be consistent with the real errors. The ensemble generation is related to the sampling strategies (Evensen, 2004) and relies on *ad hoc* evaluation of the dynamic system. In this section we focus on the error growth with respect to the two ensemble update schemes associalted either to perturbed observations (Burgers et al., 1998), or to direct transformation of the background ensemble (Bishop et al., 2001).

The initial error statistics can be classified roughly in terms of their Gaussianity or non-Gaussianity. A Gaussian error model is widely used in data assimilation approaches and is justified under linear or weakly nonlinear models: if the initial error is Gaussian, it will remain Gaussian in such a case. On the other hand, a non-Gaussian error model is usually related to the non-linear dynamic models. Lawson and Hansen (2004) has compared the performance of both filters with respect to the error growth under various contexts. They argued that the perturbed observation filter deals better with nonlinearities than the direct transformation filter, and the perturbed observation filter has a tendency to transform initial non Gaussian error to a Gaussian distribution.

In this thesis, both Gaussian and non-Gaussian errors will be studied.

5.1.2 Low rank approximation of the background error covariance matrix

The ensemble-based 4DVar scheme is defined within the framework of preconditioned incremental variation system (2.68) while handling an empirical approximation of the background error covariance matrix (Liu et al., 2008; Buehner, 2005). This low rank approximation of the background covariance matrix is directly inspired from the Ensemble Kalman filter where the covariance terms are estimated from the spread of an ensemble of samples. The central idea of ensemble method is to use the sample mean to represent the real unknown state. By approximating $\langle \mathbf{x}^b \rangle \approx \mathbf{x}^t$, we have a direct ensemble formulation of the background error covariance matrix $\mathbf{B}$:

$$\mathbf{B} \approx \frac{1}{N-1}(\mathbf{x}^b - \langle \mathbf{x}^b \rangle)(\mathbf{x}^b - \langle \mathbf{x}^b \rangle)^T. \tag{5.1}$$

Nevertheless, the huge dimension of matrix $\mathbf{B} \in \mathbb{R}^{n \times n}$ prohibits this calculation explicitly. Practically, the error covariance $\mathbf{B}$ can only be built and stored implicitly from a low dimensional descriptor. Fortunately, as the ensemble-based 4D variational assimilation scheme relies directly on a preconditioned incremental form of the cost function, we only need to manipulate the square root of $\mathbf{B}$. By handling an empirical approximation of the background covariance matrix (Lorenc, 2003; Buehner, 2005; Liu et al., 2008), we define the square root of the background error covariance matrix as:

$$\mathbf{B}^{\frac{1}{2}} \approx \mathbf{A}'_b := \frac{1}{\sqrt{N-1}}(\mathbf{x}_0^{(1)b} - \langle \mathbf{x}_0^b \rangle, \ldots, \mathbf{x}_0^{(N)b} - \langle \mathbf{x}_0^b \rangle), \tag{5.2}$$

and $\mathbf{A}'_b \in \mathbb{R}^{n \times N}$. In almost all cases, $N \ll n$, thus $\mathbf{A}'_b$ remains easy to manipulate. $\mathbf{A}'_b$ is also called the ensemble perturbation matrix. It gathers the N zero mean centered background ensemble members as a low-dimensional approximation of the background matrix:

$$\mathbf{B} \approx \mathbf{A}'_b \mathbf{A}'^T_b.$$

By introducing such a background covariance approximation in the preconditioned cost function (2.68), we get

$$J(\delta \mathbf{z}_0) = \frac{1}{2}\|\delta \mathbf{z}_0\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}} \mathbb{H}\, \partial_{\mathbf{x}} \varphi_t(\mathbf{x}_0) \mathbf{A}'_b \delta \mathbf{z}_0 - \mathbf{D}(t,x)\|^2_{\mathbf{R}} dt. \tag{5.3}$$

Note that here we do not employ the hybrid CVT form of Eq.(4.2) as done in Desroziers et al. (2014) since one major motivation for our proposed scheme is to preclude the employment of the linear tangent and adjoint operators. Introducing an auxiliary control vector based on the transformation of a full rank static matrix will unfortunately neutralize this purpose. The control vector, which depends solely on the transformation through the ensemble-based error covariance, can also be interpreted as the analysis increments that belong to the subspace spanned by the ensemble perturbations members. This is actually consistent with the assumption made in different methods of E4DVar, 4D-LETKF, *etc.*

The cost function (5.3) inherits the advantage of preconditioned system such as suppressing the presence of $\mathbf{B}$ in the background term, which can also be depicted as the uncorrelation of the background error w.r.t new variable $\delta \mathbf{z}_0$. The whole term $\partial_{\mathbf{x}} \varphi_t(\mathbf{x}_0) \mathbf{A}'_b$ handles the propagation of ensemble perturbation matrix $\mathbf{A}'_b$, which also corresponds to a forecast by the dynamical model of the centered square-root background covariance matrix. As we rely here on an empirical description of this matrix from a set of samples, we

can observe that integrating these samples in time provides us immediately an empirical expression of a low-rank approximation of the background covariance trajectory and of its square root. This avoids thus the employment of the adjoint operator. We set hence:

$$\widetilde{\mathbf{B}}_t^{\frac{1}{2}} = \partial_{\mathbf{x}} \varphi_{\,t}(\mathbf{x}_0)\mathbf{A}'_b. \tag{5.4}$$

The calculation of the propagation of ensemble perturbation matrix projected into observation space corresponds to the way the forecast error is approximated in extended Kalman filter,

$$\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\,t}(\mathbf{x}_0)\mathbf{A}'_b \approx \frac{1}{\sqrt{N-1}}(\mathbb{H}(\varphi_t(\mathbf{x}_0^{(1)b})) - \mathbb{H}(\varphi_t(\langle \mathbf{x}_0^b \rangle)), \ldots, \mathbb{H}(\varphi_t(\mathbf{x}_0^{(N)b})) - \mathbb{H}(\varphi_t(\langle \mathbf{x}_0^b \rangle))), \tag{5.5}$$

or alternatively,

$$\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\,t}(\mathbf{x}_0)\mathbf{A}'_b \approx \mathbb{H}\{\varphi_t(\mathbf{X}_0)(\mathbf{I}_N - \frac{\mathbf{1}\mathbf{1}^T}{N})\}, \tag{5.6}$$

with $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^N$. By doing so, we can well retain the non-linearity of $\mathbb{H}$ and $\mathbb{M}$.

The iterative minimization of cost function (5.3) requires the expression of the gradient:

$$\partial_{\delta\mathbf{z}_0} J = \delta\mathbf{z}_0 + \int_{t_0}^{t_f} \widetilde{\mathbf{B}}_t^{\frac{1}{2}T}\ (\partial_{\mathbf{x}}\mathbb{H})^*\ \mathbf{R}^{-1}(\partial_{\mathbf{x}}\mathbb{H}\ \widetilde{\mathbf{B}}_t^{\frac{1}{2}}\delta\mathbf{z}_0 - \mathbf{D}(t,x))dt. \tag{5.7}$$

and its Hessian reads:

$$\widetilde{\mathcal{H}} = \mathbb{I} + \int_{t_0}^{t_f} \widetilde{\mathbf{B}}_t^{\frac{1}{2}T}\ (\partial_{\mathbf{x}}\mathbb{H})^*\ \mathbf{R}^{-1}\partial_{\mathbf{x}}\mathbb{H}\ \widetilde{\mathbf{B}}_t^{\frac{1}{2}}dt. \tag{5.8}$$

Once estimated the minimizer $\widehat{\delta\mathbf{z}}_0$, the analysis reads:

$$\mathbf{x}_0^a = \mathbf{x}_0^b + \widetilde{\mathbf{A}'_b}\widehat{\delta\mathbf{z}}_0. \tag{5.9}$$

Let us emphasize that, as the covariance matrix $\widetilde{\mathbf{B}}$ is at most of rank $N-1$, the control variable has at most $N-1$ non null components in the eigenspace. Compared to the full 4DVar approach, the control variable's degrees of freedom are thus considerably lowered and the minimization computational complexity is significantly decreased. Indeed, this ensemble version has a lower computation cost if the ensemble forecasting step is distributed on a grid computing.

5.1.3 Background error covariance matrix update

As mentioned earlier, in this study we focus on situations where the background state is only poorly known. It is hence essential to allow in the estimation process a substantial deviation from the background state. So unlike typical incremental ensemble-based variational methods which keep fixed background covariance and apply a single outer loop of the Gauss-Newton minimization, we propose to update the background covariance between two consecutive outer loops. In the context of using an incremental approach and the preconditioning technique, this update actually corresponds to the change of preconditioning matrix. This proposition is consistent with the principle of the iterative ensemble Kalman filter used in (Sakov et al., 2012) where the authors argues the necessity of repropagation of the ensemble fields between two consecutive iterations in strong non-linear systems. The

update of the background error covariance can be either derived from the ensemble analysis defined from perturbed observations or by a direct transformation of the background ensemble perturbations.

Note that we obtain the analysis error covariance matrix exactly in the same way as for the background error covariance. The only difference being that, the analysis error covariance matrix is available only at the end of final outer loop process.

In terms of the ensemble updates, the methods used in EnKF must be however rearranged and justified before any applications. The main reason is that the EnKF is a sequential method which assimilate the observation data whenever they are available during the forward integration of the ensemble, while the 4DEnVar method is a variational technique which assimilate all the observation data throughout the entire assimilation window simultaneously.

The perturbed observation method

The perturbed observation method is used in the 4DEnVar by Liu et al. (2009) and in the EDA-S by Fairbairn et al. (2013) to calculate the analysis error covariance matrix at the end of one assimilation cycle. In its application, their approach based on the EnKF algorithm proposed by Houtekamer and Mitchell (1998, 2005), involves generating a perturbed observation fields to get the ensemble analysis. Before the update procedure, we must be in possession of an initial ensemble $\mathbf{X}$ generated according to the initial error statistics and the ensemble perturbation matrix $\mathbf{A}'_b$. Perturbing the observations with error observation samples constitutes *de facto* an ensemble of observations with consistent statistics,

$$\mathcal{Y}^{i,j} = \mathcal{Y}^i + \epsilon^{i,j}, \quad i = 0, \ldots, T, \quad j = 1, \ldots, N. \tag{5.10}$$

In practice ϵ is usually assumed to be distributed from a multivariate normal distribution:

$$\epsilon_t \sim N(0, \mathbf{R_t}). \tag{5.11}$$

The innovation vector of the jth member of initial ensemble $\mathbf{x}_0^{(j),k}$ at outer loop iteration k is defined as,

$$\mathbf{D}^{(j),k}(t,x) = \mathcal{Y}^j(t,x) - \mathbf{H}(\varphi_t(\mathbf{x}_0^{(j),k})), \quad j = 1, \ldots, N, \tag{5.12}$$

and a parallel realizations of minimization with regard to each member of the initial ensemble can be conducted,

$$\begin{aligned} \delta\mathbf{z}_0^{(j),k} &= (\mathbf{A}_b'^k)^{-1}\delta\mathbf{x}_0^{(j),k}, \quad j = 1, \ldots, N \\ J(\delta\mathbf{z}_0^{(j),k}) &= \frac{1}{2}\|\delta\mathbf{z}_0^{(j),k}\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0)\mathbf{A}_b'^k\delta\mathbf{z}_0^{(j),k} - \mathbf{D}^{(j),k}(t,x)\|^2_{\mathbf{R}}dt. \end{aligned} \tag{5.13}$$

Parallel descent processes provide a matrix $\delta\hat{\mathbf{Z}}_0$ representative of the analysis incremental ensembles. The updated initial ensemble field and its perturbation matrix finally read:

$$\begin{aligned} \mathbf{x}_0^{(j),k+1} &= \mathbf{x}_0^{(j),k} + \mathbf{A}_b'^k\widehat{\delta\mathbf{z}_0}^{(j),k}, \quad j = 1, \ldots, N \\ \mathbf{A}_b'^{k+1} &= \frac{1}{\sqrt{N-1}}(\mathbf{x}_0^{(1),k+1} - \langle\mathbf{x}_0^{k+1}\rangle, \ldots, \mathbf{x}_0^{(N),k+1} - \langle\mathbf{x}_0^{k+1}\rangle). \end{aligned} \tag{5.14}$$

The initial ensemble can then be updated and integrated for a new assimilation cycle. At convergence, this ensemble forecast is finally used as the initial ensemble for the next assimilation process.

The direct transformation method

The deterministic approach formulates the analysis error covariance as a linear combination of the background error $(\mathbf{x}_b - \mathbf{x}_t)$. This approach can take many forms as the transformation matrix is not unique, here we opt for a mean preserving transformation as used in Ensemble Transform Kalman filter (section 3.2.2). However, the standard ensemble transformation matrix used in ETKF is only a 3D[1] matrix. In variational system, the posterior error covariance matrix must be obtained in a way that benefits from all the observation available throughout the assimilation window. In the case of the perturbed observation strategy, the posterior error covariance matrix is calculated from the posterior ensemble.

In this approach, in order to obtain such a transformation matrix, we must consider this problem from a different perspective. We have already shown in section 2.2 that the equivalence between the analysis error covariance matrix and the inverse Hessian matrix of the cost function with regard to 4DVar system holds rigorously in a linear sense. Indeed we rely on such equivalence to deduce the transformation matrix. However, in a nonlinear scenario, Zupanski (2005) suggests this relationship is still valid with non-linear observation operators; whereas for nonlinear dynamic model, this relationship constitutes only an approximation (Li and Navon, 2001).

The inverse analysis error covariance matrix approximating the Hessian corresponds to the incremental 4DVar form (2.45):

$$(\mathbf{P}^a)^{-1} \approx \mathcal{H}_{\mathbf{B}} = \mathbf{B}^{-1} + \int_{t_0}^{t_f} \partial_{\mathbf{x}} \varphi_t^* \partial_{\mathbf{x}} \mathbb{H}^* \mathbf{R}^{-1} \partial_{\mathbf{x}} \mathbb{H} \partial_{\mathbf{x}} \varphi_t dt. \tag{5.15}$$

If we insert the ensemble representation of background error covariance matrix (5.3), we then get:

$$(\mathbf{P}^a)^{-1} \approx \mathcal{H}_{\mathbf{B}} = (\mathbf{A}'_b \mathbf{A}'^{T}_b)^{-1} + \int_{t_0}^{t_f} \partial_{\mathbf{x}} \varphi_t^* \partial_{\mathbf{x}} \mathbb{H}^* \mathbf{R}^{-1} \partial_{\mathbf{x}} \mathbb{H} \partial_{\mathbf{x}} \varphi_t dt. \tag{5.16}$$

Note that the ensemble perturbation matrix $\mathbf{A}'_b$ is not invertible since it is a singular matrix of reduced rank. Pseudo inverse must be employed here. Pre-multiplying and post-multiplying each side of the above equation with $\mathbf{A}'^{T}_b$ and $\mathbf{A}'_b$ respectively cancels the inverse background error term:

$$\mathbf{A}'^{T}_b (\mathbf{P}^a)^{-1} \mathbf{A}'_b \approx \mathbf{A}'^{T}_b \mathcal{H}_{\mathbf{B}} \mathbf{A}'_b = \mathbb{I} + \int_{t_0}^{t_f} (\partial_{\mathbf{x}} \varphi_t(\mathbf{x}_0) \mathbf{A}'_b)^T \left(\partial_{\mathbf{x}} \mathbb{H}\right)^* \mathbf{R}^{-1} \partial_{\mathbf{x}} \mathbb{H}\ \partial_{\mathbf{x}} \varphi_t(\mathbf{x}_0) \mathbf{A}'_b dt. \tag{5.17}$$

The RHS of the Eq.(5.17) is actually the Hessian matrix of the preconditioned cost function (5.3). Let us denote this Hessian matrix as $\mathcal{H}_{\mathbb{I}}$:

$$\mathbf{A}'^{T}_b (\mathbf{P}^a)^{-1} \mathbf{A}'_b \approx \mathcal{H}_{\mathbb{I}}, \tag{5.18}$$

$$(\mathbf{P}^a)^{-1} \approx (\mathbf{A}'^{T}_b)^{-1} \mathcal{H}_{\mathbb{I}} (\mathbf{A}'_b)^{-1}, \tag{5.19}$$

$$\mathbf{P}^a \approx \mathbf{A}'_b \mathcal{H}_{\mathbb{I}}^{-1} \mathbf{A}'^{T}_b. \tag{5.20}$$

The analysis error covariance matrix can be written as a product of the analysis ensemble perturbation $\mathbf{A}'_a$,

$$\mathbf{A}'_a \mathbf{A}'^{T}_a \approx \mathbf{A}'_b \mathcal{H}_{\mathbb{I}}^{-1} \mathbf{A}'^{T}_b, \tag{5.21}$$

[1] Here the term 3D is used in analogy to 3DVar against 4D.

Denoting by $\mathcal{H}_{\mathbb{I}}^{\frac{1}{2}}$ the square root of $\mathcal{H}_{\mathbb{I}}$,

$$\mathcal{H}_{\mathbb{I}} = \mathcal{H}_{\mathbb{I}}^{\frac{1}{2}^T} \mathcal{H}_{\mathbb{I}}^{\frac{1}{2}}, \tag{5.22}$$

then the analysis ensemble perturbation reads,

$$\mathbf{A}'_a \approx \mathbf{A}'_b \mathcal{H}_{\mathbb{I}}^{-\frac{1}{2}}. \tag{5.23}$$

This equation takes the same form as (3.23) used in ETKF. However, the implications of ensemble transform matrix are quite different. The updated initial ensemble perturbations can be expressed as the Hessian square-root computed from previous perturbation matrix at outer loop iteration k,

$$\mathbf{A}_b'^{k+1} = \mathbf{A}_b'^{k} \{ \frac{1}{\alpha} \mathbb{I} + \int_{t_0}^{t_f} \widetilde{\mathbf{B}}_t^{\frac{1}{2}^T} (\partial_{\mathbf{x}} \mathbb{H})^* \ \mathbf{R}^{-1} \partial_{\mathbf{x}} \mathbb{H} \ \widetilde{\mathbf{B}}_t^{\frac{1}{2}} dt \}^{-\frac{1}{2},k} \mathbf{V}. \tag{5.24}$$

Note that the parameter α corresponds to an inflation coefficient commonly used in ensemble filter and the arbitrary orthogonal matrix $\mathbf{V}$ is used to center the posterior ensemble on the updated background/analysis.

As the minimization algorithm LBFGS relies on an approximation of the inverse Hessian matrix $\mathcal{H}^{-1}$, we can use this byproduct to evaluate equation (5.24).

Note that a relationship similar to (5.24) can be found in the IEnKS method introduced in Bocquet and Sakov (2013b). Their approach is obtained in the context of ensemble square root Kalman smoother; a relatively different form can be found in 4D-LETKF method of Hunt et al. (2007).

At the initial time, the background matrix is fixed from the initial random conditions chosen. Note that in the direct transformation approach, a single minimization process is conducted with respect to the background state opposite to the previous cases where the minimization has to be done with respect to each member of the ensemble plus the background state. Finally the updated initial ensemble fields are,

$$\mathbf{x}_0^{(j),k+1} = \widehat{\mathbf{x}}_0^k + \sqrt{N-1} \mathbf{A}_b'^{(j)k+1}, \quad j = 1, \ldots, N \tag{5.25}$$

where $\widehat{\mathbf{x}}_0^k$ corresponds to the updated initial state after the kth iteration of outer loop,

$$\widehat{\mathbf{x}}_0^k = \mathbf{x}_0^k + \mathbf{A}_b'^k \widehat{\delta \mathbf{z}_0}^k \tag{5.26}$$

Both variants of the update will be assessed in the experimental section. A schematic representation of this method in shown in algorithm 9. Note that in algorithm 9, we present the perturbed observation update scheme, whereas the direct transformation scheme is shown in algorithm 11 along with the localization technique to avoid repetition.

5.1.4 Localization issues

The previous ensemble method relies on a low rank approximation of the background matrix. Such an empirical approximation built from only very few samples, when compared to the state space dimension, leads in practice to spurious correlations between distant points. For ensemble Kalman techniques, it is customary to remove these long distance correlations through localization procedures. There are generally two methods to filter such pseudo-correlations.

Covariance localization

The first approach introduces a Schur element-wise product between the background correlation matrix and a local isotropic correlation function, thus the sample error problem can be alleviated:

$$\mathbf{P}^b = \mathcal{C} \odot \mathbf{P}^e, \tag{5.27}$$

The spatial correlation function can be simply defined as a matrix $\mathcal{C}(\|x-y\|/L)$ in which we set $\mathcal{C}_{xy} = 0$ when the distance between x and y exceeds the cutoff distance L. Polynomial approximations of a Gaussian function with compact support and a hard cutoff are often employed Gaspari and Cohn (1999) to that end. They lead to sparse correlation matrices, which is computationally advantageous. One popular choice for $\mathcal{C}$ is in the form of Eq.(4.10) in Gaspari and Cohn (1999).

$$\mathcal{C}_0(z, 1/2, c) = \begin{cases} f_1(z/c) & 0 \le |z| \le c \\ f_2(z/c) & c \le |z| \le 2c \\ 0 & 2c \le |z| \end{cases}$$

where $2c$ equals the cutoff distance and f_1, f_2 are even functions given by

$$f_1(z) = -\frac{z^5}{4} + \frac{z^4}{2} + \frac{5z^3}{8} - \frac{5z^2}{3} + 1, \qquad 0 \le z \le 1,$$
$$f_2(z) = \frac{z^5}{12} - \frac{z^4}{2} + \frac{5z^3}{8} + \frac{5z^2}{3} - 5z + 4 - \frac{2}{3z}, \qquad 1 \le z \le 2.$$

In order to incorporate the localized background error matrix into our system, we approximate the square root of $\mathbf{P}_b$ by a spectral decomposition of the isotropic correlation function:

$$\mathcal{C} = E\lambda E^T, \tag{5.28}$$

where E corresponds to Fourier modes. This allows us to filter the remaining high frequency components that may lead to erroneous propagation of spurious high frequency gravity waves generated only by noise. Keeping only the r first leading Fourier modes, we define the approximated correlation square root as:

$$\mathcal{C}' = E_{n\times r}\lambda^{1/2}_{r\times r}. \tag{5.29}$$

The modified perturbation matrix is then provided by

$$\mathbf{P}'_b = (diag(\mathbf{A}'^{(1)}_b)\mathcal{C}', \ldots, diag(\mathbf{A}'^{(N)}_b)\mathcal{C}'). \tag{5.30}$$

Here the *diag* operator sets the kth vector of $\mathbf{A}'^{(k)}_b$ as the diagonal of a matrix. This localized perturbation matrix will serve us to precondition the assimilation system associated with (5.3) and (5.7).

$$J(\delta\mathbf{r}_0) = \frac{1}{2}\|\delta\mathbf{r}_0\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{P}'_b\delta\mathbf{r}_0 - \mathbf{D}(t,x)\|^2_{\mathbf{R}}dt, \tag{5.31}$$

$$\partial_{\delta\mathbf{r}_0}J = \delta\mathbf{r}_0 + \int_{t_0}^{t_f} \widetilde{\mathbf{B}}_t^{\frac{1}{2}T}(\partial_{\mathbf{x}}\mathbb{H})^*\ \mathbf{R}^{-1}(\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{P}'_b\delta\mathbf{r}_0 - \mathbf{D}(t,x))dt. \tag{5.32}$$

where the forecast background perturbation $\widetilde{\mathbf{B}}_t^{\frac{1}{2}}$ reads according to (5.4),

$$\widetilde{\mathbf{B}}_t^{\frac{1}{2}} = \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{P}'_b. \tag{5.33}$$

Note that here the analysis increment belongs to the subspace spanned by the localized ensemble perturbation matrix $\mathbf{P}'_b$ with dimension: $n \times N \times r$. The increased subspace is another advantage against the non-localized version in which the subspace is spanned by the ensemble. Nevertheless the increased dimension of control vector ($N \times r$ against N) could require more computational cost in terms of minimization process.

The calculation of the propagation of localized ensemble perturbation matrix projected into observation space follows the same principle as (5.5),

$$\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{P}'_b \approx \partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(diag(\mathbf{A}'_{b,1})\mathcal{C}', \ldots, diag(\mathbf{A}'_{b,N})\mathcal{C}') \tag{5.34}$$

Each ensemble term in the RHS of equation (5.34), $\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)diag(\mathbf{A}'_{b,j})\mathcal{C}'$ can be expanded as,

$$\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(diag(\mathbf{A}'_{b,j})\mathcal{C}') = \partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)((\mathbf{A}'_{b,j}, \ldots, \mathbf{A}'_{b,j})_r \odot \mathcal{C}'), \tag{5.35}$$

where we recall that the truncated matrix $\mathcal{C}' = E_{n\times r}\lambda^{1/2}_{r\times r}$ is the square root of the spatial correlation matrix, and the effect of Schur product is to modify each item of the matrix $(\mathbf{A}'_{b,j}, \ldots, \mathbf{A}'_{b,j})_r$ by a coefficient. Assuming this later expression can be transformed as:

$$\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(diag(\mathbf{A}'_{b,j})\mathcal{C}') \approx \partial_{\mathbf{x}}\mathbb{H}\ ((\partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(\mathbf{A}'_{b,j}, \ldots, \mathbf{A}'_{b,j})_r) \odot \mathcal{C}'), \tag{5.36}$$

Here we changed the computation priority compare to Eq.(5.35): in Eq.(5.35), the background ensemble perturbation is localized first, prior to the propagation step; in Eq.(5.36), the process of multiplying the correlation matrix is done after the propagation of unlocalized ensemble perturbation fields. The latter strategy is preferred because the evolution of the localized perturbation may severely deteriorate the ensemble spread. However, Eq.(5.36) is only an approximation of Eq.(5.35).

Evaluation of Eq.(5.36) is analogy to the approximation of the term $\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{A}'_{b,j}$ by Eq.(5.5), however, some attention must be paid to the observation operator $\mathbb{H} \in \mathbb{R}^{m\times n}$. If the dimension of the observation space m is not equal to the dimension of the state vector n, a direct application of Eq.(5.5) is not possible since our correlation matrix is an $n \times r$ matrix. In other words, our localized scheme is applied to the model space rather than to the observation space. So the Schur product must be done to the ensemble perturbation matrix at different observation times before it is transformed into the observation space.

$$\begin{aligned}&(\partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(\mathbf{A}'_{b,j}, \ldots, \mathbf{A}'_{b,j})_r) \odot \mathcal{C}' \approx \\ &\frac{1}{\sqrt{N-1}}\{\big(\varphi_t(\mathbf{x}_0^{(j),b}) - \varphi_t(\langle\mathbf{x}_0^b\rangle), \ldots, \varphi_t(\mathbf{x}_0^{(j),b}) - \varphi_t(\langle\mathbf{x}_0^b\rangle)\big)_r \odot \mathcal{C}'\}.\end{aligned} \tag{5.37}$$

We can also use the *diag* operator,

$$(\partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(\mathbf{A}'_{b,j}, \ldots, \mathbf{A}'_{b,j})_r) \odot \mathcal{C}' \approx \frac{1}{\sqrt{N-1}}\{diag\big(\varphi_t(\mathbf{x}_0^{(j),b}) - \varphi_t(\langle\mathbf{x}_0^b\rangle)\big)\mathcal{C}'\}. \tag{5.38}$$

Then the observation operator is applied to the jth localized ensemble forecast,

$$\begin{aligned}&\partial_{\mathbf{x}}\mathbb{H}\ (\partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)(\mathbf{A}'_{b,j}, \ldots, \mathbf{A}'_{b,j})_r) \odot \mathcal{C}' \approx \\ &\frac{1}{\sqrt{N-1}}\big(\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b}) \odot \mathcal{C}'_1) - \mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle) \odot \mathcal{C}'_1), \ldots, \mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b}) \odot \mathcal{C}'_r) - \mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle) \odot \mathcal{C}'_r)\big),\end{aligned} \tag{5.39}$$

where the $\mathcal{C}'_i$ corresponds the the ith column of matrix $\mathcal{C}'$. Finally we have the full expression

$$\partial_{\mathbf{x}}\mathbb{H}\,\partial_{\mathbf{x}}\varphi_{\,t}(\mathbf{x}_0)\mathbf{P}'_b \approx \tag{5.40}$$
$$\frac{1}{\sqrt{N-1}}\{\big(\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b})\odot\mathcal{C}'_1)-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle)\odot\mathcal{C}'_1),\ldots,\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b})\odot\mathcal{C}'_r)-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle)\odot\mathcal{C}'_r)\big)$$
$$\ldots,\big(\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b})\odot\mathcal{C}'_1)-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle)\odot\mathcal{C}'_1),\ldots,\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b})\odot\mathcal{C}'_r)-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle)\odot\mathcal{C}'_r)\big)\}.$$

In the applications with an observation space fitting exactly the model space ($m=n$), we could have a simpler expression of $\partial_{\mathbf{x}}\mathbb{H}\,\partial_{\mathbf{x}}\varphi_{\,t}(\mathbf{x}_0)\mathbf{P}'_b$ as the effects of localizing the model space and the observation space are the same. This kind of situation is actually very common in image data assimilation where the resolution of image data is at least equal to the resolution of model space.

$$\partial_{\mathbf{x}}\mathbb{H}\,\partial_{\mathbf{x}}\varphi_{\,t}(\mathbf{x}_0)\mathbf{P}'_b \approx \tag{5.41}$$
$$\frac{1}{\sqrt{N-1}}\{\big(\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b}))-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle)),\ldots,\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b}))-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle))\odot\mathcal{C}'\big)$$
$$\ldots,\big(\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b}))-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle)),\ldots,\mathbb{H}(\varphi_t(\mathbf{x}_0^{(j),b}))-\mathbb{H}(\varphi_t(\langle\mathbf{x}_0^b\rangle))\big)\odot\mathcal{C}'\},$$

Note that for our middle-size data system ($n\sim\mathcal{O}(10^5)$), this evaluation scheme is computationally realizable, however, for operational applications, a more effective way of evaluating the localized ensemble covariance must be adopted (Bishop et al., 2011).

Finally the analysis reads:

$$\mathbf{x}_0^a=\mathbf{x}_0^b+\widetilde{\mathbf{P}'_b}\widehat{\delta\mathbf{r}}_0. \tag{5.42}$$

Let us remark that this approach is incompatible with the direct transformation method for the background error covariance matrix updating. This is due to the inconsistency of matrix dimensions when updating the initial ensemble based on Eq.(5.25). As the dimension of $\mathbf{P}'_b$ is $n\times r\times N$ instead of $n\times N$, its jth component from the ensemble index is of size $n\times r$, therefore the ensemble perturbation can not be recovered from its localized counterpart.

Local ensemble

Another localization technique proposed by Ott et al. (2004) leads the analysis step through an EnKF procedure in a local region. This approach involves a transformation $\mathbf{M}_{l,s}\in\mathbb{R}^{l\times n}$ from the state space $\mathbb{R}^n$ to a smaller local space $\mathbb{R}^l$. The local vector is defined as:

$$\mathbf{x}_l=\mathbf{M}_{l,s}\mathbf{x}_n. \tag{5.43}$$

In the very same spirit, we can also define a transformation $\mathbf{M}_{l,o}\in\mathbb{R}^{l\times m}$ from the observation space $\mathbb{R}^n$ to local space $\mathbb{R}^l$,

$$\mathcal{Y}_l=\mathbf{M}_{l,o}\mathcal{Y}_m. \tag{5.44}$$

Then the minimization and the update of error covariance steps are done in local space $\mathbb{R}^l$ around each grid point only incorporating the model points and observations within a certain range. This range, denoted as l, and which corresponds to the concept of cut-off distance aforementioned, determines the size of the local space.

Note that the ensemble forecast step must be done globally with the full non-linear dynamic model.

The way of evaluating the local space error covariance is done by calculating the ensemble-based error covariance from the local space ensemble:

$$\begin{aligned}\mathbf{A}'_{b,l} &= \frac{1}{\sqrt{N-1}}(\mathbf{x}^{(1)b}_{0,l} - \langle \mathbf{x}^{b}_{0,l}\rangle, \ldots, \mathbf{x}^{(N)b}_{0,l} - \langle \mathbf{x}^{b}_{0,l}\rangle),\\ \partial_{\mathbf{x}}\mathbb{H}\,\partial_{\mathbf{x}}\varphi_{\,t}(\mathbf{x}_0)\mathbf{A}'_{b,l} &= \frac{1}{\sqrt{N-1}}(\mathbb{H}(\varphi_t(\mathbf{x}^{(1)b}_{0,l})) - \mathbb{H}(\varphi_t(\langle \mathbf{x}^{b}_{0,l}\rangle)), \ldots, \mathbb{H}(\varphi_t(\mathbf{x}^{(N)b}_{0,l})) - \mathbb{H}(\varphi_t(\langle \mathbf{x}^{b}_{0,l}\rangle))).\end{aligned} \tag{5.45}$$

As a matter of fact, by posing certain conditions, we can show that the localized error covariance matrix yielded by the two localization techniques are indeed equivalent. These conditions include:

Employing a simple hard cutoff correlation matrix The covariance localization correlation matrix corresponds for instance to a polynomial approximations of a Gaussian function with compact support. Within the covariance support region, the correlation of one point takes the value of 1 for its variance and reduces along the distance following a Gaussian shaped function. If we instead use a simple hard cutoff correlation matrix that takes value 1 within the support region and 0 outside, we have in principle the same form of background error covariance as the local ensemble approach that takes no weight at all for those points outside the local space.

Introducing local correlation matrix $\mathcal{C}_l$ to local ensemble approach In a reverse way, we could introduce a similar Gaussian-like mask to filter the local ensemble error covariance matrix, then employ a SVD to obtain the gaussian-filtered local perturbation matrix.

Although the two approaches may in theory use the same kind of structure of the error covariance, the analysis resulted from the two localization methods are very different. This is probably due to the way of solving the analysis increment from the minimization process. The localized covariance approach, on the one hand, performs the optimization on subspace spanned by the localized perturbation ensemble matrix with $N \times r$ columns against the n dimensional analysis increment; the local ensemble approach, on the other hand, relies on the CVT associated with the local ensemble perturbation matrix,

$$\delta\mathbf{x}^{a}_{0,l} = \mathbf{A}'_{b,l}\delta\mathbf{z}_0. \tag{5.46}$$

The analysis increment lives therefore in the subspace spanned by $\mathbf{A}'_{b,l}$, since the variation of $\delta\mathbf{x}^{a}_{0,l}$ is limited in local space, such an approximation makes more sense compared to the global space approach where the full space analysis uncertainties are represented by very few ensemble members.

It will be hard to tell which localization technique is better at this point, however, an intuitive indication would be that the global analysis obtained directly from the minimization process may better preserve the continuity and the multi-variables balance; while the global analysis reconstructed from different and independent local analysis may have difficulties in dealing with discontinuity and tends to lose multi-variables balance.

Both update strategies of background error covariance matrix described in section 5.1.3 can be combined with local ensemble approach without any difficulties. Nevertheless, this localization strategy is ideally compatible with the method of direct transformation approach associated with the update phase. A great advantage of this combination is the low computational cost if properly parallelizing minimization together with a small local space.

5.2 Summary

All these elements associated (i.e. CVT, localization and background error covariance matrix update) with a LBFGS minimization strategy constitute the proposed ensemble method. The algorithm descriptions of the overall methods are presented in algorithms 9, 10 and 11. We point out that our assimilation system with the following characteristic: perturbed observations, only one outer loop and localizing by modifying the covariance matrix is equivalent to the 4DEnVar method proposed by Liu et al. (2008, 2009); if a direct ensemble transformation update is used, it is hence close to the 4D-LETKF method (Hunt et al., 2007; Fertig et al., 2007) and to the IEnKS method (Bocquet and Sakov, 2013a) but with a minimization performed on a variational basis.

Algorithm 9 Ensemble-based variational data assimilation algorithm: : No Localization, Perturbed Observation

procedure ANALYSIS

Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, and the initial ensemble $\mathbf{X}_0^b$ as an arbitrary choice (for the 1st cycle) or as the forecast state and the ensemble forecast derived from the previous assimilation cycle respectively

Given convergence toleration ξ and ϵ

Generate ensemble observations (5.10)

Define matrix $\mathcal{X}_0^1 = [\mathbf{x}_0, \mathbf{X}_0]$.

Outer loop:

for $k = 1 : k_{max}$ **do**

Compute in parallel $\mathcal{X}^k(t)$ with the forward integration of the nonlinear dynamics (1.1)

Calculate ensemble innovations (5.12)

Update the background perturbation matrix $\mathbf{A}_b'^k$ from (5.14) and the term $\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{A}_b'^k$ from (5.5)

Initialize the ensemble increment matrix: $\delta\mathcal{X}_0^k$: $[\delta\mathbf{x}_0, \delta\mathbf{x}_0^{(1)}, \ldots, \delta\mathbf{x}_0^{(N)}]$

Do an inverse control variable transformation $\delta\mathbf{Z}_0^k = (\mathbf{A}_b'^k)^{-1}\delta\mathcal{X}_0^k$

Inner loop:

while $\|\partial_{\delta\mathbf{z}_0^{k,n,j}} J(\delta\mathbf{z}_0^{k,n,j})\| > \epsilon$ **do**

Optimize in parallel $\delta\mathbf{Z}_\mathbf{0}^\mathbf{j}$ in the inner loop, the cost function and the gradient are calculated based on the (5.13)

Iteratively searching for optimizer based on (2.42) or other method (LBFGS, algorithm 5)

end while

Do an control variable transformation

$$\delta\mathcal{X}_0^k = (\mathbf{A}_b'^k)\delta\mathbf{Z}_0^k.$$

Update the matrix $\mathcal{X}_0^k$ containing the initial condition $\mathbf{x}_0^{b,k}$ and ensemble $\mathbf{X}_0^{b,k}$,

$$\mathcal{X}_0^{k+1} = \mathcal{X}_0^k + \delta\mathcal{X}_0^k.$$

Check convergence condition:

if $\|\delta\mathbf{x}_0^k - \delta\mathbf{x}_0^{k-1}\| < \xi$, **then**

$$[\mathbf{x}_0^a, \mathbf{X}_0^a] = \mathcal{X}_0^{k+1}.$$

end if

end for

Evolve the analysis state $\mathbf{x}_0^a$ and the analysis ensemble $\mathbf{X}_0^a$ to the beginning of the next cycle through the nonlinear dynamics (1.1). The forecast state and forecast ensemble are used to initialize the next assimilation cycle.

end procedure

Algorithm 10 Ensemble-based variational data assimilation algorithm: : Localized covariance approach

procedure ANALYSIS

Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, and the initial ensemble $\mathbf{X}_0^b$ as an arbitrary choice (for the 1st cycle) or as the forecast state and the ensemble forecast derived from the previous assimilation cycle respectively

Given convergence toleration ξ and ϵ

Generate ensemble observations (5.10)

Define matrix $\mathcal{X}_0^1 = [\mathbf{x}_0^0, \mathbf{X}_0^b]$

Outer loop:

for $k = 1 : k_{max}$ **do**

Compute in parallel $\mathcal{X}^k(t)$ with the forward integration of the nonlinear dynamics (1.1)

Calculate ensemble innovations (5.12)

Update the background perturbation matrix $\mathbf{A}_b'^k$ from (5.14)

Update the localized perturbation matrix $\mathbf{P}_b'^k$ from (5.30) and its propagation $\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_{\ t}(\mathbf{x}_0)\mathbf{P}_b'$ from (5.45)

Initialize the ensemble increment matrix: $\delta\mathcal{X}_0$: $[\delta\mathbf{x}_0, \delta\mathbf{x}_0^{(1)}, \ldots, \delta\mathbf{x}_0^{(N)}]$

Do an inverse control variable transformation $\delta\mathbf{R}_0^k = (\mathbf{P}_b'^k)^{-1}\delta\mathcal{X}_0^k$

Inner loop:

while $\|\partial_{\delta\mathbf{r}_0^{k,n,j}} J(\delta\mathbf{r}_0^{k,n,j})\| > \epsilon$ **do**

Optimize in parallel $\delta\mathbf{R}_\mathbf{0}^\mathbf{j}$ in the inner loop, the cost function and the gradient are calculated based on the (5.13)

Iteratively searching for optimizer based on (2.42) or other method (LBFGS, algorithm 5)

end while

Do an control variable transformation

$$\delta\mathcal{X}_0 = (\mathbf{P}_b'^k)\delta\mathbf{Z}_0^k.$$

Update the matrix $\mathcal{X}_0^k$ containing the initial condition $\mathbf{x}_0^{b,k}$ and ensemble $\mathbf{X}_0^{b,k}$,

$$\mathcal{X}_0^{k+1} = \mathcal{X}_0^k + \delta\mathcal{X}_0^k.$$

Check convergence condition:

if $\|\delta\mathbf{x}_0^k - \delta\mathbf{x}_0^{k-1}\| < \xi$, **then**

$$[\mathbf{x}_0^a, \mathbf{X}_0^a] = \mathcal{X}_0^{k+1}.$$

end if

end for

Evolve the analysis state $\mathbf{x}_0^a$ and the analysis ensemble $\mathbf{X}_0^a$ to the beginning of the next cycle through the nonlinear dynamics (1.1). The forecast state and forecast ensemble are used to initialize the next assimilation cycle.

end procedure

Algorithm 11 Ensemble-based variational data assimilation algorithm: Local ensemble approach

procedure ANALYSIS

Prescribe the initial condition $\mathbf{x}_0^0 = \mathbf{x}_0^b$, and the initial ensemble $\mathbf{X}_0^b$ as an arbitrary choice (for the 1st cycle) or as the forecast state and the ensemble forecast derived from the previous assimilation cycle respectively

Given convergence toleration ξ and ϵ

Define local space $\mathbf{R}^l$ and the transformation matrix $\mathbf{M}_l$

Define matrix $\mathcal{X}_0^1 = [\mathbf{x}_0^0, \mathbf{X}_0^b]$

Outer loop:

for $k = 1 : k_{max}$ **do**

Compute in parallel $\mathcal{X}^k(t)$ with the forward integration of the nonlinear dynamics (1.1)

Calculate innovation vector $\mathbf{D}_t^k$

Initialize the increment vector $\delta\mathbf{x}_0$

Parallelizing minimization computation in the local space $\mathbf{R}^l$ around each grid point p:

Transform the ensemble background fields, innovation vector and the initial increment to local space

$$\mathbf{X}_{t,l}^k = \mathbf{M}_{l,s}\mathbf{X}_t^k, \tag{5.47}$$

$$\mathbf{D}_{t,l}^k = \mathbf{M}_{l,o}\mathbf{D}_t^k, \tag{5.48}$$

$$\delta\mathbf{x}_{0,l} = \mathbf{M}_{l,s}\delta\mathbf{x}_0, \tag{5.49}$$

Calculate the local background error perturbation matrix and the propagation of local observation ensemble perturbation matrix according to Eqs.(5.45)

Do an inverse control variable transformation $\delta\mathbf{z}_{0,l} = (\mathbf{A}_{b,l}'^k)^{-1}\delta\mathbf{x}_{0,l}$

Inner loop: Solve for the initial increment $\delta\mathbf{z}_{0,l}$ which minimize the problem

$$J(\delta\mathbf{z}_{0,l}^k) = \frac{1}{2}\|\delta\mathbf{z}_{0,l}^k\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\partial_{\mathbf{x}}\mathbb{H}\ \partial_{\mathbf{x}}\varphi_t(\mathbf{x}_0)\mathbf{A}_{b,l}'^k\delta\mathbf{z}_{0,l}^k - \mathbf{D}_{t,l}^k\|_{\mathbf{R}}^2 dt \tag{5.50}$$

while $\|\partial_{\delta\mathbf{z}_{0,l}^{k,n}} J(\delta\mathbf{z}_{0,l}^{k,n})\| > \epsilon$ **do**

Iteratively searching for optimizer based on (2.42) or other method (LBFGS, algorithm 5)

end while

Update the local ensemble perturbation matrix (5.24) and the local initial ensemble(5.25)

Reconstruct the analysis and the ensemble form local space to state space

end for

Evolve the analysis state $\mathbf{x}_0^a$ and the analysis ensemble $\mathbf{X}_0^a$ to the beginning of the next cycle through the nonlinear dynamics (1.1). The forecast state and forecast ensemble are used to initialize the next assimilation cycle.

end procedure

Part III

Applications

Chapter 6

Validation of Ensemble-based 4DVar With Shallow Water Model

In this chapter, a preliminary test will be done with a nonlinear shallow water model. The nonlinear shallow water model is obtained by a depth-integration of the 3D Reynolds-averaged Navier-Stokes equation. The derivation process is done based on the following hypothesis:

- The height of the fluid surface can be neglected compared to its width and length. This implies that we will neglect the vertical shear and vertical acceleration, and only work with the horizontal velocity component $\boldsymbol{u} \triangleq \boldsymbol{u}_H = (u, v)^T$.
- The pressure is hydrostatically distributed along the vertical, that is, $p = p_0 + \rho_0 g(h - z)$
- The fluid is incompressible.

6.1 2D nonlinear shallow water model

A thorough derivation of the shallow water will be provided in Chapter 8 in the context of a stochastic representation.The 2D nonlinear shallow water model with which we will work is described by the following system:

$$\begin{cases} \partial_t \eta + \partial_x(hu) + \partial_y(hv) &= 0, \\ \partial_t u + u\partial_x u + v\partial_y v - fv &= -g\partial_x \eta, \\ \partial_t v + u\partial_x v + v\partial_y v + fu &= -g\partial_y \eta. \end{cases} \tag{6.1}$$

where η denotes the free surface of the evolving fluid, h is the water column height, $\eta = h + b$ where b is known and corresponds to the bottom height with respect to a reference height and u, v are the depth-averaged lengthwise and transverse velocity respectively. Here the state vector corresponds to the concatenation of these three variables: $\mathbf{x} = (\eta, u, v)^T$. The schematic diagram of these notations are shown in figure 6.1 (extracted from Cushman-Roisin and Beckers (2011)).

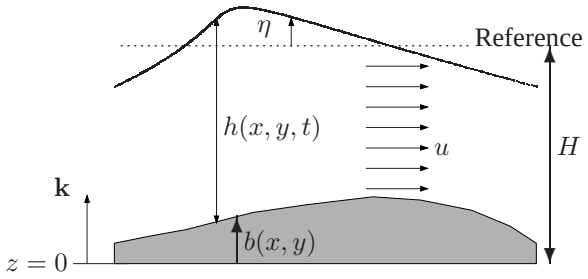


Figure 6.1 – Schematic diagram of unsteady flow over an irregular bottom (Cushman-Roisin and Beckers, 2011).

The conserved form of 2D SWEs is,

$$\frac{\partial \eta}{\partial t} + \frac{\partial (hu)}{\partial x} + \frac{\partial (hv)}{\partial y} = 0, \tag{6.2a}$$

$$\frac{\partial (hu)}{\partial t} + \frac{\partial (hu^2)}{\partial x} + \frac{\partial (huv)}{\partial y} = -gh\partial_x \eta - ghS_b x + hfv + \nu\Big(\frac{\partial (hu_x)}{\partial x} + \frac{\partial (hu_y)}{\partial y}\Big), \tag{6.2b}$$

$$\frac{\partial (hv)}{\partial t} + \frac{\partial (huv)}{\partial x} + \frac{\partial (hv^2)}{\partial y} = -gh\partial_y \eta - ghS_b y - hfu + \nu\Big(\frac{\partial (hv_x)}{\partial x} + \frac{\partial (hv_y)}{\partial y}\Big). \tag{6.2c}$$

We deal with an initial value evolution problem constrained with given boundary conditions. This problem can be compactly written in the form of a symbolic conservation law:

$$\begin{cases} \partial_t \mathbf{U} + \boldsymbol{\nabla}(\mathbf{G}, \mathbf{H}) &= \mathbf{S} + \boldsymbol{\nabla}(\mathbf{G}^d, \mathbf{H}^d), \\ \mathbf{U}(x,0) &= \mathbf{U}_0(x) \\ \mathbf{U}(0,t) &= \mathbf{U}_l(t), \qquad\qquad \mathbf{U}(L,t) = \mathbf{U}_r(t), \end{cases} \tag{6.3}$$

where

$$\mathbf{U} = (\eta, hu, hv)^T, \tag{6.4a}$$

$$\mathbf{G} = (hu, hu^2, huv)^T, \tag{6.4b}$$

$$\mathbf{H} = (hv, huv, hv^2)^T, \tag{6.4c}$$

$$\mathbf{S} = (0, -gh\partial_x \eta - ghS_{bx} + hfv, -gh\partial_y \eta - ghS_{by}\eta - hfu)^T \tag{6.4d}$$

$$\mathbf{G}^d = (0, \nu\partial hu_x, \nu\partial hu_y)^T, \tag{6.4e}$$

$$\mathbf{H}^d = (0, \nu\partial hv_x, \nu\partial hv_y)^T, \tag{6.4f}$$

Here $\mathbf{U}$ is the conserved quantities, $\mathbf{G}$ and $\mathbf{H}$ are the convective flux, $\mathbf{G}^d$ and $\mathbf{H}^d$ are diffusive flux, S_{bx} and S_{by} are the bed friction terms, f is the Coriolis force parameter. It is customary to split the term $gh\partial_i \eta, i = x, y$ between flux gradient and source terms,

$$gh\partial_i \eta = \partial_i \frac{1}{2} gh^2 + gh\partial_i b, \tag{6.5}$$

where $\partial_i b$ is the bottom gradient. Vectors $\mathbf{G}$, $\mathbf{H}$ and $\mathbf{S}$ are thus modified as,

$$\mathbf{G} = (hu, hu^2 + \frac{1}{2}gh^2, huv)^T, \tag{6.6a}$$

$$\mathbf{H} = (hv, huv, hv^2 + \frac{1}{2}gh^2)^T, \tag{6.6b}$$

$$\mathbf{S} = (0, -gh\partial_x b - ghS_{fx} + hfv, -gh\partial_y b - ghS_{by}\eta - hfu)^T \tag{6.6c}$$

The first equation corresponds to our 2D shallow water equation defined by (6.2) which defines how the state vector $\mathbf{x}(t)$ evolves in time. The second and third equations define the initial and boundary conditions of the dynamical system.

The numerical integration of this system, that we will refer in the assimilation techniques as the direct model, determines a trajectory from a given initial state. It corresponds to a traditional simulation of the fluid dynamics from an initial condition. It does not take into account any observations of the state variables. In this section, we will describe first the discrete implementation on which we relied for such a numerical simulation. This model allows providing background trajectories of the system. We will see in the following sections how these trajectories will be corrected by the observations in each assimilation techniques presented in this document.

The literature provides many methods (Vreugdenhil, 1994; Toro, 2009) to solve CFD problems. The most classical methods are the finite difference, finite element and the finite volume methods. The first method is the most natural way to approximate PDE, however it may encounter some difficulties near discontinuities. The finite elements are more suited for solid mechanics and are ideal to approximate complicated domain geometry. At last, the finite volume methods is generally applied to integral form of conservation laws and handles discontinuities in solutions. Among all the proposed methods, we choose to use the finite volume methods which are usually used in CFD. In order to better illustrate the idea of finite volume methods, we will start by introducing a 1D case applied to 1D nonlinear shallow water model. Then the full 2D nonlinear shallow water formulation will be presented.

6.2 1D nonlinear shallow water model

In this section, a one-dimensional nonlinear shallow water model is presented and tested with ensemble-based 4DVar methods. The 1D shallow water model is based on the additional hypothesis:

- There is no flux in the transverse direction.
- To simplify the calculation, we neglect all external forces but the Coriolis force.

Therefore the 1D shallow water model reads,

$$\left\{\begin{array}{lcl} \partial_t \eta + \partial_x (hu) & = & 0, \\ \partial_t u + u\partial_x u & = & fv - g\partial_x \eta, \\ \partial_t v + u\partial_x v & = & -fu. \end{array}\right. \tag{6.7}$$

where as previously η is the free surface, h is the water column height, $\eta = h + b$ with b the bottom height with respect to a reference height and u, v are the depth-averaged lengthwise and transverse velocity respectively. Note that although we used the term "1D",the transverse velocity v still appears in the equation.

6.2.1 1D finite volume method

We use finite volume method to numerically simulate the flow. Here we will briefly describe the fundamental aspects of finite volume method. The conserved form of 1D shallow water

equation is a prerequisite to implement the finite volume approach. The 1D system reads,

$$\frac{\partial \eta}{\partial t} + \frac{\partial (hu)}{\partial x} = 0, \tag{6.8a}$$

$$\frac{\partial (hu)}{\partial t} + \frac{\partial (hu^2 + \frac{1}{2}gh^2)}{\partial x} = -gh\frac{\partial b}{\partial x} + hfv, \tag{6.8b}$$

$$\frac{\partial (hv)}{\partial t} + \frac{\partial (huv)}{\partial x} = -hfu. \tag{6.8c}$$

Here we used the relationship

$$gh\frac{\partial \eta}{\partial x} = \frac{\partial (\frac{1}{2}gh^2)}{\partial x} + gh\frac{\partial b}{\partial x}.$$

The finite volume method is based on the control volume formulation. The space discretization is illustrated in figure 6.2. Note that here we use a cell centered scheme.

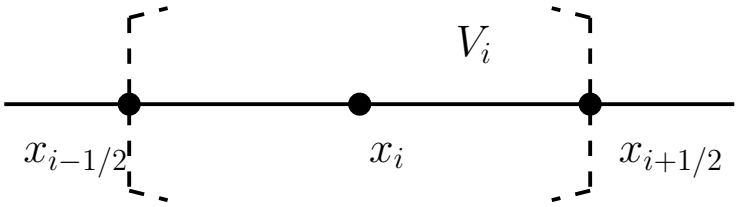


Figure 6.2 – Spatial discretization of 1D FVM scheme. V_i is the control volume with x_i at the center and $x_{i\pm1/2}$ at the two interfaces.

The system Eqs.(6.8) can be expressed in the form of a symbolic conservation law:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} = \mathbf{S}, \tag{6.9}$$

where the vector $\mathbf{U}$ contains the movement quantities $(\eta, hu, hv)^T$, the vector $\mathbf{F}$ contains the convective flux, and the vector $\mathbf{S}$ is the source term containing the Coriolis force and the topographic variation.

Before we go any further, we define the spatial average integrated value of $\bar{\mathbf{U}}$ on $\mathcal{V}_i$ at time level t^n equals to,

$$\bar{\mathbf{U}}_i^n \approx \frac{1}{\Delta x}\int_{\mathcal{V}_i} \mathbf{U}(t^n, x)dx \tag{6.10}$$

Then we deduce a corresponding spatial integral form of the conservation law by integrating Eq.(6.9) over control volume $\mathcal{V}_i$:

$$\frac{\partial}{\partial t}\int_{\mathcal{V}_i} \mathbf{U}(t,x)dx = -\int_{\mathcal{V}_i} \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x}dx + \int_{\mathcal{V}_i} \mathbf{S}dx, \tag{6.11}$$

The integration of flux gradient over control volume can be interprets as the difference of flux between two interfaces of the control volume,

$$\frac{\partial}{\partial t}\int_{\mathcal{V}_i} \mathbf{U}(t,x)dx = -\big(\mathbf{F}(\mathbf{U}(t^n, x_{i+1/2})) - \mathbf{F}(\mathbf{U}(t^n, x_{i-1/2}))\big) + \int_{\mathcal{V}_i} \mathbf{S}(t^n, x_i)dx, \tag{6.12}$$

Integrating above equation again from time level t^n to t^{n+1}, we have,

$$\begin{aligned}\int_{\mathcal{V}_i} \mathbf{U}(t^{n+1}, x_i)dx = &\int_{\mathcal{V}_i} \mathbf{U}(t^n, x_i)dx \\ &- \Big(\int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{U}(t^n, x_{i+1/2}))dt - \int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{U}(t^n, x_{i-1/2}))dt\Big) \\ &+ \int_{t^n}^{t^{n+1}} \int_{\mathcal{V}_i} \mathbf{S}(t^n, x_i)dxdt,\end{aligned} \tag{6.13}$$

We also define the temporal average integrated value over t^n to t^{n+1},

$$\bar{\mathbf{F}}(\mathbf{U}^n_{i+1/2}) \approx \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{U}(t^n, x_{i+1/2}))dt, \tag{6.14}$$

Using the spatial average (6.10) and time average (6.14), we deduce the form of the following **discrete equation**:

$$\bar{\mathbf{U}}^{n+1}_i = \bar{\mathbf{U}}^n_i - \frac{\Delta t}{\Delta x}\Big(\bar{\mathbf{F}}(\mathbf{U}^n_{i+1/2}) - \bar{\mathbf{F}}(\mathbf{U}^n_{i-1/2})\Big) + \Delta t\bar{\mathbf{S}}^n_i \tag{6.15}$$

Note that the movement quantities $\mathbf{U}$ do not exist at interfaces $i-1/2$ or $i+1/2$, so it is necessary to assume that the flux $\mathbf{F}^n_{i+1/2}$ can be calculated based on the values at cell center of $\mathbf{U}^n_i$ and $\mathbf{U}^n_{i+1}$. So the problem remains to compute the convective flux $\mathbf{F}$.

One intuitive method is to define the flux at interface as to the average of two adjacent cells.

$$\mathbf{F}^n_{i+1/2} = \frac{1}{2}\big(\mathbf{F}(\mathbf{U}^n_i) + \mathbf{F}(\mathbf{U}^n_{i+1})\big)$$

However, this form is proved to be very unstable and incapable of dealing with discontinuities at interface.

A better solution consists to rely on a Godunov-type schemes which consists in reconstructing a piecewise-polynomial interpolation. This flux interpolation is then advanced in time according to the model equation and finally projected on its cell borders. Generally, one can divid the Godunov-type schemes into two subclasses: the upwind methods and central methods. In the upwind scheme, a polynomial reconstruction is defined in every cell. This polynomial interpolation is then used to compute a new cell average at the same location for the next time step. The evolution procedure requires the determination of the self-similar exact solution $\mathbf{U}_{i+1/2}(0)$. Indeed here the convective flux is calculated by solving following intercell Riemann problem for each interface $x_{i+1/2}$:

$$\begin{cases} \partial_t\mathbf{U} + \partial_x\mathbf{F} &= 0, \\ \mathbf{U}(0,x) &= \begin{cases} \mathbf{U}_L & if \quad x < x_{i+1/2} \\ \mathbf{U}_R & if \quad x > x_{i+1/2} \end{cases} \end{cases} \tag{6.16}$$

$\mathbf{U}_L$ and $\mathbf{U}_R$ are respectively the left and right states of the Riemann problem. Their values are guessed from the average values of the neighboring cells. The method can be used if the grid is built such is there are no discontinuity within each cells. Both analytic and approximate Riemann solver can be employed. The analytic solution of such Riemann problem is detailed in the appendices. The approximate solver include the Roe solver, the HLLE solver and the HLLC solver etc. Here we adopt the Roe solver to find the expression

of the flux $\mathbf{F}_{i+1/2}$ (Roe, 1981; Ambrosi, 1995) to find the convective flux between interface. The Roe's Riemann solver finds the exact solution of the approximated Riemann problem,

$$\partial_t \mathbf{U} + \tilde{A}\partial_x \mathbf{U} = 0,$$

where $\tilde{A}$ is the jacobian matrix of flux

$$\tilde{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}},$$

The numerical flux is defied as,

$$\mathbf{F}^n_{i+1/2} = \frac{1}{2}\big(\mathbf{F}(\mathbf{U}^n_i) + \mathbf{F}(\mathbf{U}^n_{i+1}) - \hat{\mathbf{R}}|\hat{\mathbf{\Lambda}}|\hat{\mathbf{L}}(\mathbf{U}^n_i - \mathbf{U}^n_{i+1})\big) \tag{6.17}$$

where the matrix $\hat{\mathbf{R}}, \hat{\mathbf{\Lambda}}$ and $\hat{\mathbf{L}}$ correspond to the right eigenvectors, the absolute eigenvalues and the left eigenvectors of $\tilde{A}$ respectively.

Considering system (6.8), $\tilde{A}$ takes the form,

$$\tilde{A} = \begin{pmatrix} 0 & 1 & 0 \\ -u^2 + gh & 2u & 0 \\ -uv & v & u \end{pmatrix}, \tag{6.18}$$

Noting c the phase speed of wave,

$$c = \sqrt{gh},$$

the eigenvalues of $\tilde{A}$ are

$$\begin{aligned} \lambda_1 &= u - c, \\ \lambda_2 &= u, \\ \lambda_3 &= u + c, \end{aligned} \tag{6.19}$$

and the matrix $|\hat{\mathbf{\Lambda}}|, \hat{\mathbf{R}}$ and $\hat{\mathbf{L}}$ read respectively,

$$|\hat{\mathbf{\Lambda}}| = \begin{pmatrix} |\hat{u} - \hat{c}| & 0 & 0 \\ 0 & |\hat{u}| & 0 \\ 0 & 0 & |\hat{u} + \hat{c}| \end{pmatrix}, \tag{6.20}$$

$$\hat{\mathbf{R}} = \begin{pmatrix} 1 & 0 & 1 \\ \hat{u} - \hat{c} & 0 & \hat{u} + \hat{c} \\ \hat{v} & 1 & \hat{v} \end{pmatrix}, \tag{6.21}$$

$$\hat{\mathbf{L}} = \begin{pmatrix} \frac{1}{2} + \frac{\hat{u}}{2\hat{c}} & -\frac{1}{2\hat{c}} & 0 \\ 0 & 0 & -1 \\ \frac{1}{2} - \frac{\hat{u}}{2\hat{c}} & \frac{1}{2\hat{c}} & 0 \end{pmatrix}, \tag{6.22}$$

where the hat indicates Roe average of the following quantities,

$$\begin{aligned} \hat{h} &= \sqrt{h_L h_R}, \\ \hat{u} &= \frac{\sqrt{h_L}u_L + \sqrt{h_R}u_R}{\sqrt{h_L} + \sqrt{h_R}}, \\ \hat{c} &= \sqrt{\frac{gh}{2}(h_L + h_R)}. \end{aligned} \tag{6.23}$$

6.3 2D finite volume method

The derivation of a 2D finite volume scheme is similar to the 1D case, except that in a two-dimensional scenario, the domain is divided into a mesh of 2D control volumes. For sake of simplicity we used a regular cartesian mesh with a fixed step Δx for the x-axis and fixed step Δy for the y-axis, however the method can be extended to irregular grids with little efforts. The computational domain is shown in figure 6.3. The integration over each

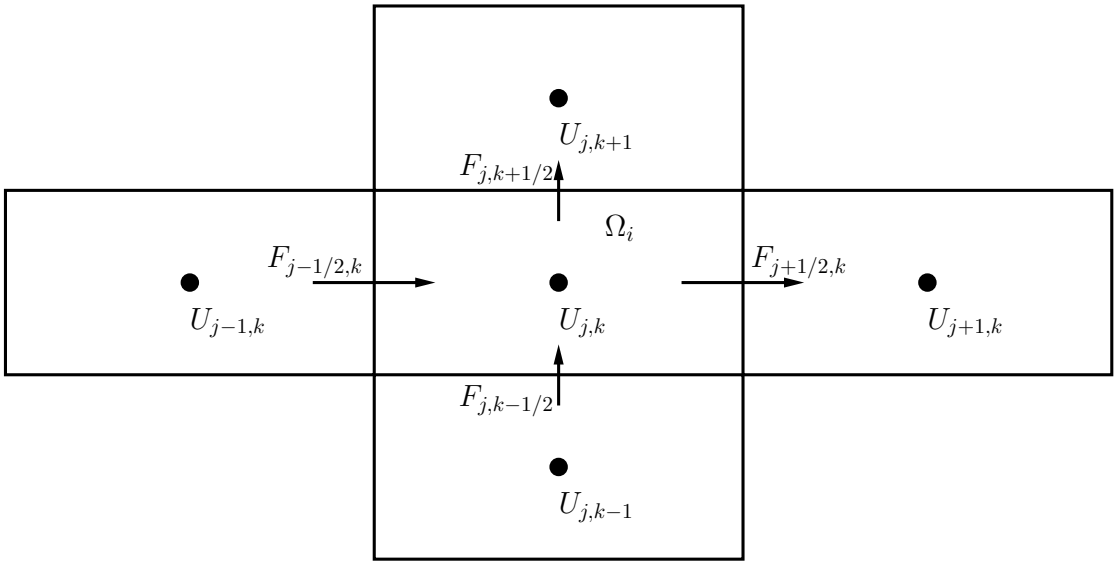


Figure 6.3 – Cartesian 2D FVM grid. Ω_i is the control volume with the quantity function $U_{j,k}$ at the center and the flux function $F_{j\pm1/2,k\pm1/2}$ at the interfaces.

control volume Ω of system (6.2) combined Green's theorem yields,

$$\frac{\partial}{\partial t}\iint_{\Omega}\mathbf{U}d\Omega+\oint_{\partial\Omega}\mathbf{F}\cdot\mathbf{n}ds=\iint_{\Omega}\mathbf{S}d\Omega+\oint_{\partial\Omega}\mathbf{F^d}\cdot\boldsymbol{n}ds$$

where $\mathbf{F}\cdot\mathbf{n}=Gn_x+Hn_y$, $\mathbf{F^d}\cdot\mathbf{n}=G^dn_x+H^dn_y$.

We also need to define the approximation of the spatial average integral value of movement quantities vector $\mathbf{U}$ on cell $\Omega_{j,k}$ at time t^n:

$$\bar{\mathbf{U}}^n_{j,k}\approx\frac{1}{\Delta x\Delta y}\iint_{\Omega_i}\mathbf{U}(t^n,\mathbf{x})d\Omega \tag{6.24}$$

and the temporal average integrated value over t^n to t^{n+1},

$$\bar{\mathbf{F}}(\mathbf{U}^n_{j+1/2,k})\approx\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}}\mathbf{F}(\mathbf{U}(t^n,\mathbf{x}_{j+1/2,k}))dt, \tag{6.25}$$

The time discretization of the temporal derivative term then yields,

$$\begin{aligned}\frac{\partial}{\partial t}\iint_{\Omega_i}\mathbf{U}d\Omega&\approx\frac{\iint_{\Omega_i}\mathbf{U}^{n+1}d\Omega-\iint_{\Omega_i}\mathbf{U}^nd\Omega}{\Delta t}\\&\approx\frac{\Delta x\Delta y}{\Delta t}(\bar{\mathbf{U}}^{n+1}_{j,k}-\bar{\mathbf{U}}^n_{j,k})\end{aligned} \tag{6.26}$$

From here on, we will drop the bars from $\mathbf{U}, \mathbf{F}$ without risk of ambiguity. The convective flux term integration becomes a sum of integrals along each cell edge i,

$$\begin{aligned}\oint_{\partial\Omega} \mathbf{F}\cdot\mathbf{n}ds &= \sum_{i=m(j,k)} \mathbf{F}_{\mathbf{i},\mathbf{j},\mathbf{k}}\cdot\mathbf{n}\Delta l_i \\ &= (\mathbf{F}_\perp\Delta y)_{j+1/2,k} - (\mathbf{F}_\perp\Delta y)_{j-1/2,k} + (\mathbf{F}_\perp\Delta x)_{j,k+1/2} - (\mathbf{F}_\perp\Delta x)_{j,k-1/2}. \quad (6.27)\end{aligned}$$

where $m(j,k)$ consists in the neighboring cells of cell (j,k), Δl_i stands for the length of the boundary between adjacent cells.

The $\mathbf{F}_\perp$ is evaluated by aforementioned Roe's flux, for example, with respect to interface $\mathbf{x}(j+1/2,k)$, the flux reads,

$$\mathbf{F}_{\perp,j+1/2,k} = \frac{1}{2}[\mathbf{F}_{\perp,j,k} + \mathbf{F}_{\perp,j+1,k} - -\hat{\mathbf{R}}|\hat{\mathbf{\Lambda}}|\hat{\mathbf{L}}(\mathbf{U}^n_{j,k} - \mathbf{U}^n_{j+1,k})]$$

The Roe solver is a very efficient upwind method which deals as well with discontinuities and smooth areas. For the 2D SWE, the analysis has been given by Priestley (1987). For this paper we used Priestley's auxiliary vector for Roe decomposition implementation.

The discretization of the diffusive flux term $\mathbf{F}^\mathbf{d}$ is done likewise.

$$\begin{aligned}\oint_{\partial\Omega} \mathbf{F}^\mathbf{d}\cdot\boldsymbol{n}ds &= \sum_{i=m(j,k)} \mathbf{F}_{\mathbf{i},\mathbf{j},\mathbf{k}}\cdot\mathbf{n}\Delta l_i \\ &= (\mathbf{F}^d_\perp\Delta y)_{j+1/2,k} - (\mathbf{F}^d_\perp\Delta y)_{j-1/2,k} + (\mathbf{F}^d_\perp\Delta x)_{j,k+1/2} - (\mathbf{F}^d_\perp\Delta x)_{j,k-1/2}\end{aligned} \quad (6.28)$$

with $\mathbf{F}^d_{\perp,j,k}$ determined by a simple centered scheme. The gradient at each cell boundary is needed to calculate this term. One may use Gauss's theorem to evaluate the gradient of cell i and calculate the gradient over boundary by a distance-weighted average of two adjacent cell center gradient values.

The treatment of the source term $\mathbf{S}$ can be done either explicitly or implicitly. An explicit discretization is simply,

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t\mathbf{S}(\mathbf{U}^n).$$

Such a scheme requires a small value of Δt to maintain the numerical stability (CFL condition). Implicit scheme are more flexible with respect to the time step value. Such a scheme is described below.

Time integration scheme

In this section, we focus on the time integration discretization corresponding to the LHS of the following equation:

$$\frac{d}{dt}\mathbf{U} = -\nabla\mathbf{F}(\mathbf{U}).$$

As the RHS of the latter equation was dealt in the previous subsection, we come to solve an ODE. We decide to use the Runge Kutta method whose general formulation is recalled

below:

$$\begin{aligned}
\mathbf{U}^{n+1} &= \mathbf{U}^n + \Delta t \sum_{i=1}^{r} b_i k_i, \\
T_i &= t^n + c_i \Delta t, \\
\mathbf{U}_i &= \mathbf{U}^n + \Delta t \sum_{j=1}^{r} a_{ij} k_j, \\
k_i &= \mathbb{M}(T_i, \mathbf{X}_i).
\end{aligned} \tag{6.29}$$

Despite that the 4th order method is the most commonly used, we decided to hold the 3rd order Runge Kutta method, given in the appendices, as it provides a good compromise between the order accuracy and the computation time. A previous test have been lead using each method and it turned out that both methods yields very similar results. As we aim to carry out data assimilation techniques which computational load depends on the direct model cost, we preferred holding the 3rd order method.

Boundary conditions

In our study, we study a fluid contained in a tank modeled as a closed rectangular shaped domain with nonporous walls. Within this context, we decide to impose reflective boundary conditions on the walls of the domain. In practice, we consider ghost cells around the domain that will share the same height value that its neighbors, and the same velocity value with a negative sign. This will constrain the force the fluid to go backwards, *i.e* inside the domain.

6.4 Twin synthetic experiments

We set up a twin experiment from two reference trajectories obtained by running the numerical model from two different initial conditions characterized by different error statistics. The background initial height field was fixed as a smooth slope tilted along the x-axis by 20%. The background initial velocity field was fixed as zero.

In the following, we refer to the first synthetic experiment as **case A**. In this case, we define the reference initial condition as a 20% tilted initial flat surface) perturbed with an homogenous Gaussian noise defined by an exponential covariance of de-correlation length (about 20% of the lengthy scale L_x) and standard deviation (5 mm for height, 1 mm/s for velocity). The noise is an isotropic Gaussian field sampled in the Fourier domain (Beyou, 2013). This 20% tilted initial waves plus the noise constitutes the prior configuration that we have on the initial condition 6.4. This first experiment is thus characterized by a trajectory whose initial state is similar to this *a priori* experimental condition up to a Gaussian random field.

In the second case, referred as **case B** hereafter, we assumed that the wave is generated by height difference. Here the initial reference height was a smooth slope tilted along the x-axis and y-axis by respectively 21% and 10%. The initial reference velocity field was fixed as a Gaussian field with a standard deviation of 1 mm/s.

The calculation domain for case A is discretized on $n_x \times n_y = 26 \times 11$ rectangular grid, and we refine the grid in case B to $n_x \times n_y = 101 \times 41$. In both cases the dynamic model time step is set to $1e^{-3}s$.

The synthetic observations were generated by adding i.i.d. Gaussian noise perturbations to the reference free surface's height and velocity fields at each grid points per 50

frames of dynamic states. The synthetic observation errors in terms of height and velocity are specified as 0.25 mm and 1 mm/s respectively. Here we adopted a single assimilation window containing 5 observations uniformly distributed in the DA window.

Note that in both cases we defined the difference of maximum and minimum surface height of initial background state as characteristic height Δh, we also define the characteristic velocity U as an approximation of wave phase velocity and the final characteristic time of the assimilation window equals to $t_f U/L_x$. For case A, $\Delta h = 20\% L_x = 50\ mm$, $U = \sqrt{g\Delta h} = 0.9007\ mm/s$, and $\frac{t_f}{L_x/U} = 0.9006$; for case B, $\Delta h = 20\% L_x = 50\ mm$, $U = \sqrt{g\Delta h L_x/\sqrt{L_x^2 + \Delta h^2}} = 0.8950\ mm/s$ and $\frac{t_f}{L_x/U} = 0.8951$. All the characteristic values are used to adimensionalize our results.

The figure 6.4 summarizes the different initial condition configurations considered for the twin experiment:

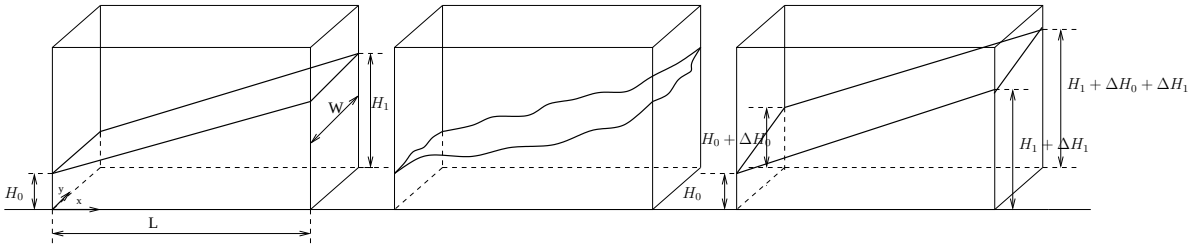


Figure 6.4 – A priori initial experimental configuration (on the left) and the true synthetic initial conditions with a Gaussian noise–Case A–(in the middle) and with a 10% slope along the y-axis–Case B– (on the right).

In order to construct the background error covariance matrix, we use different strategies with respect to the incremental 4DVar and the 4DEnVar techniques. For the incremental 4DVar, we adopt a static diagonal matrix $B = \sigma_b^2 \mathbb{I}$, where σ_b was optimally tuned as the standard deviation between the true solution and the *a priori* experimental initial state; for the 4DEnVar, as the background error covariance is derived from the ensemble fields, it is crucial that the initial ensemble represents correctly the background errors. In case A, we generated an initial ensemble perturbing the *a priori* initial state with the same homogenous Gaussian perturbations of covariance σ_b^2 as the reference. It is important to underline that within this case, although the initial error is indeed Gaussian noise, the evolution of such Gaussian noise driven by non-linear shallow water model can result in non-Gaussian random noise.

In case B, we set up two kind of perturbation of the initial condition. The first one corresponds to the Gaussian perturbation of the prior configuration as previously. However, a Gaussian error with zero mean leads to a typical biased assimilation system given our experimental setup. Dee (2005) proposed several bias detection strategies to deal with such problems. We adopt a simply idea of modeling the initial error as:

$$\boldsymbol{\eta} = \mathbf{b} + \tilde{\boldsymbol{\eta}} \quad \text{with} \quad \langle \tilde{\boldsymbol{\eta}} = 0 \rangle, \tag{6.30}$$

where the bias vector b is fixed in time and is estimated through the DA system as the initial condition. The algorithm described in chapter 5 remains essentially the same. The second case implements a parameter perturbation strategy, consisting in defining the initial members from a random drawing of different slopes. On the different figures, this

approach is indicated by the "PP" suffix whereas the absence of suffix denotes a Gaussian perturbation.

Let us note that the structure of the **B** matrix with regard to standard 4DVar will be restricted in a diagonal form in order to reduce the rather high computational cost.

We also highlighted the method of Liu et al. (2008) (indicated by "Liu-et-al" suffix on figures) using the perturbed observation, a single outer loop and localized covariance. The strategies with several outer loops, perturbed observation or direct transformation are indicated by suffix "OL", "OP" and "DT" respectively. Covariance localization and local ensemble are indicated by suffix "LC", "LE" accordingly. Since the covariance localization is always associated with the perturbed observation approach, 'OP' is omitted for the sake of simplicity. The same applies to 'LE'. We point out that the initial state (perturbed surface height and null velocity field), was integrated for a few time steps before we started the assimilation process. This provided us the guaranty of balanced velocity perturbations that complied well with the nonlinear dynamic model.

6.4.1 Comparison tools

RMSE analysis The Root Mean Square Error(RMSE) is a way of measuring of the differences between values predicted by a model or an estimator and the values actually observed. The RMS between a predicted state $\mathbf{x}^f$ and the observed state $\mathbf{x}^{obs}$ is defined by:

$$\text{RMSE} = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(\mathbf{x}^f - \mathbf{x}^{obs})^2}, \tag{6.31}$$

When we deal with the synthetic data, we will base our analysis on the comparison of the background, observation and assimilated states' RMSE with respect to the true solution. Furthermore, the observations time step is many times bigger than the numerical time step, thus we will be able to assess the results on a very small amount of observations. We will also consider the RMSE comparison on a semilogarithmic graph.

For case A, we will study the assimilation considering only velocity observations. For case B, which is harder, we will study the assimilation considering only height observations, velocity observations and finally both height and velocity observations.

6.4.2 Results on case A (20% slope on x-axis with additive Gaussian perturbation on the initial surface height)

In this case, we compared the 4DVar and 4DEnVar on the basis of a partially observed system in which only the velocity measurements are available. As explained above, the initial background matrices were constructed on a similar basis. They both correspond to a Gaussian distribution of covariance $\sigma_b^2\mathbb{I}$, σ_b, corresponding to the true deviation between the initial solution of the reference and the given *a priori* background. The background error is however biased as its mean is unknown.

The tests were carried out for a coarse resolution to assess certain parameter sensitivities used in 4DEnVar. The Gaussian random noise fields (Beyou et al., 2013b) used to construct the reference and initial ensemble hinges on the standard deviation and the de-correlation length. Therefore we designed two slightly different tests. One test uses the ensemble initialized by the same standard deviation and de-correlation length as the

reference; the other used the ensemble initialized by slightly different standard deviation and de-correlation length. The former is called 'perfect background ensemble' while the latter is called 'imperfect background ensemble'.

Under the circumstance of perfect background ensemble, figure 6.5a pictures for both assimilation techniques the root mean square error (RMSE) curves of the reconstructed free surface curves and its associated velocity component. Note that no localization is used here. We observe that the RMSE curve of 4DEnVar methods are generally flatter than the 4DVar curves. Similar results (even better on the unobserved height surface) are achieved when then sampling number is big enough ($N = 32$ in this case). The accuracy of the analysis is directly dependent on the ensemble size, the relationship is roughly that doubling the ensemble size reduces half the error. At this coarse scale, an ensemble of only few samples allows a good representation of the background error statistics without any localization.

Here we focus on comparing the ensemble update scheme (denoted by suffix combination of 'OP-OL' and 'DT-OL' respectively). Note that the update scheme only makes sense when extra outer loop is employed (usually 2). The results clearly are slightly in favor of a direct transformation strategy. The gap between two ensemble update scheme is even larger when the ensemble dimension is larger.

Another important issue is that increasing ensemble size has its limit, in this case, employing more than 32 members will not bring great improvements to the analysis considering the associated computational cost. This is shown by the 64 members ensemble case in table 6.1. In this table, I listed both the final RMSE and the temporal mean RMSE. The two types of RMSE roughly corresponds to two distinctive objects of data assimilation. Generally speaking, a lower final RMSE indicates better performance in terms of forecast ability; while a lower temporal mean RMSE indicated better performance in terms of the reconstruction ability. A flatter curve can also be revealed by the relative ratio between the two RMSE.

Type	N	OL Iter	IL Iter	RMSE(t_f)e^{-4}	RMSE($\bar{t}$)
Observation	-	-	-	-	$(9.758, 9.804)$
Background	-	-	-	$(275.5, 92.32)$	$(362.5, 115.2)$
4DVar	-	3	100	$(1.450, 0.8495)$	$(6.645, 5.693)$
4DEnVar-OP	8	1	20	$(5.815, 12.47)$	$(12.85, 20.63)$
4DEnVar-OP	16	1	40	$(2.859, 2.424)$	$(6.091, 7.342)$
4DEnVar-OP	16	2	40,20	$(2.637, 2.222)$	$(5.574, 7.184)$
4DEnVar-OP	32	1	100	$(1.830, 0.7037)$	$(3.952, 1.496)$
4DEnVar-OP	32	2	100,20	$(0.9707, 0.5378)$	$(2.404, 1.076)$
4DEnVar-OP	64	1	100	$(2.325, 0.6573)$	$(4.320, 1.635)$
4DEnVar-OP	64	2	100,20	$(0.7480, 0.1951)$	$(1.759, 0.7443)$
4DEnVar-DT	8	1	20	$(5.815, 12.35)$	$(12.32, 20.38)$
4DEnVar-DT	16	1	40	$(2.805, 2.449)$	$(6.091, 7.247)$
4DEnVar-DT	16	2	40,30	$(2.138, 1.821)$	$(5.173, 5.965)$
4DEnVar-DT	32	1	100	$(1.888, 0.435)$	$(4.018, 1.157)$
4DEnVar-DT	32	2	100,50	$(0.9531, 0.4809)$	$(1.757, 1.068)$

Table 6.1 – RMSE comparison table. Type: Group of methods with perfect background ensemble and no localization (OP, observation perturbation; DT: direct transformation); N: ensemble members; OL Iter: Outer loop iteration; IL Iter: Inner loop iterations; RMSE(t_f): final RMSE; RMSE($\bar{t}$): mean RMSE.

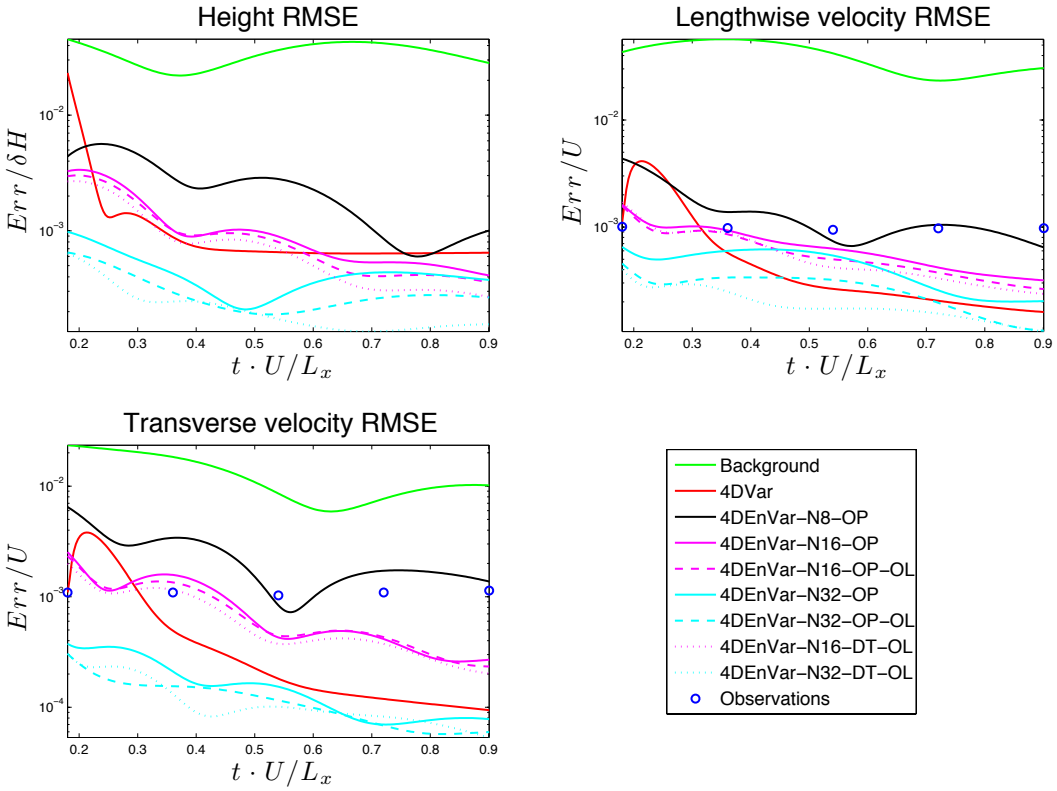
The aspects of localization is summarized in figure 6.5b and table 6.2. We notice that the localization is not so effective here: the curves denoting the localization covariance approach ('N8-LC' with black line and 'N16-LC' with magenta line in figure 6.5b) is obviously higher than the curves denoting no localization ('N8-OP' with black line and 'N16-OP' with magenta line in figure 6.5a). Since the background error statistic can be perfectly represented by the initial ensemble, the localization therefore is unnecessary and even detrimental to the error covariance as actual long range correlation are discarded. Despite its negative effect, we see that the analysis obtained by localized approach with extra outer loop can still be comparable to 4DVar (Note the curves of 'N16-OL-LC' with magenta dashed line and 'N16-OL-LE' with dash-dot magenta line in figure 6.5b and the reduced value of RMSE when extra outer loop is used in table 6.2). This could be explained by another effect introduced by the localization techniques: enhanced analysis increment subspace. This goal is achieved differently for the two types of localization techniques (ie. Schur product, or local ensemble approach): the localized covariance approach introduces an enhanced control vector; the local ensemble approach, on the other hand, keeps a control vector of the same size as the ensemble members but reduces the analysis vector dimension.

Another interesting conclusion could be drawn here is that the localized covariance approach actually benefit more from the extra outer loop. In table 6.2, we see that the RMSE in function of the column 'OL Iter': after a single outer loop of the approach of local ensemble suffixed by 'DT-LE' is much lower than the approach of localized covariance suffixed by 'OP-LC', however after an extra outer loop, the two approach yield comparable results. We also note that the increased sampling number can tolerate bigger length of cut-off distance. This can be implied from the RMSE versus the column 'COD/L_x'. The RMSE values of $40\%L_x$ cut-off distance is significantly better that the values of $20\%L_x$

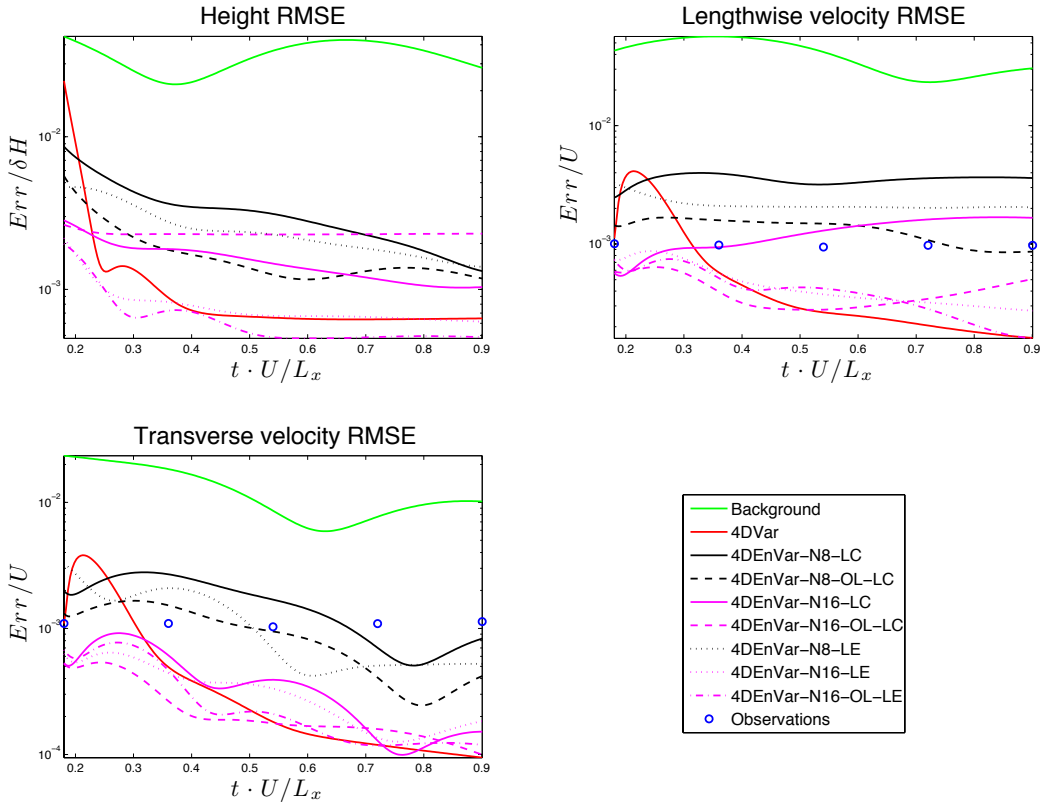
Type	N	COD/L_x	OL Iter	IL Iter	RMSE(t_f)e^{-4}	RMSE($\bar{t}$)
Observation	-	-	-	-	-	$(9.758, 9.804)$
Background	-	-	-	-	$(275.5, 92.32)$	$(362.5, 115.2)$
4DVar	-	-	3	100	$(1.450, 0.8495)$	$(6.645, 5.693)$
4DEnVar-OP-LC	8	20%	1	100	$(32.51, 7.485)$	$(31.78, 14.59)$
4DEnVar-OP-LC	8	20%	2	100,100	$(7.762, 3.774)$	$(12.08, 8.374)$
4DEnVar-OP-LC	16	20%	1	100	$(31.49, 6.480)$	$(29.93, 12.21)$
4DEnVar-OP-LC	16	20%	2	100,100	$(9.097, 2.147)$	$(7.475, 3.605)$
4DEnVar-OP-LC	16	40%	1	100	$(18.34, 2.783)$	$(16.39, 5.265)$
4DEnVar-OP-LC	16	40%	2	100,100	$(4.554, 0.896)$	$(3.581, 2.141)$
4DEnVar-OP-LC	32	60%	1	100	$(10.34, 0.981)$	$(8.967, 3.143)$
4DEnVar-OP-LC	32	60%	2	100,100	$(2.109, 0.442)$	$(1.987, 0.890)$
4DEnVar-DT-LE	8	20%	1	20	$(18.39, 4.715)$	$(19.43, 10.34)$
4DEnVar-DT-LE	16	20%	1	50	$(2.655, 1.656)$	$(4.156, 2.981)$
4DEnVar-DT-LE	16	20%	2	50,50	$(1.437, 1.071)$	$(3.734, 2.685)$
4DEnVar-DT-LE	32	40%	1	100	$(2.675, 0.3248)$	$(4.8118, 0.9550)$
4DEnVar-DT-LE	32	40%	2	100	$(1.525, 0.5949)$	$(2.057, 1.371)$

Table 6.2 – RMSE comparison table. Type: Group of methods with perfect background ensemble and localization (OP, observation perturbation; DT: direct transformation; LC, Localized covariance; LE, Local ensemble); N: ensemble members; COD/L_x: ratio of cut-off distance divided by characteristic length; OL Iter: Outer loop iteration; IL Iter: Inner loop iterations; RMSE(t_f): final RMSE; RMSE($\bar{t}$): mean RMSE.

cut-off distance with the 16 ensemble members suffixed by 'OP-LC'. A similar conclusion made to the EnKF system can be found in Houtekamer and Mitchell (2001, 2005).



(a)



(b)

Figure 6.5 – RMSE comparison between various variational methods: (a) group of methods with perfect background ensemble and no localization (OP, observation perturbation; DT: direct transformation; OL, extra outer loop), (b) group of methods with perfect background ensemble and localization (LC, Localized covariance; LE, Local ensemble)

The sensitivity of analysis subject to the initial ensemble error statistics is tested with imperfect background ensemble. Here I construct the initial members with a 30% length scale L_x as the de-correlation length (with same σ_b). The results with regard to the unlocalized strategies and the localized strategies are shown in tables 6.3 and 6.4 respectively. The RMSE curves are omitted since they look similar to the previous tests.

We could conclude from the table 6.3 that if no localization is considered, the ensemble-based variational methods with or without the ensemble update are incapable of yielding comparable result to 4DVar. Indeed without the use of a localization filter, the 4DVar with a static background provides clearly better results. However, if localization is well implemented, both assimilation techniques yield similar quality results in terms of the surface height or velocity components' RMSE (table 6.4). This is because unlike the previous perfect initial background ensemble test, the background ensemble is initialized by a longer de-correlation length which makes the spurious correlation accountable. Those spurious correlation needs to be eliminated by localization process. Compared to the unlocalized approach, the localization improves clearly the ensemble performances for a small ensemble size as already stated in several studies (Houtekamer and Mitchell, 2001; Fairbairn et al., 2013).

Comparing the two localization techniques, the localized covariance strategy (RMSE values with suffix 'LC' in table 6.4) outperforms the local ensemble strategy (RMSE values with suffix 'LE' in table 6.4) in this context. This could be explained from various perspectives. Although the error covariance can be well corrected in both approaches, the two techniques still differ in the way of formulating the analysis: the global analysis obtained by 'LC' approach can retain a more balanced multivariable relationship than the local analysis obtained by the 'LE' approach. Besides, the 'LC' approach can be conducted with larger cut-off distance, so the magnitude of the error covariance reduces less than the 'LE' approach in which increasing the local space size can bring out identical problems encountered by the limited number of ensemble members.

The effectiveness of introducing several outer loops in 4DEnVar again is proved in tables 6.3 and 6.4. We can witness that the RMSE values associated with 2 outer loops are much lower than those of only one outer loop. In fact, the analysis associate with only one outer loop diverges completely from the analysis obtained by the 4DVar method. However, it is hard to tell how much the analysis can benefit from the ensemble field update scheme proposed in the previous chapter. Figure 6.6 compares the performance of two nested loops algorithms: the algorithm 6 employing ensemble update and the algorithm 3 indicated by suffix 'Courtier' used in Courtier et al. (1994); Haben (2011). We observe that both algorithms can reduce well the RMSE in terms of the observed velocity components since the iterative analysis is estimated primarily based on the gap between the previous analysis trajectory and the observation. Our proposed algorithm 6 yield slightly better result regarding to lengthwise velocity and leads to great improvement regarding to the unobserved height component. This could be attributed to the background ensemble repropagation introduced in our algorithm. It can be concluded that the ensemble repropagation can indeed help the iterative analysis in redefining the subspace to which the iterative analysis increment belongs. In addition to that, the repropagation of ensemble re-ensures that the analysis increment of different variables are in balanced state, therefore alleviate the problem of balance loss introduced by the localization technique.

Type	N	OL Iter	IL Iter	RMSE(t_f)e^{-4}	RMSE($\bar{t}$)
Observation	-	-	-	-	$(9.758, 9.804)$
Background	-	-	-	$(275.5, 92.32)$	$(362.5, 115.2)$
4DVar	-	3	100	$(1.450, 0.8495)$	$(6.645, 5.693)$
4DEnVar-OP	8	1	100	$(38.10, 6.307)$	$(56.15, 10.07)$
4DEnVar-DT	8	2	60,20	$(26.09, 5.844)$	$(37.45, 12.83)$
4DEnVar-OP	16	1	100	$(37.67, 6.841)$	$(56.09, 11.52)$
4DEnVar-OP	16	2	100,20	$(12.65, 4.724)$	$(22.63, 12.23)$
4DEnVar-DT	16	2	100,20	$(16.92, 5.781)$	$(23.11, 12.10)$
4DEnVar-OP	32	1	100,20	$(37.20, 7.165)$	$(54.97, 12.88)$
4DEnVar-OP	32	2	100,20	$(12.10, 4.054)$	$(19.34, 9.599)$
4DEnVar-DT	32	2	100,20	$(12.58, 3.642)$	$(20.73, 9.026)$

Table 6.3 – RMSE comparison table. Type: group of methods with imperfect background ensemble and no localization (OP, observation perturbation; DT: direct transformation); N: ensemble members; OL Iter: Outer loop iteration; IL Iter: Inner loop iterations; RMSE(t_f): final RMSE; RMSE($\bar{t}$): mean RMSE.

Type	N	COD/L_x	OL Iter	IL Iter	RMSE(t_f)e^{-4}	RMSE($\bar{t}$)
Observation	-	-	-	-	-	$(9.758, 9.804)$
Background	-	-	-	-	$(275.5, 92.32)$	$(362.5, 115.2)$
4DVar	-	-	3	100	$(1.450, 0.8495)$	$(6.645, 5.693)$
4DEnVar-OP-LC	8	20%	1	100	$(17.94, 4.878)$	$(27.73, 9.995)$
4DEnVar-OP-LC	8	20%	2	100,40	$(6.472, 2.556)$	$(13.56, 7.343)$
4DEnVar-OP-LC	16	40%	1	100	$(13.19, 4.767)$	$(20.64, 9.074)$
4DEnVar-OP-LC	16	40%	2	100,50	$(4.523, 1.549)$	$(8.334, 4.953)$
4DEnVar-OP-LC	32	60%	1	100	$(10.78, 4.231)$	$(15.32, 7.855)$
4DEnVar-OP-LC	32	60%	2	100,100	$(2.768, 0.634)$	$(5.779, 3.992)$
4DEnVar-DT-LE	8	20%	1	50	$(22.13, 4.242)$	$(34.29, 8.982)$
4DEnVar-DT-LE	8	20%	2	50,50	$(9.581, 3.879)$	$(20.42, 8.895)$
4DEnVar-DT-LE	16	30%	1	100	$(21.02, 3.091)$	$(30.45, 7.610)$
4DEnVar-DT-LE	16	30%	2	100,100	$(7.003, 3.307)$	$(14.69, 6.528)$
4DEnVar-DT-LE	32	30%	1	100	$(18.90, 2.182)$	$(27.69, 5.623)$
4DEnVar-DT-LE	32	30%	2	100,100	$(4.268, 1.664)$	$(12.84, 4.268)$

Table 6.4 – RMSE comparison table. Type: Group of methods with imperfect background ensemble and no localization (OP, observation perturbation; DT: direct transformation; LC, Localized covariance; LE, Local ensemble); N: ensemble members; COD/L_x: ratio of cut-off distance divided by characteristic length; OL Iter: Outer loop iteration; IL Iter: Inner loop iterations; RMSE(t_f): final RMSE; RMSE($\bar{t}$): mean RMSE.

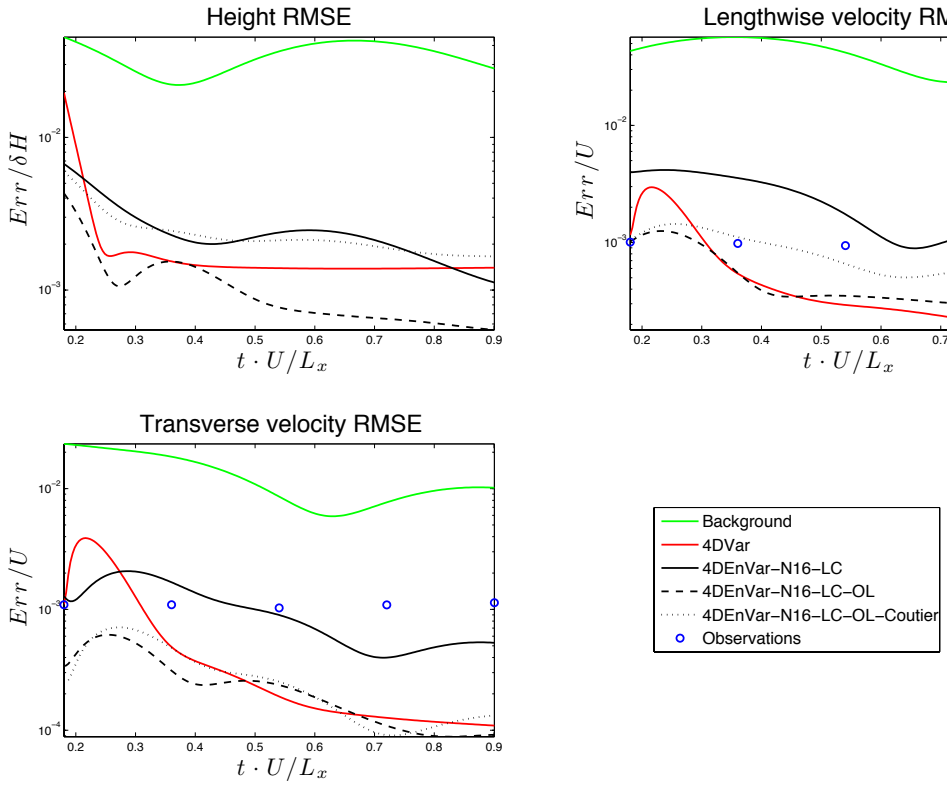


Figure 6.6 – RMSE comparison between outer loop schemes: No outer loop (black line), Algorithm 6 (black dashed line), Algorithm 3 (black dotted line).

6.4.3 Results on Case B: (21% slope on x-axis and 10% slope on y-axis)

For the case B, the assimilation techniques were evaluated with partially observed systems where only the free surface height or the velocity components are observable, and also a fully observed case in which measurements for the whole system are available. The results corresponding to each case are reported in the subsequent sections.

Height observations

Let us recall that in this case, the initial condition of the reference is far apart from the known *a priori* configuration fixed as the background – without slope in the y coordinate. As explained previously we tested two different initial background covariance setups. An ensemble of 8 members is sufficient to yield comparable results with 4DVar.

The RMSE curves corresponding to this case are gathered in figure 6.7. We observe that the 4DEnVar assimilation technique globally leads to better results than the standard 4DVar assimilation technique. The initial free surface is strongly corrected in both methods, however the unobserved velocity components are well corrected only for the ensemble technique.

Note that the mechanisms that come into play for the reconstruction of unobserved components are different in both cases. For standard 4DVar, the adjoint operator of the dynamic model (tangent and adjoint operators for incremental case) propagates a balanced increment from the height to the velocities; for 4DEnVar, the correlation terms in the background error covariance, derived from the initial ensemble, play an important role in

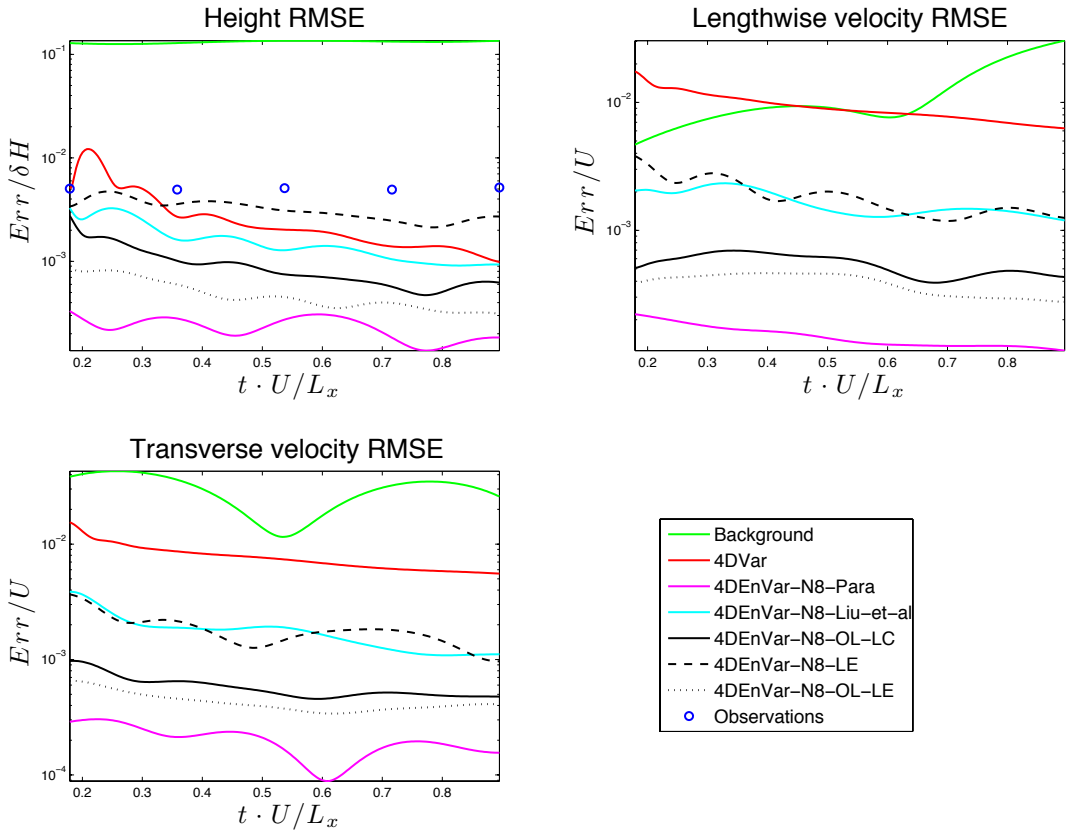


Figure 6.7 – RMSE comparison between an incremental 4DVar and 4DEnVar assimilation approaches: partial observed through noisy free surface height

reconstructing the analysis velocity fields.

The ensemble technique was the most efficient when the initial noise complied with the physics of the observed phenomenon (i.e. random slopes drawing versus Gaussian noise). For non-physical initial noise, the localization is mandatory and the cutoff distance was set as the optimal cutoff distance minimizing the average RMSE (the same value was used to define the size of local space). This distance increases as the ensemble member increases. Both results are consistent with Houtekamer and Mitchell (2001); Fairbairn et al. (2013).

We can also observe from the RMSE curves that several outer loops clearly improves the results (OL-LC against Liu-et-al, OL-LE against LE), which highlights the pertinence of the background covariance update. The method with local ensemble yielded slightly better results than the method with localized covariance.

Few general remarks can be done here. The 4DVar method used here relies on a simple diagonal covariance matrix. Its results could be improved with a better description of the background error covariance matrix. However, extra computational cost in order to handle this full matrix can not be overlooked.

Velocity observations

The results corresponding to this case are shown in figure 6.8. In this case, we retrieved almost the same behavior as in the previous case. When associated to an unphysical background covariance built from a Gaussian perturbation, the 4DEnVar techniques require to

rise the size of the ensemble in order to obtain results of comparable quality to the standard 4DVar. A localization filter is needed in that case, and a too low number of members leads to weaker results. Compared to ensemble techniques, the incremental 4DVar technique shows some difficulty to correct efficiently the unobserved surface height component. The 4DEnVar, with an initial ensemble built from random slope parameters, leads to better results with a moderate number of members ($N = 16$ in this case). A lower number ($N = 8$) yields good results for the observed variables but not for the unobserved surface height. An increase of the ensemble size ($N = 32$) slightly improves the quality of the results.

This good behavior can be observed even for a small number of members. This result highlights the gain of performance that can be achieved by fixing an adequate physical perturbation in the ensemble assimilation. For a physically adapted perturbation, the performance is improved for all the components, whether they are observed or not; the ensemble size can also be significantly reduced and no localization appears to be necessary

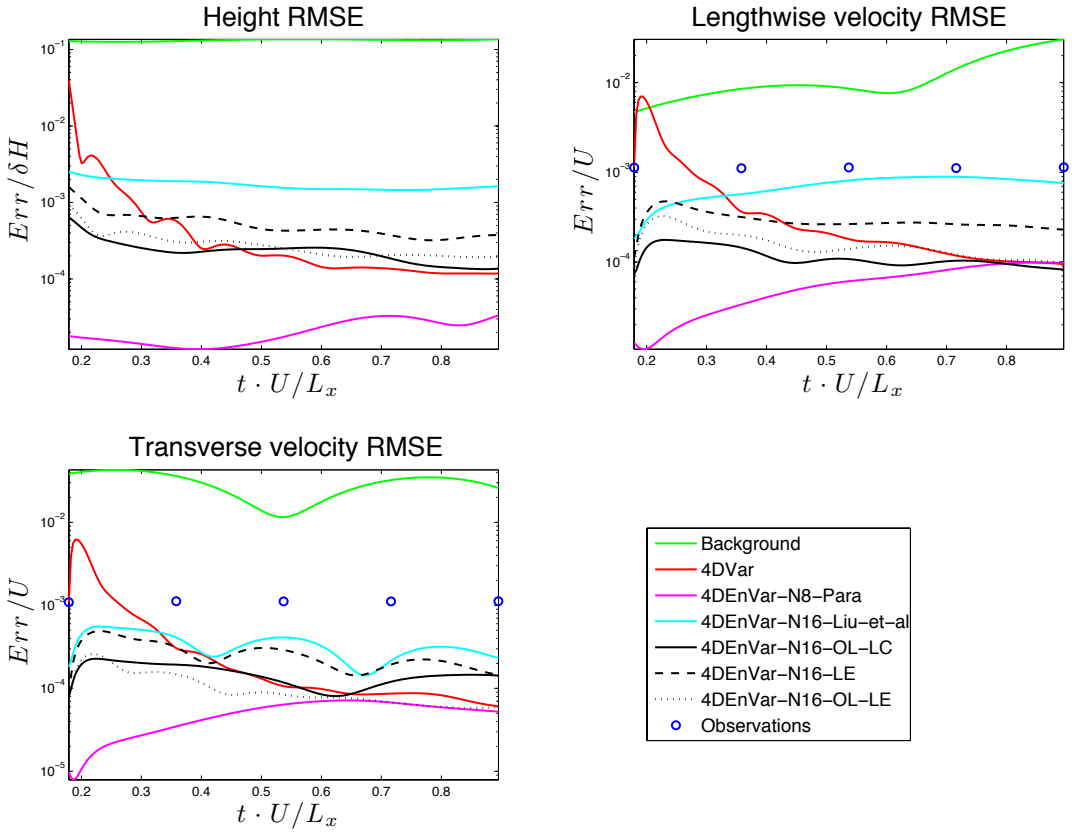


Figure 6.8 – RMSE comparison between an incremental 4DVar and 4DEnVar assimilation approaches: partial observed through noisy velocity field.

Height and velocity observations

Finally, we compared the assimilation techniques assuming a fully observable system where measurements of both height and velocity fields are provided. Similarly to the previous tests, the ensemble technique with the background covariance obtained from a Gaussian perturbation requires the augmentation of the ensemble size to obtain comparable results to those obtained with the standard 4DVar. A smaller ensemble leads to worse results and

Type	CPU Time	Memory demands
4DVar	3200s	Small
4DEnVar-PP(No Localization, N=8)	120s	Small
4DEnVar-LC(N=32)	2400s	Huge
4DEnVar-LE(N=32)	600s	Small

Table 6.5 – Comparison of the CPU time (seconds) (2 × 2.66 GHz Quad-Core Intel Xeon) and memory demands (16 GB in total) with 10^5 level of state size between different methods.

the use of a localization filter is necessary. The results obtained for the physical background covariance are slightly better than the results obtained for 4DVar.

Nevertheless, the advantage of ensemble methods are less enhanced than in the previous cases. This is probably due to the fact that under this circumstance, each variable component is mainly corrected by the corresponding observations rather than indirectly inferred from other observed components. However, in real world applications, assimilation problem are rarely fully observable, hence the capacity of ensemble methods to correct efficiently the unobserved system's components appear to be very attractive.

In terms of the two localization schemes, with a low number of samples, 'LE' approach gave better results than 'LC' approach. When increasing the sample numbers, the difference between the two localization approaches decreases. This is because the local space used in 'LE' method was already rather small compare to the full state space, so the bad effects induced by a small ensemble number was less severe in this case.

The computational time of the different methods is indicated in table 6.5. The parameter perturbation approach performs the best in terms of the CPU time and the memory demanding, this is quite natural as no localization is applied. But to devise a parameter perturbation that fit to the initial error statistics is not a trivial task regarding to real observations. Between the two localization techniques, the LE approach only used one fourth of the computational time of LC approach but its limitations are also obvious: we must restrict the local space rather small as increasing the local space leads both to increase in quadratic way the computation cost and to a bad localization. The LC approach, on the other hand, maintains a nearly constant computational cost in function of the cut-off distance. It is nevertheless associated with rather high CPU time and enormous memory demand.

On the contrary, if we double the ensemble members, the LC method takes 4 times more computation time to respond while using moderate localization; in LE method, the computational time only doubles. The different consequences of the de-correlation distance and the ensemble member resulted in the computational time may be an important factor on how to choose the optimal value regarding to different approach.

6.5 Summary

We have conducted here a complete assessment of the several assimilation schemes proposed in chapter 5. Various strategies related to the ensemble-based DA method are discussed in the texts: the observation perturbation versus the direct transformation, the localized covariance versus the local ensemble, the parameter sampling versus the Gaussian sampling, the effect of introducing extra outer loop, etc. It is nevertheless difficult to give the best

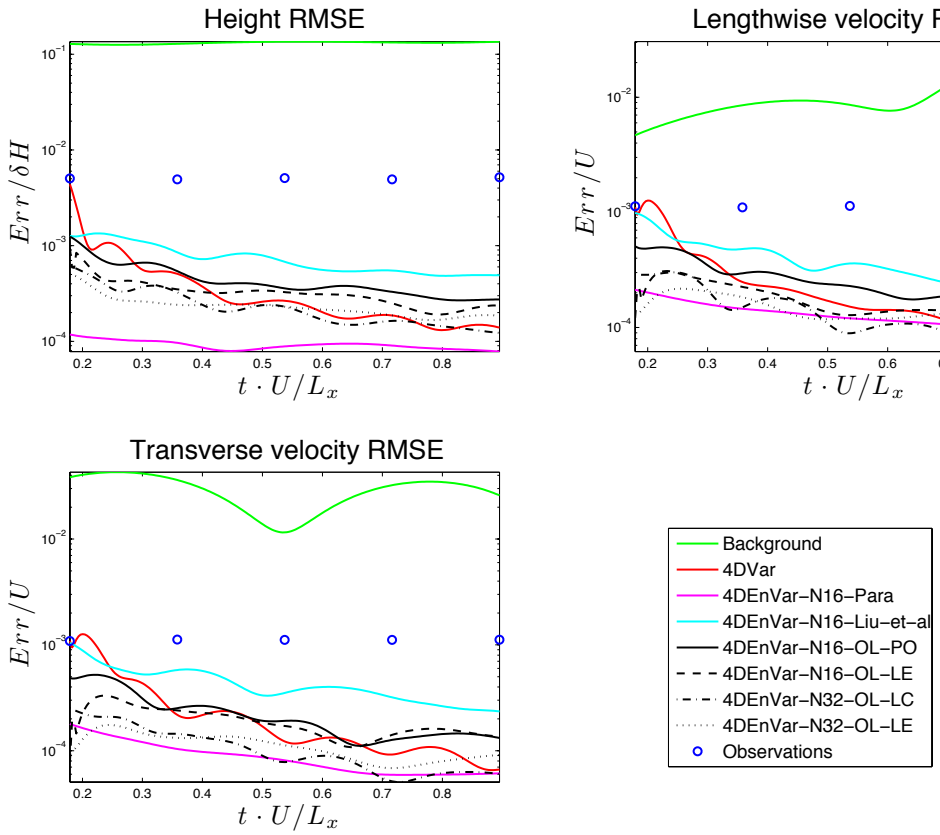


Figure 6.9 – RMSE comparison between an incremental 4DVar and 4DEnVar assimilation approaches, fully observed system (i.e. free surface height and velocity fields).

approach unless we have a profound knowledge of both the dynamical model and the observation connected to the DA. We will apply this method to real image data in next chapter.

Chapter 7

Application With Image Data

7.1 Image data from depth camera Kinect

We carried out another evaluation on a real world experiment in which the free surface of a fluid contained in a rectangular flat bottom tank of size $L_x \times L_y = 250$ mm $\times$ 100 mm was observed. Following the pioneering work of Combes et al. (2011, 2015) we used the Kinect sensor to observe the evolution of a unidirectional wave generated by an initial free surface height difference $\Delta h = 1$ cm on a grid of 222×88 pixels. The flow surface was located between 680 and 780 mm from the device. When the attenuation coefficient of the liquid is larger than 113 m^{-1}, the Kinect sensor displays a mean measurement error of 0.5 mm with standard deviations of about 0.5 mm for both flat and sinus-like surfaces. The sensor captures successfully sinus-like varying elevations with spatial periods smaller than 20 mm and amplitudes smaller than 2 mm. In the following the characteristic velocity U is considered as an approximation of wave phase velocity $\sqrt{g\Delta h}$.

Figure 7.1 – Experimental set with the Kinect sensor.

7.1.1 Data processing

These observations are characterized by a high level of noise and exhibit large regions of missing data along the boarders due to light reflections on the tank's wall (see figure 7.2a). We cope with these incomplete observations through the following these steps:

- The singular points are considered as bad points and are eliminated.

- The missing boundary areas located besides the long borders are filled with observations averaged by all the pixels' values in the same section.
- The missing boundary areas located besides the short borders are filled with observations extrapolated from the height profile.
- The missing inner hole areas are filled with pseudo-values which are computed as the mean of all the adjacent pixels' observation values.

Finally we obtain a sequence of height data on a 248×98 grid. However, owning to the computational limitations, we are inclined to run our model on a 124×49 grid. The observations are therefore interpolated to the coarse grid. Such interpolation is identical to consider a mask observation operator.

In terms of the observation errors: for a point in the unobserved region, we set the observation error as a function of the distance from the closest observed point. Thus, the longer the distance, the larger the error. The observation error is however bounded by a maximal value 60% of the height difference Δh. Within the observed region, we set the observation error homogeneously to the instrument error $\sigma_o = 5\%\Delta h$. Note that Δh is a characteristic height defined as the mean wave amplitude over time. The other characteristic quantities are calculated in the same way as in the synthetic cases. Likewise we define the characteristic velocity U as an approximation of the wave phase velocity and the final characteristic time of the assimilation window equals to $t_f U/L_x$. In this case, $\Delta h = 12\ mm$, $U = \sqrt{g\Delta h} = 0.5425\ mm/s$, and $\frac{t_f}{L_x/U} = 0.4608$. All the characteristic values are used to adimensionalize our results.

7.1.2 Dynamical model

The flow featured by this experimental configuration can be approximately described by a 2D shallow water model as well.

7.1.3 Assimilation scheme configuration

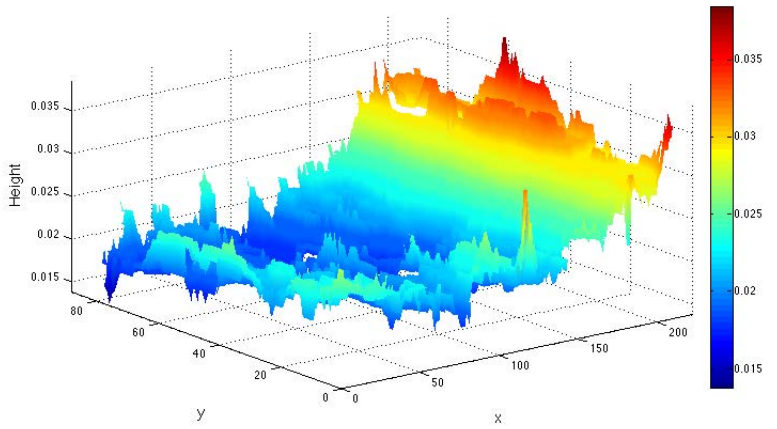
In this case, the initial background was completely unknown hence, it was set to a filtered observation with interpolated values on the missing data regions at the initial time on a 248×98 grid. This high resolution is interpolated to a 124×49 grid. Figure 7.2b shows the background state reconstructed from the first observation.

We noticed that the observed free surface behaved roughly as an unidirectional wave along the x-axis. Thus, we set the initial velocity field as a smooth linear slope where the velocity at the top of the wave was set as 23% of the wave velocity and the velocity at the bottom of the wave was set to 0 (see figure 7.2c).

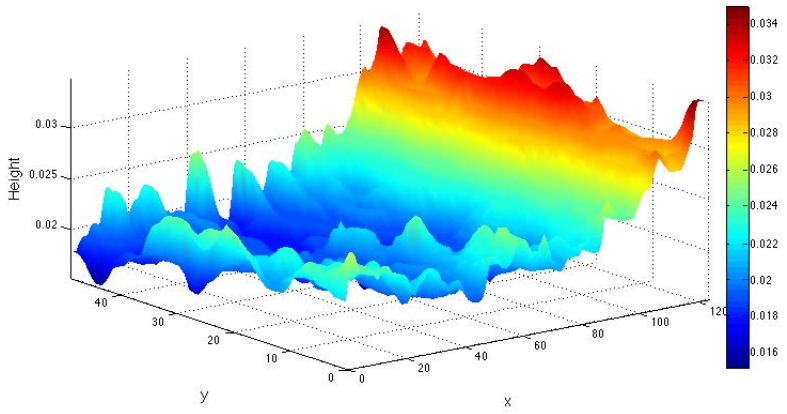
The observation time interval needs to be carefully tuned to ensure the synchronization of the observation sequence and the evolution of the background state. We set $\Delta t_{obs} = 30\Delta t_{dyn}$.

Similarly to the synthetic case, the assimilation started at the second image in order to construct balanced ensemble through the integration. The initial ensemble is defined by adding isotropic Gaussian perturbation fields with standard deviation $\sigma_b = 3.6\%\Delta h$ to the background state. The cutoff distance and the size of local space are fixed as 15% of the length L_x.

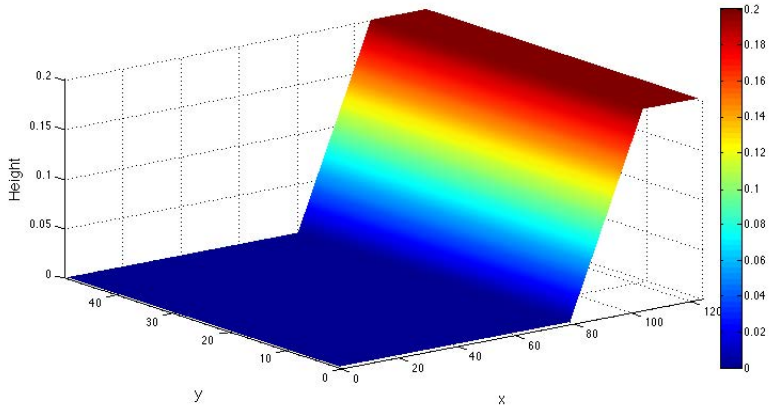
The assimilation scheme was adapted to sliding assimilation windows to avoid long range temporal correlations. Each window contains 5 observations. We adopted 5 windows



(a)



(b)



(c)

Figure 7.2 – (a) The height field observed from the kinect camera, (b) The corresponding height background state and (c) primary velocity magnitude at $t = 0$.

over 9 observation times, each window consists in 5 observations and each window starts

at the 1st, 2nd, ... , 5th observation respectively.

7.1.4 Results and discussion

When we deal with real observations, the ground truth will not be available. We can only compare the background and the assimilated states' RMSE with respect to the observations. The true solution is assumed to lie within an interval around the observation. But no precise comparison can be found. Thus instead of a RMSE study, we focus here on the direct comparison of the height field.

The results obtained by both assimilation techniques are displayed in figure 7.3 in terms of the average surface height of the wave crest as a function of time. This is quite intuitive as we were dealing here with a single wave simulation. We thus particularly focused on in the wave crest's region rather than on the other flat regions. We observe from these results that the 4DVar and the En4DVar can both follow the observation trajectory tendency. While the 4DVar tends to underestimate the surface height at the beginning of the assimilation window, the group of En4DVar yielded very similar results between the 1st and 4th image. After the fifth image, the result of En4DVar by Liu *et al.* diverges from the observation trajectory.

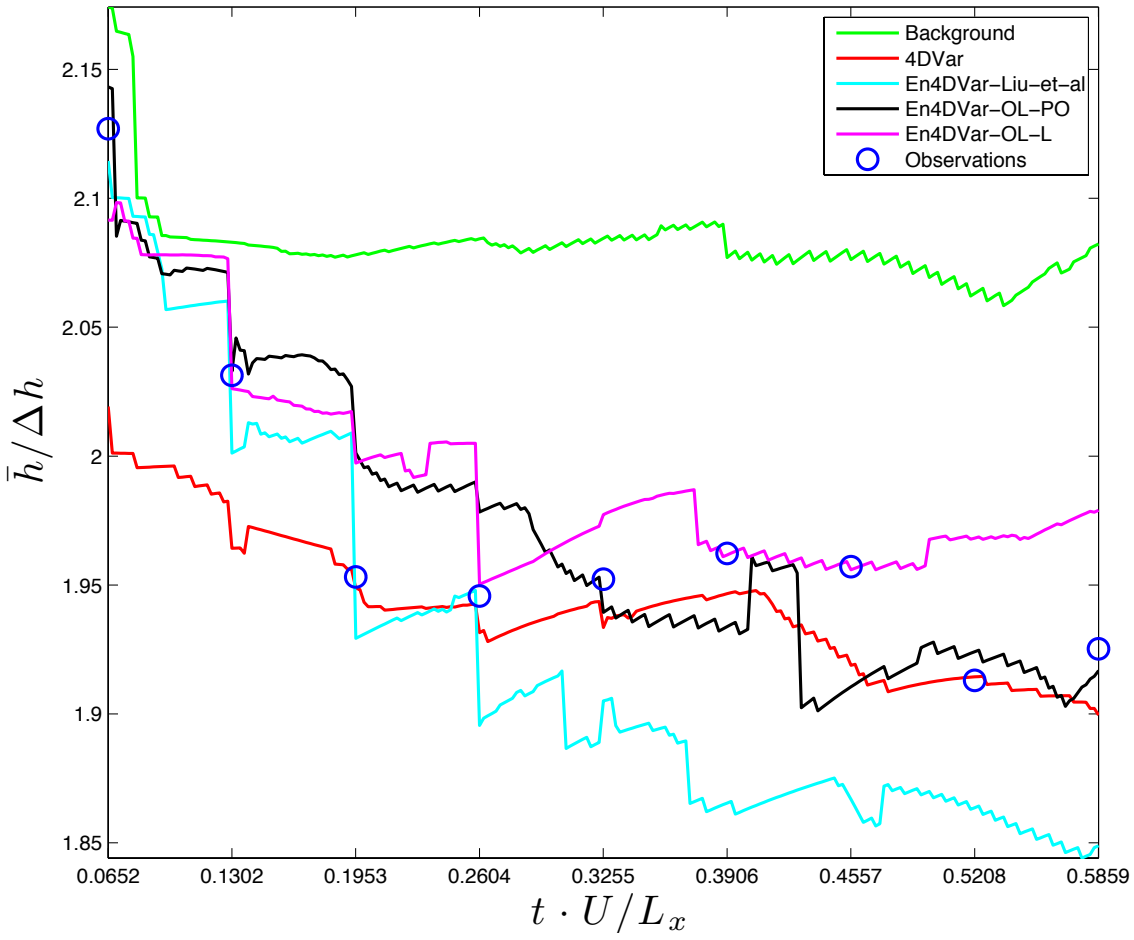


Figure 7.3 – Mean surface height of the wave crest region as a function of time - comparison of different variational data assimilation approaches results

We also compared the global free surface height distribution in figure 7.4. According to these free surfaces, we can see that the 4DVar solution showed some difficulties to handle the discontinuities at the boundaries of the regions in which the data have been extrapolated. Discontinuities in the 4DVar solution between the observed regions and the very noisy region appeared clearly. The En4DVar provided much more satisfying results on the boarders. They were smoother and corresponded clearly to a better compromise between the observation and the model.

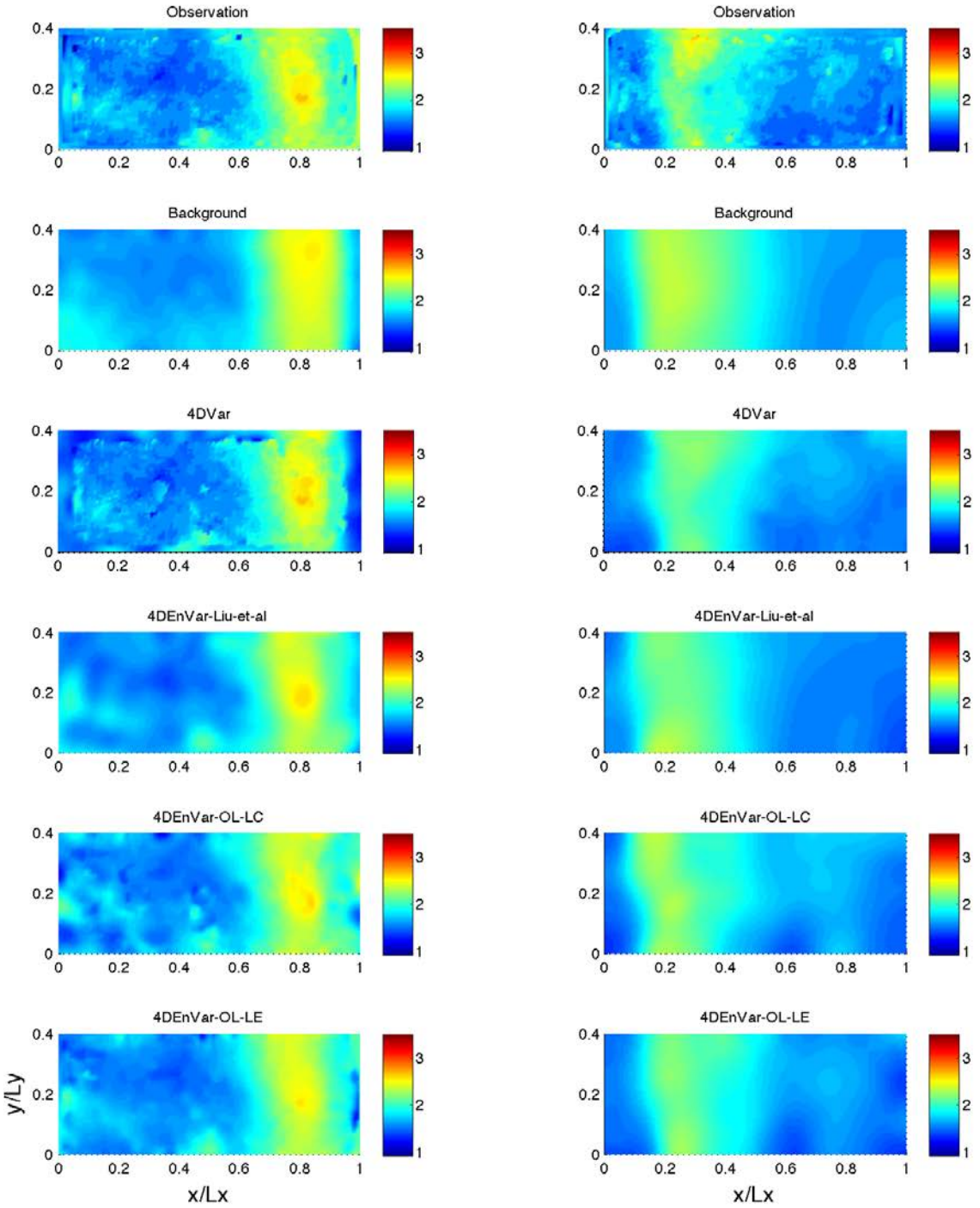


Figure 7.4 – Height field comparison, left column at $t \cdot U/L_x \approx 0.0652$, right column at $t \cdot U/L_x \approx 0.5859$, from top to bottom: Background, En4DVar-Liu-et-al, En4DVar-OL-LC, En4DVar-OL-LE, 4DVar, Observation

7.2 Sea surface temperature image data from satellite sensors

We are now interested in another kind of dynamics which depicts the geophysical flow in a larger space and time scale than the shallow water model. At these large scales, the Coriolis

force is thus dominant in these scenario compared to the inertial force (characterized by low Rossby number). And geophysical flows normally respect the geostrophic state. So we focus in this chapter on the surface quasi-geostrophic model, which is a 2D simplification of 3D quasi-geostrophic equations. Those quasi-geostrophic equations are usually encountered in the modeling of a stratified fluid undergoing strong rotation effects. A general quasi-geostrophic equation for nonlinear motions is given as (Cushman-Roisin and Beckers, 2011),

$$\frac{\partial q}{\partial t} + J(\psi, q) = 0, \tag{7.1}$$

where q is the potential vorticity,

$$q = \boldsymbol{\nabla}^2\psi + \frac{\partial}{\partial z}(\frac{f_0^2}{N^2}\frac{\partial \psi}{\partial z}), \tag{7.2}$$

Here, ψ, is the stream function, J denotes the Jacobian operator: $J(a,b) = (\partial a/\partial x)(\partial b/\partial y) - (\partial a/\partial y)(\partial b/\partial x)$ and N^2 stands for the stratification frequency. The second term of the RHS of (7.2) cancels when we do not consider the stratification effects.

7.2.1 Dynamical model configuration

The evolution of the surface buoyancy or of the temperature field, θ, are governed by similar relationship (Held et al., 1995):

$$\frac{\partial \theta}{\partial t} + J(\psi, \theta) = 0, \quad \text{at} \quad z = 0. \tag{7.3}$$

This corresponds to the inviscid SQG equation. The dissipative SQG equation is given by Constantin and Wu (1999):

$$\partial_t\theta + u \cdot \boldsymbol{\nabla}\theta + \kappa(-\Delta)^\alpha\theta = 0, \tag{7.4}$$

where $0 < \alpha \leq 1$, κ denotes the viscosity coefficient and $\kappa \geq 0$. Function u denotes the velocity filed that can be deduced from the scalar field θ by Riesz transforms:

$$u = (-\partial_y, \partial_x)\psi = (-\partial_y(-\Delta)^{-\frac{1}{2}}\theta, \partial_x(-\Delta)^{-\frac{1}{2}}\theta), \tag{7.5a}$$

$$(-\Delta)^{\frac{1}{2}}\psi = \theta. \tag{7.5b}$$

The operator $(-\Delta)^{\frac{1}{2}}$ is the fractional Laplacian operator. Such an operator can be understood by means of Fourier spectral method. In fact, if the Laplacian operator satisfies,

$$(-\Delta)\varphi_j = \lambda_j\varphi_j,$$

we thus can define φ_j as the orthonormal eigenfunctions and λ_j as the associated eigenvalues. consequently, it can be shown that if θ can be expressed as Fourier series on the basis of φ_j,

$$\theta = \sum_{j=0}^{\infty} \hat{\theta}_j\varphi_j, \tag{7.6}$$

where $(\hat{\bullet})$ denotes the Fourier coefficient. Then the fractional Laplacian holds for,

$$(-\Delta)^\alpha\theta = \sum_{j=0}^{\infty} \hat{\theta}_j\lambda_j^\alpha\varphi_j. \tag{7.7}$$

Note that the basis φ and λ have different expressions for different boundary conditions. For periodic boundary $x \in [a, b], L = b - a$, these two function read: $\lambda_j = (\frac{2\pi j}{L})^2$ and $\varphi_j = e^{i\frac{2\pi j}{L}(x-a)}$ (Bueno-Orovio et al., 2012). The equation above allow us to solve efficiently the SQG equation in the spectral space.

7.2.2 Numerical scheme

Space discretization

We rely on a 2D square domain $[0, L]^2$ to study the numerical scheme of the Fourier spectral methods. The Fourier series of the 2D function θ can be written as,

$$\theta(x, y) = \sum_{k_1=-\infty}^{+\infty} \sum_{k_2=-\infty}^{+\infty} \hat{\theta}(k_1, k_2) e^{i2\pi k_1/L} e^{i2\pi k_2/L}, \tag{7.8}$$

Numerically, a truncated series expansion is calculated in which the dimension of Fourier discrete vector is set to the number of spatial discretization points N^2 (N is the spatial grid points in one direction) (Chehab et al., 2012).

$$\theta(x, y) \approx \sum_{k_1=-\frac{N}{2}+1}^{\frac{N}{2}} \sum_{k_2=-\frac{N}{2}+1}^{\frac{N}{2}} \hat{\theta}(k_1, k_2) e^{i2\pi k_1/L} e^{i2\pi k_2/L}, \tag{7.9}$$

and the Fourier coefficient $\hat{\theta}$ is calculated likewise. The dissipative term reads,

$$(-\Delta)^\alpha \theta = \sum_{k_1=-\frac{N}{2}+1}^{\frac{N}{2}} \sum_{k_2=-\frac{N}{2}+1}^{\frac{N}{2}} \|\mathbf{k}\|^\alpha \hat{\theta}(k_1, k_2) e^{i2\pi k_1/L} e^{i2\pi k_2/L}. \tag{7.10}$$

In practice, the Fourier coefficient $\hat{\theta}$ and the inverse construction of θ can be computed efficiently by fast Fourier transform (FFT) algorithm.

Time discretization

A simple forward Euler discretization scheme can be used to split the time derivative. Suppose that at time step $[t_n, t_{n+1}]$, θ^n is known and the advection term is approximated using a fixed point algorithm, we seek then the value of $\theta^{n+1,m}$ according to:

$$\frac{\theta^{n+1,m} - \theta^n}{\Delta t} + u^{n+1,m-1} \cdot \boldsymbol{\nabla} \theta^{n+1,m-1} + \kappa(-\Delta)^\alpha \theta^{n+1,m} = 0, \tag{7.11}$$

where $\Delta t = t_{n+1} - t_n$, $u^{n+1} = (-\partial_y(-\Delta)^{-\frac{1}{2}}\theta^{n+1}, \partial_x(-\Delta)^{-\frac{1}{2}}\theta^{n+1})$ and $m = 1, 2, \ldots, M$. The Fourier transformation of (7.11) is possible when considering (7.9) and (7.10). For the (k_1, k_2)-th Fourier mode, the time discretization reads,

$$\hat{\theta}^{n+1,m}_{k_1,k_2} = \frac{1}{1 + \kappa|k_1, k_2|^\alpha \Delta t} [\hat{\theta}^n_{k_1,k_2} + \Delta t \hat{f}_{k_1,k_2}(u^{n+1,m-1}, \theta^{n+1,m-1})], \tag{7.12}$$

where the f denotes the advection term $u \cdot \boldsymbol{\nabla}\theta$.

7.2.3 Image data processing

The aim here is to assimilation the sea surface temperature (SST) image data. Here SST image data are assumed to be a passive scalar tracers transported by the flow. There are in principle two ways of assimilating the image data into the dynamic model. The first approach relies on a motion estimator of the image sequences. The velocity fields supplied by those motion estimation techniques are then considered as pseudo-observations. In this case image data are thus used indirectly in the assimilation process. Therefore, the performance of this approach largely depend on the efficiency of the motion estimator.

In this study we seek to use directly the image data through the constitution of an observation model linking the model state variables and the image data. The passive scalar hypothesis leads us naturally to consider a transport equation on the luminance function I. This relation also called the optical-flow constraint reads:

$$\frac{dI(\mathrm{x},t)}{dt} = \frac{\partial I(\mathrm{x},t)}{\partial t} + \boldsymbol{\nabla} I(\mathrm{x},t) \cdot u_p(x,t) = 0, \tag{7.13}$$

In this equation u_p denotes the vector fields transporting I at the pixel scale. This equation simply states that the material derivative of luminance function is zero. In reality, image data are usually denser in space and scarcer in time. The validity of the above equation is then likely to be not valid in large regions of the image plane. For this reason, we opt instead for the displaced frame difference operator.

Displaced frame difference

This displaced frame difference equation is given by,

$$DFD(d_t(\mathrm{x})) = I(\mathrm{x} + \Delta x, t + \Delta t) - I(\mathrm{x},t). \tag{7.14}$$

d_t denotes the displacement field. We can easily see that the DFD equation can be viewed as an integrated version of the optical flow constraint equation (7.13).

$$I(\mathrm{x} + \Delta x, t + \Delta t) - I(\mathrm{x},t) = 0. \tag{7.15}$$

Based on this relation, we can construct a nonlinear observation model,

$$I(\mathrm{x},k) = I(\mathrm{x}_{k+1}, k+1) + \gamma_k. \tag{7.16}$$

Here the nonlinear observation operator $\mathbb{H}$ is implicitly embraced in the displaced image function $I(\mathrm{x}_{k+1}, k+1)$ where $\mathrm{x}_{k+1} = \mathrm{x} + d(x)$ and $d(x) = \int_k^{k+1} u_p dt$. The velocity field u_p is derived from the physical flow velocity u by,

$$u_p = u\frac{\Delta t_{obs}}{\Delta x}, \tag{7.17}$$

where Δx is the grid spacing and Δt_{obs} denotes the time interval of two image sequence. γ_k is an additional Gaussian random noise with covariance $\mathbf{R}$. It is worth recalling that although the tangent linear observation operator $\partial_{\mathbf{x}}\mathbb{H}$ appears explicitly in the gradient expression (5.7), its approximation relying on propagation of ensemble perturbation matrix projected into observation space (Eqs. (5.5) and (5.6)) only needs the intervention of the full nonlinear observation operator $\mathbb{H}$. This is considered as an advantage of ensemble-based method.

The observation error covariance matrix related to the image data is assumed to be a diagonal matrix: $\mathbf{R}(x, x') = \sigma_k^2(x)\delta(x - x')$ at image time level k. Its variance terms can be estimated from the ensemble,

$$\sigma_k^2(\mathrm{x}) = \frac{1}{N-1}\sum_{i=1}^{N}(I(\mathrm{x} + d^i(\mathrm{x}), k+1) - \bar{I}_d(\mathrm{x}, k+1))^2 + \epsilon, \tag{7.18}$$

where ϵ is a minimum variance to avoid singular values when calculating the inverse observation error covariance matrix, d^i is the i-th ensemble member's displacement associated with the velocity and $\bar{I}_d$ is the ensemble mean of the displaced image.

7.2.4 Assimilation scheme configurations

Another great advantage of ensemble-based assimilation method compared to variational methods is that their implementations are to a large extent model independent. In standard 4DVar approach, different dynamical models require the construction of corresponding linear tangent and adjoint models. Design and maintenance of such adjoint model constitutes a substantial portion of the complete work of 4DVar. In ensemble-based method, on the contrary, as long as the dynamic model is ready, the ensemble-based data assimilation scheme can be directly applied. Consequently, the extra works of ensemble-based method is to find a better ensemble representing of the background error statistics and better update schemes able to maintain the ensemble spread. To achieve this goal, a systematic study of the error statistics must be carried out and there are certain cases where one can only resort to a 'try and error' strategy.

Here we will use the same strategies proposed in chapter 5. The problem remains to find several terms related to the DA process.

We began with a synthetic case in which the image data sequences are converted from the sequences of state variable matrix. The observations were generated by adding i.i.d. Gaussian noise 0.1 to the reference state per 25 frames of dynamic states. We choose a single assimilation window with 9 observations. The background initial state is obtained by adding an homogeneous Gaussian noise with variance 1. The Gaussian random fields with the same variance is used to generate the ensemble with the background state as the mean.

7.2.5 Results and discussions

The buoyancy (or surface temperature) fields are shown is figure 7.5. These scalar maps are taken as the observations of our assimilation problem. The figures 7.5b and 7.5c illustrate the surface buoyancy fields before and after the assimilation process. The ground truth of buoyancy field is shown in 7.5d for evaluation purpose. We can see immediately that the initial analysis state is closer to the ground truth state. The analysis state can better reconstruct the small scale variations and discontinuities, which have been overlooked in the background field.

The RMSE comparison and the vorticity maps are shown in figure 7.6. The RMSE values are calculated in such a way as a function of the discrepancy of the target trajectory and the observation in the image space that we can observe well their performance at the image level. The background state and the analysis state must be transformed into image

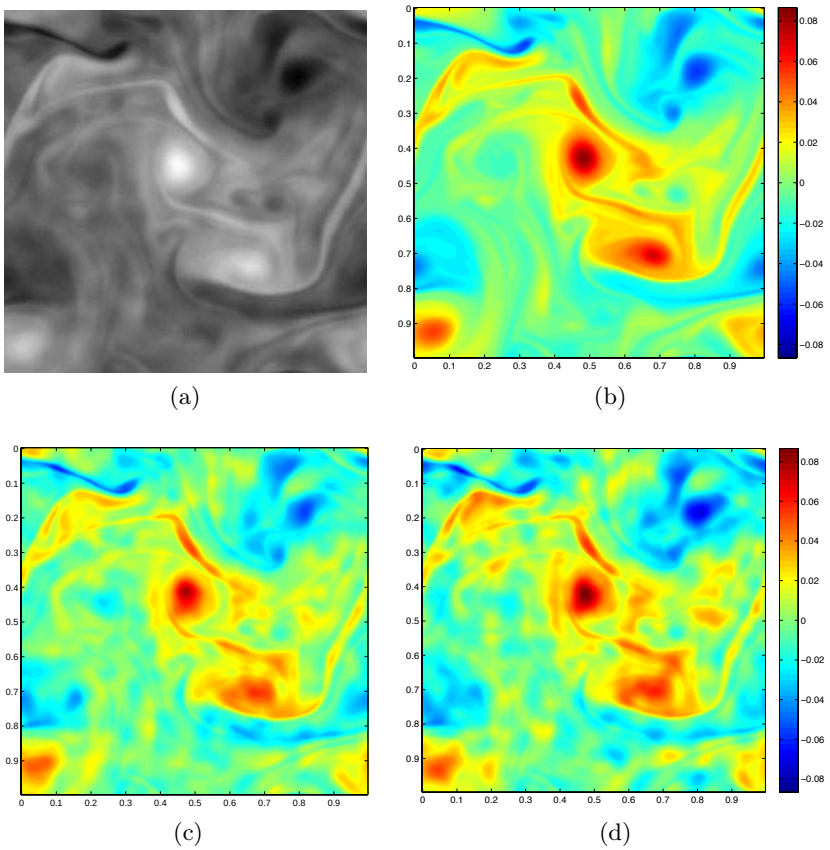


Figure 7.5 – The synthetic flow: (a) Image observation, (b) The background buoyancy state, (c) The analysis buoyancy state, (d) The ground truth buoyancy state at the initial analysis time $t = 0.25s$.

space. The RMSE equation reads:

$$\text{RMSE} = \frac{1}{n}\sqrt{\sum_{i=1}^{n}(\tilde{I}^{f,a}(x) - I^{obs}(x))^2}. \tag{7.19}$$

The figures 7.6b, 7.6c and 7.6d demonstrate the surface vorticity fields of the background, the analysis and the ground truth respectively. The performance of our ensemble-based algorithm is again verified.

7.3 Summary

In this chapter, we have shown two applications with image data assimilation. The former case couples Kinect-captured surface height image data with a shallow water model; the latter case concentrates on typical image data processing and studies the assimilation of image data into a SQG model.

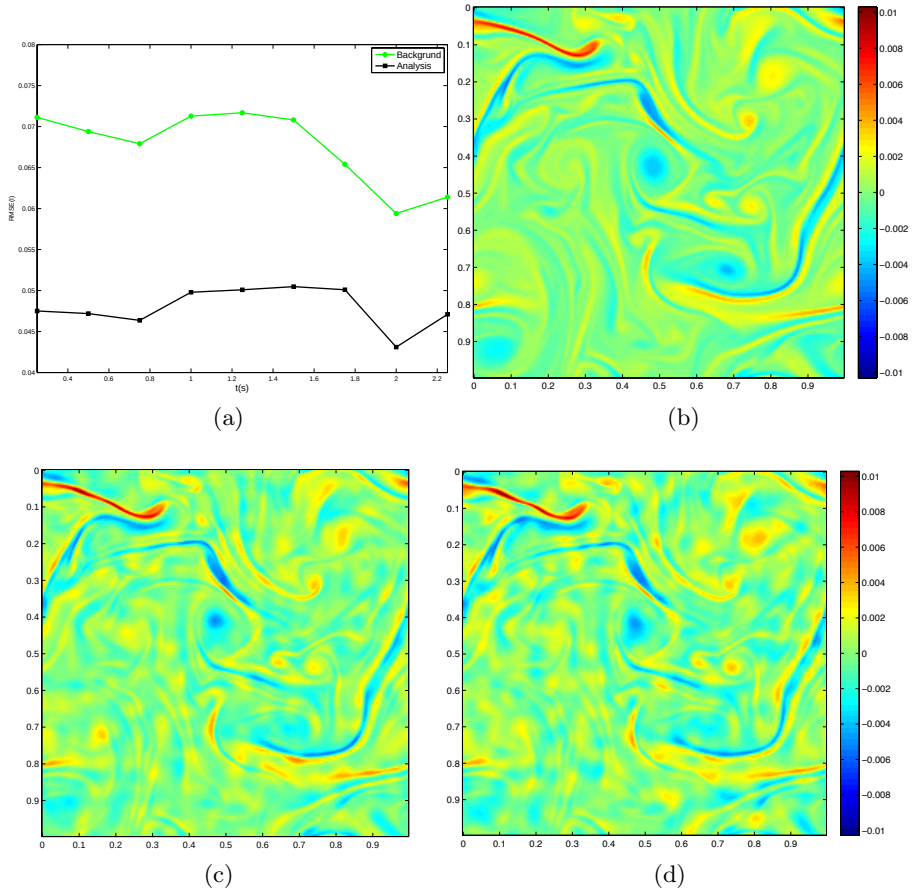


Figure 7.6 – (a) The RMSE comparison in observation space, (b) The background vorticity state, (c) The analysis vorticity state, (d) The ground truth vorticity state at the initial analysis time $t = 0.25s$.

Part IV

Stochastic Model Approach

Chapter 8

Stochastic Shallow Water Equations

In this chapter we will discuss the ensemble-based data assimilation techniques in the framework of stochastic dynamic models. The stochastic Navier-Stokes equations discussed in this chapter are based on the recent work of Mémin (2014) and Kadri Harouna and Mémin (2014).

8.1 Why stochastic modeling?

In Computational Fluid Dynamics (CFD), the flow simulated on different grid resolutions may appear to be quite different. This inconsistency is caused by both physical or numerical processes which can hardly be properly modeled by abruptly coarse-graining the simulation. From a numerical point of view, the discretization error naturally decreases as the resolution is refined. This can be assured by a Grid Convergence Study to determine the tolerable resolution on the condition that all relevant physical processes are considered. The Direct Numerical Simulations (DNS) has provided very promising results in recent years with the increase of the available computational power. However, regarding to the usual geophysical or engineering applications, this paradigm, which consists in simulating the smallest scales of the flow, is hardly applicable due to the dimension of the problem to handle. A common approach is therefore to simulate the statistically averaged flow on a coarse grid and modeling the small scale using technique called the subgrid-scale parameterization. These subgrid-scale process is usually related to the turbulent fluctuations. Two approaches have been successfully implemented for large scale simulation: the Large-Eddies Simulations (LES) and the Reynolds Average Numerical Simulations (RANS). The LES approach (Deardorff, 1970; Moeng, 1984) defines the governing equations of the resolved-scale variables as a convolution of the real equations with a low-pass filter, and leaves the subgrid-scale fluctuations to be determined elsewhere; on the other hand, the RANS approach is built on the idea of Reynolds decomposition.

For geophysical models, a bunch of processes can explain the unresolved components. Those processes are related to the mesoscale and microscale geophysical systems. When one faces a simulation at different resolutions, it is important to understand the actions of those unresolved components on the resolved one and to take them into account in the dynamics. Stochastic modeling has been already applied to formulate such subgrid effects (Frederiksen et al., 2013). In our approach, instead of considering additional random forcing terms, we try to incorporate the subgrid effects from an uncertainty principle included from the start in the dynamics physical derivation.

The basic idea of the stochastic flow dynamics is built on the assumption that the Lagrangian fluid particle displacement can be separated in two parts: a smooth differentiable part and a non-differentiable stochastic part:

$$\boldsymbol{X}(x,t) = \mathbf{w}(\boldsymbol{X}(x,t),t)dt + \boldsymbol{\sigma}(\boldsymbol{X}(x,t),t)d\boldsymbol{B_t}, \tag{8.1}$$

where $\boldsymbol{X}$ is the flow map, $\mathbf{w}$ is the deterministic velocity and $d\boldsymbol{B_t}$ is a vector of d-dimensional Wiener increment (d=2,3). The term $\sigma(x,t)d\boldsymbol{B_t}$ represents the stochastic component whereas $\mathbf{w}$, denotes the smooth drift component. The flow fields are defined on the flow domain Ω. From Eq. 8.1, we can have the Eulerian velocity fields,

$$\boldsymbol{U}(x,t) = \mathbf{w}(x,t) + \boldsymbol{\sigma}(x,t)d\boldsymbol{B_t}. \tag{8.2}$$

We remark here that Constantin and Iyer (2008) and Mikulevicius and Rozovskii (2004) also proposed a closely related representation of Navier-Stokes equations on stochastic Lagrangian paths.

The Reynolds transport theorem plays an essential role in formulating the Navier-Stokes equations. It allows expressing conservation laws (mass, momentum and energy). Similarly our derivation process relies on a stochastic extension of the Reynolds transport theorem. We review here some important aspects related to this theorem. A comprehensive derivation process of this theorem (Mémin, 2014) is presented in appendix A.

The stochastic differential equation 8.1 holds for $\boldsymbol{X}$ if it satisfies the stochastic integral defined through,

$$\boldsymbol{X}(x,t) - \boldsymbol{X}(x,t_0) = \int_{t_0}^{t} \mathbf{w}(\boldsymbol{X}(x,t),t)dt + \int_{t_0}^{t} \boldsymbol{\sigma}(\boldsymbol{X}(x,t),t)d\boldsymbol{B_t} \quad \text{for} \quad t_0 \leq t \leq t_f, \tag{8.3}$$

Ito's formula states (Allen, 2007) that a scalar function $\phi(\boldsymbol{X},t)$ in terms of stochastic process $\boldsymbol{X}$ also satisfies a stochastic differential equation.

Here we consider a 1D case to illustrate Ito's rule. Imagine that ϕ is any quantity transported by the flow, if $\phi(x,t)$ is twice continuous differentiable, then the Taylor series expansion of ϕ reads,

$$d\phi = \frac{\partial \phi}{\partial t}dt + \frac{\partial \phi}{\partial x}d\boldsymbol{X} + \frac{1}{2}\frac{\partial^2 \phi}{\partial x^2}(d\boldsymbol{X})^2. \tag{8.4}$$

Inserting Eq.(8.2), we get,

$$d\phi = \frac{\partial \phi}{\partial t}dt + \frac{\partial \phi}{\partial x}(\mathbf{w}dt + \boldsymbol{\sigma}d\boldsymbol{B_t}) + \frac{1}{2}\frac{\partial^2 \phi}{\partial x^2}(\mathbf{w}dt + \boldsymbol{\sigma}d\boldsymbol{B_t})^2, \tag{8.5}$$

According to Ito's rule, the term associated with $(d\boldsymbol{B_t})^2$ is in the order of 'dt' and has to be retained, (which is often informally written as $d\boldsymbol{B_t} \propto \sqrt{dt}$). The other higher order terms $(dt)^2$ and $dtd\boldsymbol{B_t}$ can be discarded. This leaves us with:

$$d\phi = (\frac{\partial \phi}{\partial t} + \frac{\partial \phi}{\partial x}w + \frac{1}{2}\sigma^2\frac{\partial^2 \phi}{\partial x^2})dt + \frac{\partial \phi}{\partial x}\boldsymbol{\sigma}d\boldsymbol{B_t}. \tag{8.6}$$

which constitutes the simpler scalar Ito differential role. A more general form can be written, when $\boldsymbol{\sigma}d\boldsymbol{B_t}$ is not anymore a scalar but rather a vector:

$$d\phi(\boldsymbol{X},t) = (\frac{\partial \phi}{\partial t} + \boldsymbol{\nabla}\phi \cdot \mathbf{w} + \frac{1}{2}\sum_{i,j} d\langle X_i, X_j\rangle \frac{\partial^2 \phi}{\partial x_i \partial x_j})dt + \boldsymbol{\nabla}\phi \cdot \boldsymbol{\sigma}d\boldsymbol{B_t}. \tag{8.7}$$

The term $d\langle X_i, X_j\rangle$ is the quadratic co-variation defined as

$$\langle X, Y\rangle_t = \lim_{\max|t_{i+1}-t_i|\to 0} \sum (X(t_{i+1}) - X(t_i))(Y(t_{i+1}) - Y(t_i))^T, \tag{8.8}$$

where the convergence has to be interpreted as a convergence in probability. In our case the quadratic variation of the uncertainty component reads,

$$\begin{aligned}\langle \int_0^t (\boldsymbol{\sigma} d\boldsymbol{B_t})_i, \int_0^t (\boldsymbol{\sigma} d\boldsymbol{B_t})_j\rangle &= \int_0^t \sum_k \sigma_{ik}\sigma_{kj} ds, \\ &\triangleq \int_0^t a_{ij} ds. \end{aligned} \tag{8.9}$$

The term a_{ij} denotes the uncertainty variance tensor. It corresponds to the time derivative of the quadratic variation process and gathers the diagonal element of the covariance matrix tensor: $\mathbf{Q}_{ij} = \int_\Omega \sum_k \sigma_{ik}(x,z)\sigma_{jk}(z,y)dz$. It has the same unit as kinematic viscosity$(m^2 s^{-1})$, likewise the diffusion tensor σ_{ij} has the unity of $(ms^{-\frac{1}{2}})$.

A function, ϕ, which satisfies the Ito formula (8.7) can only be a deterministic function. In our case, $\phi(x,t)$ is necessarily a random function (otherwise a conservation property would cancel the stochastic part), and its stochastic differential must be evaluated by the so called Ito-Wentzell formula:

$$d\phi(\boldsymbol{X}, t) = d_t\phi + \boldsymbol{\nabla}\phi \cdot d\boldsymbol{X} + \frac{1}{2}\sum_{i,j} d\langle X_i, X_j\rangle \frac{\partial^2\phi}{\partial x_i \partial x_j} dt + \sum_i d\langle \frac{\partial \phi}{\partial x_i}, \int_0^t (\boldsymbol{\sigma} \boldsymbol{B_t})^i\rangle dt. \tag{8.10}$$

Compared to the Ito's formula (8.7), $\frac{\partial\phi}{\partial t}$ is replaced by $d_t\phi$ since ϕ is a random function of $\boldsymbol{X}$ and a covariation term appears as well.

The term $d_t\phi$ denotes the differential of ϕ at a fixed grid point. It necessarily satisfies a stochastic differential equation. Such an stochastic differential equation allow us to formulate the stochastic Reynolds transport theorem (appendix A).

Denoting q as a scalar function bounded in a material fluid volume $\mathcal{V}$ transported by (8.1), the rate of change of q is given by:

$$d\int_{\mathcal{V}} q dx = \int_{\mathcal{V}} [d_t q + \boldsymbol{\nabla}\cdot(q\mathbf{w})dt - \frac{1}{2}\sum_{ij} \frac{\partial^2}{\partial x_i \partial x_j}(a_{ij}q)|_{\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}=0} dt + \frac{1}{2} \parallel \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} \parallel^2 q dt + \boldsymbol{\nabla}\cdot(q\boldsymbol{\sigma} d\boldsymbol{B_t})] dx, \tag{8.11}$$

where the third term must be computed for a divergence free diffusion tensor $\boldsymbol{\sigma}$. A divergence free stochastic component provides a simpler version of the stochastic Reynolds transport theorem. In that case, the differential of q within the material volume reads,

$$d\int_{\mathcal{V}} q dx = \int_{\mathcal{V}} [d_t q + \boldsymbol{\nabla}\cdot(q\mathbf{w})dt - \frac{1}{2}\sum_{ij} \frac{\partial^2}{\partial x_i \partial x_j}(a_{ij}q)dt + \boldsymbol{\nabla} q \cdot \boldsymbol{\sigma} d\boldsymbol{B_t})] dx. \tag{8.12}$$

Through the stochastic Reynolds transport theorem, the mass conservation and the momentum conservation equation can be (almost) directly derived. In the following sections we will briefly present how this principle can be applied to derive a stochastic expression of the shallow water equations.

8.2 The stochastic 2D nonlinear shallow water model

We are particularly interested in the stochastic version of shallow water equations. The 2D shallow water equation with uncertainty is given through equations (42) in Mémin (2014),

$$\Big(\frac{\partial \mathbf{w}^h}{\partial t} + \mathbf{w}\nabla^T\mathbf{w}^h - \frac{1}{2}\sum_{i,j}\partial_{x_i}\partial_{x_j}(a_{ij}\mathbf{w}^h)\Big)\rho = -g\rho\nabla h, \tag{8.13a}$$

$$d_t h + \Big(\nabla\cdot(h\mathbf{w}^h) - \frac{1}{2}\sum_{i,j}\partial_{x_i}\partial_{x_j}(a_{ij}h)\Big)dt + \nabla h(\boldsymbol{\sigma} d\boldsymbol{B_t}) = 0, \tag{8.13b}$$

$$\nabla d\hat{p} = -\rho(\mathbf{w}^h\boldsymbol{\nabla})(\boldsymbol{\sigma} d\boldsymbol{B_t}). \tag{8.13c}$$

Note that $\mathbf{w}^h$ stands for the horizontal velocities u and v. The 3D Eulerian displacement fields as:

$$\begin{pmatrix} U(x,y,t) \\ V(x,y,t) \\ W(x,y,t) \end{pmatrix} = \begin{pmatrix} udt + (\boldsymbol{\sigma} d\boldsymbol{B_t})_x \\ vdt + (\boldsymbol{\sigma} d\boldsymbol{B_t})_y \\ wdt + (\boldsymbol{\sigma} d\boldsymbol{B_t})_z \end{pmatrix}. \tag{8.14}$$

Besides the dynamic model, the uncertainty component must satisfy a divergence free constraint condition,

$$\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}^h = 0, \tag{8.15a}$$

$$\boldsymbol{\nabla}\cdot(\boldsymbol{\nabla}\cdot\boldsymbol{a}^h) = 0, \tag{8.15b}$$

The 2D shallow water equation with uncertainty is derived through depth average integration of the Navier-Stokes equations under uncertainty. The assumption allowing this derivation is still the hydrostatic equilibrium relation.

8.2.1 Continuity equation

The typical flow to which the shallow water equation can be applied is usually non-viscous and incompressible. Therefore we have

$$\boldsymbol{\nabla}\cdot(\mathbf{w}dt + \boldsymbol{\sigma} d\boldsymbol{B_t}) = 0. \tag{8.16}$$

Now we integrate above equation along the depth from the bottom h_b to the surface h_s using Leibnitz integration rules take into consideration of the BC at the bottom and the free surface. First we rephrase the Leibnitz integration rule:

$$\frac{\partial}{\partial x}\int_{h_b}^{h_s} u dz = \int_{h_b}^{h_s}\frac{\partial u}{\partial x}dz + u|_{h_s}\frac{\partial h_s}{\partial x} - u|_{h_b}\frac{\partial h_b}{\partial x}.$$

The integration give us:

$$\begin{aligned}
&\int_{h_b}^{h_s}\Big((\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z})dt + \frac{\partial(\boldsymbol{\sigma} d\boldsymbol{B_t})_x}{\partial x} + \frac{\partial(\boldsymbol{\sigma} d\boldsymbol{B_t})_y}{\partial y} + \frac{\partial(\boldsymbol{\sigma} d\boldsymbol{B_t})_z}{\partial z}\Big)dz = \\
&\frac{\partial}{\partial x}\int_{h_b}^{h_s} u dz dt + \frac{\partial}{\partial y}\int_{h_b}^{h_s} v dz dt - u|_{h_s}\frac{\partial h_s}{\partial x}dt + u|_{h_b}\frac{\partial h_b}{\partial x}dt - v|_{h_s}\frac{\partial h_s}{\partial y}dt + v|_{h_b}\frac{\partial h_b}{\partial y}dt \\
&+ w|_{h_s}dt - w|_{h_b}dt + \frac{\partial}{\partial x}\int_{h_b}^{h_s}(\boldsymbol{\sigma} d\boldsymbol{B_t})_x dz + \frac{\partial}{\partial x}\int_{h_b}^{h_s}(\boldsymbol{\sigma} d\boldsymbol{B_t})_y dz \\
&- (\boldsymbol{\sigma} d\boldsymbol{B_t})_x|_{h_s}\frac{\partial h_s}{\partial x} + (\boldsymbol{\sigma} d\boldsymbol{B_t})_x|_{h_b}\frac{\partial h_b}{\partial x} - (\boldsymbol{\sigma} d\boldsymbol{B_t})_y|_{h_s}\frac{\partial h_s}{\partial y} + (\boldsymbol{\sigma} d\boldsymbol{B_t})_y|_{h_b}\frac{\partial h_b}{\partial y} \\
&+ (\sigma d\boldsymbol{B_t})_z|_{h_s} - (\boldsymbol{\sigma} d\boldsymbol{B_t})_z|_{h_b} = 0.
\end{aligned} \tag{8.17}$$

In order to simplify the equation above, we must consider the kinematic boundary conditions at the surface and the bottom from the stochastic transport principle,

$$W|_{h_s} = d_t h_s + (\boldsymbol{\nabla} \cdot (h_s \mathbf{w}^h) - \frac{1}{2}\sum \partial_{x_i}\partial_{x_j}(a_{ij}h_s))dt + \boldsymbol{\nabla} \cdot (h_s \boldsymbol{\sigma} d\boldsymbol{B_t}) \tag{8.18a}$$

$$W|_{h_b} = (\boldsymbol{\nabla} \cdot (h_b \mathbf{w}^h) - \frac{1}{2}\sum \partial_{x_i}\partial_{x_j}(a_{ij}h_b))dt + \boldsymbol{\nabla} \cdot (h_b \boldsymbol{\sigma} d\boldsymbol{B_t}) \tag{8.18b}$$

By introducing the kinematic boundary conditions (8.18) into system (8.17), most of the terms associated with the boundaries cancel themselves in addition to the fact that the bottom is fixed. Besides we define the depth-averaged field flow as,

$$\int_{h_b}^{h_s} u dz = h\bar{u}. \tag{8.19}$$

Then the Eq.(8.17) can be finally written as

$$d_t h + \Big(\frac{\partial(h\bar{u})}{\partial x} + \frac{\partial(h\bar{v})}{\partial y} - \frac{1}{2}\sum \partial_{x_i}\partial_{x_j}(a_{ij}h)\Big)dt + \frac{\partial(h\boldsymbol{\sigma} d\boldsymbol{B_t})_x}{\partial x} + \frac{\partial(h\boldsymbol{\sigma} d\boldsymbol{B_t})_y}{\partial y} = 0 \tag{8.20}$$

Note that $h = h_s - h_b$ and $d_t h_s$ equals $d_t h$ when the bottom is fixed.

8.2.2 Momentum conservation equation

To deduce the momentum conservation equation, we start from the transportation of momentum,

$$\int_\Omega \Big[d_t(\rho w_i) + \boldsymbol{\nabla} \cdot (\rho w_i \mathbf{w})dt - \frac{1}{2}\sum_{ij} \frac{\partial^2}{\partial x_i \partial x_j}(a_{ij}\rho w_i)|_{\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}=0}dt + \frac{1}{2} \parallel \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} \parallel^2 \rho w_i dt$$
$$+\boldsymbol{\nabla} \cdot (\rho w_i \boldsymbol{\sigma} d\boldsymbol{B_t})\Big]dx = \int_\Omega \rho \mathbf{g} dt dx + \int_{\partial\Omega} \mathbf{T}\mathbf{n} dt dl. \tag{8.21}$$

Transform the RHS of above formula by Gauss's Theorem,

$$\int_\Omega (\rho \boldsymbol{g} + \boldsymbol{\nabla} \cdot \boldsymbol{T})dt dx,$$

where $\boldsymbol{T}$ is the Cauchy stress tensor which equals

$$\boldsymbol{T} = -pI + \boldsymbol{\tau},$$

where

$$p = -tr(T)/3,$$

and

$$\tau_{ij} = \sigma_{ij} - \frac{\sigma_{kk}}{3}\delta_{ij}.$$

$\boldsymbol{\tau}$ is the deviatoric tensor which can be formulated by a constitutive relation,

$$\tau_{ij} = \lambda S_{kk}\delta_{ij} + 2\mu S_{ij},$$

where $S_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$ is the strain rate tensor. So the divergence of the stress tensor can be put as,

$$\boldsymbol{\nabla} \cdot T = -\boldsymbol{\nabla} p + \mu \boldsymbol{\nabla}^2 \boldsymbol{U} + (\lambda + \mu)\boldsymbol{\nabla}(\boldsymbol{\nabla} \cdot \boldsymbol{U}).$$

By Stokes hypothesis $3\lambda + 2\mu = 0$, which leads to:

$$\tau_{ij} = 2\mu(S_{ij} - \frac{1}{3}S_{kk}\delta_{ij}),$$

finally we have,

$$\boldsymbol{\nabla} \cdot T = -\boldsymbol{\nabla} p + \mu\big(\boldsymbol{\nabla}^2 \boldsymbol{U} + \frac{1}{3}\boldsymbol{\nabla}(\boldsymbol{\nabla} \cdot \boldsymbol{U})\big).$$

Balancing the Brownian motion term,

$$\int_\Omega \Big[\frac{\partial(\rho w_i)}{\partial t} + \boldsymbol{\nabla} \cdot (\rho w_i \boldsymbol{w}) - \frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i \partial x_j}(a_{ij}\rho w_i)|_{\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}=0} + \frac{1}{2}|\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}|^2 \rho w_i\Big] dx = \int_\Omega \Big(\rho \boldsymbol{g} - \boldsymbol{\nabla} p + \mu\big(\boldsymbol{\nabla}^2 \boldsymbol{w} + \frac{1}{3}\boldsymbol{\nabla}(\boldsymbol{\nabla} \cdot \boldsymbol{w})\big)\Big) dx, \quad (8.22)$$

Recall the fluid studied here is homogenous, incompressible and inviscid, thus the above equations can be greatly simplified to ($\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = 0$ and $\boldsymbol{\nabla} \cdot \boldsymbol{w} = 0$),

$$\frac{\partial \boldsymbol{w}}{\partial t} + \boldsymbol{\nabla} \cdot (\boldsymbol{w}\boldsymbol{w}) - \frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i \partial x_j}(a_{ij} w) = \boldsymbol{g} - \frac{1}{\rho}\boldsymbol{\nabla} p, \quad (8.23)$$

To build the shallow water equations, we restart from the momentum transportation Eq.(8.21), plus the force tensor analysis we have done so far, we have a local version of the momentum conservation equation.

$$d_t(\rho w_i) + \boldsymbol{\nabla} \cdot (\rho w_i \mathbf{w})dt - \frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i \partial x_j}(a_{ij}\rho w_i)dt + \boldsymbol{\nabla} \cdot (\rho w_i \boldsymbol{\sigma} d\boldsymbol{B_t}) = \rho \mathbf{g} - \boldsymbol{\nabla} p.$$

We intends to use the same procedure as the one applied to the continuity equation, take the x-directional momentum equation for example,

$$d_t u + \Big(\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z} - \frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i \partial x_j}(a_{ij}\mathbf{w})\Big)dt + \boldsymbol{\nabla} \cdot (u\boldsymbol{\sigma} d\boldsymbol{B_t}) = -\frac{1}{\rho}\frac{\partial p}{\partial x}dt. \quad (8.24)$$

We hence integrate the above equation along the depth from the bottom h_b to the surface h_s, we express below the procedure term by term for sake of clarity (from now on the bar notation associated with the depth average velocity is dropped).

- The temporal derivative term:

$$\int_{h_b}^{h_s} d_t u dz = d_t(hu) - u|_{h_s} dh_s. \quad (8.25)$$

- The convection terms:

$$\int_{h_b}^{h_s}(\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial uw}{\partial z})dzdt = \frac{\partial}{\partial x}\int_{h_b}^{h_s} u^2 dzdt + \frac{\partial}{\partial y}\int_{h_b}^{h_s} uvdzdt - u^2|_{h_s}\frac{\partial h_s}{\partial x}dt + u^2|_{h_b}\frac{\partial h_b}{\partial x}dt - uv|_{h_s}\frac{\partial h_s}{\partial y}dt + uv|_{h_b}\frac{\partial h_b}{\partial y}dt + uw|_{h_s}dt - uw|_{h_b}dt. \quad (8.26)$$

- The diffusion term (recalling $\boldsymbol{\nabla}\cdot\boldsymbol{\nabla}\cdot\boldsymbol{a}=0$):

$$\begin{aligned}
&\frac{1}{2}\int_{h_b}^{h_s}\sum\partial_i\partial_j(a_{ij}u)dz=\\
&\frac{1}{2}\int_{h_b}^{h_s}(a_{xx}\frac{\partial^2u}{\partial x^2}+a_{yy}\frac{\partial^2u}{\partial y^2}+a_{zz}\frac{\partial^2u}{\partial z^2}+2\partial_xa_{xx}\frac{\partial u}{\partial x}+2\partial_ya_{yy}\frac{\partial u}{\partial y}+2\partial_za_{zz}\frac{\partial u}{\partial z})dz=\\
&\frac{1}{2}\Big(a_{xx}(\frac{\partial}{\partial x}\int_{h_b}^{h_s}\partial_xudz-\partial_xu|_{h_s}\frac{\partial h_s}{\partial x}+\partial_xu|_{h_b}\frac{\partial h_b}{\partial x})\\
&+a_{yy}(\frac{\partial}{\partial y}\int_{h_b}^{h_s}\partial_yudz-\partial_yu|_{h_s}\frac{\partial h_s}{\partial y}+\partial_yu|_{h_b}\frac{\partial h_b}{\partial y})\\
&+a_{zz}\big(\partial_zu|_{h_s}-\partial_zu|_{h_b}\big)+2\partial_xa_{xx}(\frac{\partial}{\partial x}\int_{h_b}^{h_s}udz-u|_{h_s}\frac{\partial h_s}{\partial x}+u|_{h_b}\frac{\partial h_b}{\partial x})\\
&+2\partial_ya_{yy}(\frac{\partial}{\partial y}\int_{h_b}^{h_s}udz-u|_{h_s}\frac{\partial h_s}{\partial y}+u|_{h_b}\frac{\partial h_b}{\partial y})+2\partial_za_{zz}(u|_{h_s}-u|_{h_b}).
\end{aligned}\tag{8.27}$$

- The noise term:

$$\begin{aligned}
&\int_{h_b}^{h_s}\Big(\frac{\partial(u\boldsymbol{\sigma}d\boldsymbol{B_t})_x}{\partial x}+\frac{\partial(u\boldsymbol{\sigma}d\boldsymbol{B_t})_y}{\partial y}+\frac{\partial(u\boldsymbol{\sigma}d\boldsymbol{B_t})_z}{\partial z}\Big)dz=\\
&\frac{\partial}{\partial x}\int_{h_b}^{h_s}(u\boldsymbol{\sigma}d\boldsymbol{B_t})_xdz+\frac{\partial}{\partial y}\int_{h_b}^{h_s}(u\boldsymbol{\sigma}d\boldsymbol{B_t})_ydz\\
&-(u\boldsymbol{\sigma}d\boldsymbol{B_t})_x|_{h_s}\frac{\partial h_s}{\partial x}+(u\boldsymbol{\sigma}d\boldsymbol{B_t})_x|_{h_b}\frac{\partial h_b}{\partial x}-(u\boldsymbol{\sigma}d\boldsymbol{B_t})_y|_{h_s}\frac{\partial h_s}{\partial y}+(u\boldsymbol{\sigma}d\boldsymbol{B_t})_y|_{h_b}\frac{\partial h_b}{\partial y}\\
&+(u\boldsymbol{\sigma}d\boldsymbol{B_t})_z|_{h_s}-(u\boldsymbol{\sigma}d\boldsymbol{B_t})_z|_{h_b},
\end{aligned}\tag{8.28}$$

- The pressure term: the total pressure term p can be decomposed as the summation of a deterministic pressure part and a zero mean random pressure as well. In this way, the fast variation forcing part involved in Eq.(8.24) is to be cancelled by the random pressure part. This is a logical inference since the random pressure is generated due to the fluctuation of random velocity. Here we intend to keep the random forcing term in conservational momentum equation, instead we consider the total pressure term p alone. We know well that, under hydrostatic condition, the pressure, p, is only a function of depth from the reduced z-directional momentum equation:

$$\rho g=-\frac{\partial p}{\partial z},$$

with an additional dynamic boundary condition at the free surface:

$$p|_{h_s}=p|_{atm},$$

where $p|_{atm}$ is the atmosphere pressure at the interface of water-air. A depth integration of the equation (8.2.2) gives,

$$p|_{h_s}=p|_{h_b}-\rho g(h_s-h_b).\tag{8.29}$$

Now that $p|_{h_s}$ should equal the atmosphere pressure $p|_{atm}$, the pressure at any depth level z reads,

$$p|_z=\rho g(h_s-z)+p|_{atm},\tag{8.30}$$

and its gradient,

$$\frac{\partial p|_z}{\partial x} = \rho g \frac{\partial h_s}{\partial x}. \tag{8.31}$$

By assuming $p|_{atm}$ is homogeneous acrossing the interface, the depth integration of the pressure gradient term yields,

$$-\int_{h_b}^{h_s} \frac{1}{\rho}\frac{\partial p}{\partial x} dz = -gh\frac{\partial h_s}{\partial x}. \tag{8.32}$$

Note that $h = h_s - h_b$ and $\partial_x h_s$ only equals $\partial_x h$ when the bottom is flat. To comply with previous notations, we will use η.

Let us focus on the terms associated with the boundary. Note that the vertical velocity component satisfies $w|_{h_s}dt + (\boldsymbol{\sigma} d\boldsymbol{B_t})_z|_{h_s} = W|_{h_s}$ and $w|_{h_b}dt + (\boldsymbol{\sigma} d\boldsymbol{B_t})_z|_{h_b} = W|_{h_b}$. By regrouping the terms associated with the velocity at the surface and the bottom respectively, we have:

$$\begin{aligned} u|_{h_s}\Big(&- dh_s - u|_{h_s}\frac{\partial h_s}{\partial x}dt - v|_{h_s}\frac{\partial h_s}{\partial y}dt + w|_{h_s}dt - \frac{1}{2}\big(- 2\partial_x a_{xx}\frac{\partial h_s}{\partial x} - 2\partial_y a_{yy}\frac{\partial h_s}{\partial y} + 2\partial_z a_{zz}\big)dt \\ &- (\sigma d\boldsymbol{B_t})_x|_{h_s}\frac{\partial h_s}{\partial x} - (\sigma d\boldsymbol{B_t})_y|_{h_s}\frac{\partial h_s}{\partial y} + (\sigma d\boldsymbol{B_t})_z|_{h_s}\Big) = -\frac{1}{2}u|_{h_s}(a_{xx}\frac{\partial^2 h_s}{\partial x^2} + a_{yy}\frac{\partial^2 h_s}{\partial y^2}), \end{aligned} \tag{8.33}$$

and

$$\begin{aligned} u|_{h_b}\Big(u|_{h_b}\frac{\partial h_b}{\partial x}dt + v|_{h_b}\frac{\partial h_b}{\partial y}dt - w|_{h_b}dt - \frac{1}{2}\big(2\partial_x a_{xx}\frac{\partial h_b}{\partial x} + 2\partial_y a_{yy}\frac{\partial h_b}{\partial y} - 2\partial_z a_{zz}\big)dt \\ + (\sigma d\boldsymbol{B_t})_x|_{h_b}\frac{\partial h_b}{\partial x} + (\sigma d\boldsymbol{B_t})_y|_{h_b}\frac{\partial h_b}{\partial y} + (\sigma d\boldsymbol{B_t})_z|_{h_b}\Big) = \frac{1}{2}u|_{h_b}(a_{xx}\frac{\partial^2 h_b}{\partial x^2} + a_{yy}\frac{\partial^2 h_b}{\partial y^2}). \end{aligned} \tag{8.34}$$

Then we all the terms associated with the gradient of velocity at the surface and the bottom,

$$\begin{aligned} -\frac{1}{2}u|_{h_s}(a_{xx}\frac{\partial^2 h_s}{\partial x^2} + a_{yy}\frac{\partial^2 h_s}{\partial y^2}) - \frac{1}{2}(\partial_x u|_{h_s}a_{xx}\frac{\partial h_s}{\partial x} + \partial_y u|_{h_s}a_{yy}\frac{\partial h_s}{\partial y}) = \\ -\frac{1}{2}\big(a_{xx}\frac{\partial(u|_{h_s}\partial_x h_s)}{\partial x} + a_{yy}\frac{\partial(u|_{h_s}\partial_y h_s)}{\partial y}\big), \end{aligned} \tag{8.35}$$

$$\begin{aligned} \frac{1}{2}u|_{h_b}(a_{xx}\frac{\partial^2 h_b}{\partial x^2} + a_{yy}\frac{\partial^2 h_b}{\partial y^2}) + \frac{1}{2}(\partial_x u|_{h_b}a_{xx}\frac{\partial h_s}{\partial x} + \partial_y u|_{h_b}a_{yy}\frac{\partial h_b}{\partial y}) = \\ \frac{1}{2}\big(a_{xx}\frac{\partial(u|_{h_b}\partial_x h_b)}{\partial x} + a_{yy}\frac{\partial(u|_{h_b}\partial_y h_b)}{\partial y}\big). \end{aligned} \tag{8.36}$$

Taking account of the assumption we made for the shallow-water system (no vertical variation for the horizontal velocities), $u|_{h_s} = u|_{h_b}$, then (8.35) and (8.36) sum to:

$$-\frac{1}{2}\big(a_{xx}\frac{\partial(u\partial_x h)}{\partial x} + a_{yy}\frac{\partial(u\partial_y h)}{\partial y}\big). \tag{8.37}$$

Other terms associated with the gradient of velocity along the bottom and the surface can be interpreted as the lateral stress. So finally we have,

$$
\begin{aligned}
&d_t(hu) + \Big(\frac{\partial(hu^2)}{\partial x} + \frac{\partial(huv)}{\partial y} \\
&- \frac{1}{2}\big(a_{xx}\frac{\partial(h\partial_x u)}{\partial x} + a_{yy}\frac{\partial(h\partial_y u)}{\partial y} + 2\partial_x a_{xx}\frac{\partial(hu)}{\partial x} + 2\partial_y a_{yy}\frac{\partial(hu)}{\partial y} + a_{xx}\frac{\partial(u\partial_x h)}{\partial x} + a_{yy}\frac{\partial(u\partial_y h)}{\partial y}\big)\Big)dt \\
&+ \frac{\partial}{\partial x}\big(hu(\boldsymbol{\sigma} d\boldsymbol{B_t})_x\big) + \frac{\partial}{\partial y}\big(hu(\boldsymbol{\sigma} d\boldsymbol{B_t})_y\big) = -\mathbf{g}h\frac{\partial \eta}{\partial x}dt. \qquad (8.38)
\end{aligned}
$$

Note that the diffusion terms is equivalent to the compact form $\sum_{i,j}\frac{1}{2}\partial_i\partial_j(a_{ij}hu) - a_{ij}\partial_j h\partial_i u$. The momentum equation regarding another velocity component takes a similar form,

$$
\begin{aligned}
&d_t(hv) + \Big(\frac{\partial(huv)}{\partial x} + \frac{\partial(hv^2)}{\partial y} \\
&- \frac{1}{2}\big(a_{xx}\frac{\partial(h\partial_x v)}{\partial x} + a_{yy}\frac{\partial(h\partial_y v)}{\partial y} + 2\partial_x a_{xx}\frac{\partial(hv)}{\partial x} + 2\partial_y a_{yy}\frac{\partial(hv)}{\partial y} + a_{xx}\frac{\partial(v\partial_x h)}{\partial x} + a_{yy}\frac{\partial(v\partial_y h)}{\partial y}\big)\Big)dt \\
&+ \frac{\partial}{\partial x}\big(hv(\boldsymbol{\sigma} d\boldsymbol{B_t})_x\big) + \frac{\partial}{\partial y}\big(hv(\boldsymbol{\sigma} d\boldsymbol{B_t})_y\big) = -\mathbf{g}h\frac{\partial \eta}{\partial y}dt, \qquad (8.39)
\end{aligned}
$$

If we are only interested in the mean ensemble average depth evolution (or imposing that $\boldsymbol{\sigma} d\boldsymbol{B_t}\boldsymbol{\nabla} h = 0$), The above equations can be greatly simplified due to the canceling of the Brownian random terms. Thus, we have,

$$
\frac{\partial \bar{h}}{\partial t} + \frac{\partial(\bar{h}u)}{\partial x} + \frac{\partial(\bar{h}v)}{\partial y} - \frac{1}{2}\sum_{(i,j)^h}\partial_{x_i}\partial_{x_j}(a_{ij}\bar{h}) = 0, \qquad (8.40a)
$$

$$
\begin{aligned}
&\frac{\partial(\bar{h}u)}{\partial t} + \frac{\partial(\bar{h}u^2)}{\partial x} + \frac{\partial(\bar{h}uv)}{\partial y} \\
&- \frac{1}{2}\big(a_{xx}\frac{\partial(\bar{h}\partial_x u)}{\partial x} + a_{yy}\frac{\partial(\bar{h}\partial_y u)}{\partial y} + 2\partial_x a_{xx}\frac{\partial(\bar{h}u)}{\partial x} + 2\partial_y a_{yy}\frac{\partial(\bar{h}u)}{\partial y} + a_{xx}\frac{\partial(u\partial_x \bar{h})}{\partial x} + a_{yy}\frac{\partial(u\partial_y \bar{h})}{\partial y}\big) \\
&= -\mathbf{g}h\frac{\partial \bar{\eta}}{\partial x}, \qquad (8.40b)
\end{aligned}
$$

$$
\begin{aligned}
&\frac{\partial(\bar{h}v)}{\partial t} + \frac{\partial(\bar{h}uv)}{\partial x} + \frac{\partial(\bar{h}v^2)}{\partial y} \\
&- \frac{1}{2}\big(a_{xx}\frac{\partial(\bar{h}\partial_x v)}{\partial x} + a_{yy}\frac{\partial(\bar{h}\partial_y v)}{\partial y} + 2\partial_x a_{xx}\frac{\partial(\bar{h}v)}{\partial x} + 2\partial_y a_{yy}\frac{\partial(\bar{h}v)}{\partial y} + a_{xx}\frac{\partial(v\partial_x \bar{h})}{\partial x} + a_{yy}\frac{\partial(v\partial_y \bar{h})}{\partial y}\big) \\
&= -\mathbf{g}h\frac{\partial \bar{\eta}}{\partial y}. \qquad (8.40c)
\end{aligned}
$$

8.3 1D stochastic shallow water equation

Some tests shown in the next chapter are carried out with a 1D stochastic shallow water equation since it is less time-consuming and can reveal the essence attributes associated

with the problem. The 1D stochastic shallow water equation simply reads,

$$d_t h + \Big(\frac{\partial (hu)}{\partial x} - \frac{1}{2}\frac{\partial^2 (a_{xx} h)}{\partial x^2}\Big)dt + \frac{\partial h}{\partial x}(\sigma d\boldsymbol{B_t})_x = 0. \tag{8.41a}$$

$$d_t(hu) + \Big(\frac{\partial (hu^2 + \frac{1}{2}gh^2)}{\partial x} - \frac{1}{2}\big(a_{xx}\frac{\partial (hu_x)}{\partial x} + 2\partial_x a_{xx}\frac{\partial (hu)}{\partial x} + a_{xx}\frac{\partial (uh_x)}{\partial x}\big)\Big)dt + u\frac{\partial h}{\partial x}(\sigma d\boldsymbol{B_t})_x = 0, \tag{8.41b}$$

$$d_t(hv) + \Big(\frac{\partial (huv)}{\partial x} - \frac{1}{2}\Big(a_{xx}\frac{\partial (hv_x)}{\partial x} + 2\partial_x a_{xx}\frac{\partial (hv)}{\partial x} + a_{xx}\frac{\partial (vh_x)}{\partial x}\big)\Big)dt + v\frac{\partial h}{\partial x}(\sigma d\boldsymbol{B_t})_x = 0. \tag{8.41c}$$

and the mean depth and its associated horizontal fields satisfy,

$$\frac{\partial h}{\partial t} + \frac{\partial (hu)}{\partial x} - \frac{1}{2}\frac{\partial^2 (a_{xx} h)}{\partial x^2} = 0. \tag{8.42a}$$

$$\frac{\partial hu}{\partial t} + \frac{\partial (hu^2 + \frac{1}{2}gh^2)}{\partial x} - \frac{1}{2}\big(a_{xx}\frac{\partial (hu_x)}{\partial x} + 2\partial_x a_{xx}\frac{\partial (hu)}{\partial x} + a_{xx}\frac{\partial (uh_x)}{\partial x}\big) = 0, \tag{8.42b}$$

$$\frac{\partial hv}{\partial t} + \frac{\partial (huv)}{\partial x} - \frac{1}{2}\Big(a_{xx}\frac{\partial (hv_x)}{\partial x} + 2\partial_x a_{xx}\frac{\partial (hv)}{\partial x} + a_{xx}\frac{\partial (vh_x)}{\partial x}\big) = 0. \tag{8.42c}$$

8.4 Summary

In this chapter, we started by the motivation of a stochastic modeling, and we briefly reviewed the extended stochastic version of the Reynolds transport theorem proposed in Mémin (2014). A complete derivation of the shallow water equation under uncertainty has been presented as well.

Chapter 9

Ensemble-based Parameter Estimation Scheme

In the previous chapter, we demonstrated how the stochastic 2D shallow water equation can be formulated on the basis of a decomposition of the unknown fluid particles displacement fields. The decomposition process is built in terms of a resolved part and an unresolved part modeled as a stochastic process. Such a formalism allows us to obtain the 2D shallow water equation in the forms of (8.13) and (8.41). We can observe that the dissipative terms emerged both in the continuity equation and the momentum conservation equation. It is similar in spirit to the eddy viscosity model used in LES and RANS. Nevertheless, the eddy viscosity model is built on several assumptions that will be discussed in the next section whereas our subgrid model, originated from the stochastic representation of the uncertainty or errors associated to the model itself. No extra assumption is needed. However, their similarities indeed bring us another perspective for the interpretation of the subgrid scale effect.

9.1 Eddy viscosity and Smagorinsky subgrid model

One simple method for the modeling of the subgrid process is the eddy viscosity model approach. This approach formulates the subgrid scale motion in the same way as the molecular diffusion,

$$\tau'_{ij} = 2\mu_t(S_{ij} - \frac{1}{3}S_{kk}\delta_{ij})^1,$$

where μ_t denotes the eddy dynamic viscosity coefficient, which yield the eddy kinematic viscosity coefficient ν_t when divided by ρ. Note that τ' is different from τ, the deviatoric tensor aforementioned. In terms of incompressible flow, $S_{kk} = 0$, and S_{ij} is the rate of the strain tensor. In a 3D case,

$$S_{ij} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}) & \frac{1}{2}(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}) \\ \frac{1}{2}(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}) & \frac{\partial v}{\partial y} & \frac{1}{2}(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}) \\ \frac{1}{2}(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}) & \frac{1}{2}(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}) & \frac{\partial w}{\partial z} \end{bmatrix}.$$

Note that in terms of geophysical flows, the horizontal and vertical subgrid effects are quite different. Usually the horizontal eddy viscosity coefficient is set much larger than

[1] Einstein double indice summarize convention

the vertical eddy viscosity in order to respect the physics. Consequently we focus on the modeling of horizontal eddy coefficient ν_h. It is based on Boussinesq assumption proposed by Smagorinsky (1963),

$$\nu_h = C\Delta_x\Delta_y\sqrt{(\frac{\partial u}{\partial x})^2 + (\frac{\partial v}{\partial y})^2 + \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})^2}, \tag{9.1}$$

where C is a constant, Δx and Δy are the space grid size, the square root term corresponds to the Frobenius norm of the rate of strain tensor S_{ij}. Considering the continuity equation, we have the standard trace free Smagorinsky subgrid stress $\tau' = \rho\nu_t S_{ij}$. The divergence of this stress corresponds to the following diffusive terms,

$$\boldsymbol{\nabla}\cdot\tau' = \rho\begin{bmatrix} \frac{\partial}{\partial x}(\nu_h\frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(\nu_h\frac{\partial u}{\partial y}) \\ \frac{\partial}{\partial x}(\nu_h\frac{\partial v}{\partial x}) + \frac{\partial}{\partial y}(\nu_h\frac{\partial v}{\partial y}) \\ \frac{\partial}{\partial x}(\nu_h\frac{\partial w}{\partial x}) + \frac{\partial}{\partial y}(\nu_h\frac{\partial w}{\partial y}) \end{bmatrix}.$$

Note that here we ignored the vertical subgrid-scale terms.

If we compare the x-direction momentum conservation equation of (8.40) to the deterministic 2D nonlinear shallow water considering this eddy viscosity model, we can see the relationship between the quadratic variation tensor a_{ij} and the eddy kinematic viscosity coefficient ν_h. Indeed, if we prescribe the quadratic variation tensor equals to the eddy viscosity coefficient:

$$a_{ij} = \nu_h\mathbb{I}. \tag{9.2}$$

The uncertainty stress tensor in a 2D case can be expand as,

$$\sum_{ij}\frac{\partial^2}{\partial x_i\partial x_j}(\nu_h u) = \nu_h\Delta u + \boldsymbol{\nabla}^2\nu_h u + 2\partial_x\nu_h\partial_x u + 2\partial_y\nu_h\partial_y u.$$

Compared to the eddy viscosity model, we have extra terms as $\boldsymbol{\nabla}^2\nu_h u + \partial_x\nu_h\partial_x u + \partial_y\nu_h\partial_y u$. A standard trace free Smagorinsky subgrid tensor $\boldsymbol{\nabla}\cdot\boldsymbol{\tau} = \boldsymbol{\nabla}\cdot(C\|S_{ij}\|S_{ij})$, contains however a term $2\sum_j\partial_{x_j}\|S_{ij}\|\partial_{x_k}w^j$ in which $j = 1, 2, 3$ and $k = 1$. Consequently, the relationship between our uncertainty tensor and Smagorinsky subgrid tensor is far more complex. Nevertheless, if the rate of the strain tensor norm is rather flat, those two models can have the same forms.

As stated before, one main incentive of our approach is to find a better way of exploring the fine resolution image data. An interesting idea is to estimating the quadratic variation tensor $\boldsymbol{a}$ or the diffusion tensor $\boldsymbol{\sigma}$ from the statistical variations of small scale velocity component embedded in the data. The relation between $\boldsymbol{\sigma}$ and $\boldsymbol{a}$ can be found through

$$a_{ij} = \sigma_{ik}\sigma_{jk}. \tag{9.3}$$

Let us note that $\boldsymbol{a}$ is a block diagonal tensor.

In the following chapters, we will restrain our interests in the horizontal mean flow fields described in (8.40) and (9.27). But considering the stochastic driven term in models into DA system is also applicable (Miller et al., 1999).

9.2 Estimation of the quadratic variation tensor from data

In the following we explore several strategies to estimate the quadratic variation tensor on the basis of the realized measurement variance.

9.2.1 Estimator as a diagonal projection operator

Let us first make the assumption that the uncertainty term lies on the isoheight surface. This assumption is based on the fact that in the applications of oceanography the fluid is stratified and the effect of uncertainty mainly lies in the horizontal scale thus the tangent height plane rather than the height gradient direction. The diffusion tensor is defined:

$$\sigma_{ij} = \alpha(\delta_{ij} - \frac{\partial_{x_i} h(x)\partial_{x_j} h(y)}{\parallel \nabla h \parallel_2^2})\delta(x-y),$$

where α can be arbitrary fixed to maintain the consistency of unit. For sake of simplicity, we can fix α invariant over space and time, and set $\alpha = \sqrt{dxdy/dt}$. An interesting implication of this assumption is the cancellation of Brownian motion terms, in other words, it provides the same system as the one associated to the mean depth fields.

9.2.2 Estimator from realized temporal/spatial variance

A basic method is to consider the realized temporal variance. Suppose we have a sequence of small-scale velocity data ranging form 0 to T at the uniformly distributed times where $T = (M+1)\Delta t$,

$$w_0, w_1, \ldots, w_M,$$

which are observed value of $\boldsymbol{U}$. A discrete-time approximation process of Eq.8.2 reads,

$$\boldsymbol{U}_i = \mathbf{w}_i + \boldsymbol{\sigma}\eta_i/\sqrt{\Delta t}, \tag{9.4}$$

where η_i follows normal distribution. The second moment of random variable $\boldsymbol{U}$ can be approximated by its observed values,

$$\sum_{i=0}^{T} \boldsymbol{a} = \Delta t \sum_{i=0}^{T} w(t_i)w(t_i)^T. \tag{9.5}$$

Note here $\boldsymbol{a}$ is fixed along time interval $[0, T]$. This realized variance needs to be multiplied by a model constant (Kadri Harouna and Mémin, 2014),

$$\boldsymbol{a} = C\frac{(\Delta t)^2}{T - \Delta t}\sum_{i=0}^{T} w(t_i)w(t_i)^T, \tag{9.6}$$

where C is a scaling factor given by,

$$C = \Big(\frac{L}{l}\Big)^{5/3}, \tag{9.7}$$

where L is the scale of large-scale models and l is the smallest scale considered in the Kolmogorov-Ridcharson scaling.

If we intend to benefit from the high-resolution observation given that the model is running on a much coarser grid. We can consider the local spatial variance of a small domain $\mathcal{C}$ on observation space localized around the corresponding grid point on model space. Denoting k as the model grid index, l as a set of observation grid index belonging to $\mathcal{C}$ and n the sum of observation within $\mathcal{C}$,

$$\boldsymbol{a}(t_i, x_k) = C\frac{\delta t}{n^2 - 1}\sum_{l\in\mathcal{C}} \big[(w(t_i, x_l) - \bar{w}(t_i, x_{\mathcal{C}}))(w(t_i, x_l) - \bar{w}(t_i, x_{\mathcal{C}}))^T\big], \tag{9.8}$$

where $\bar{w}$ is local mean velocity,

$$\bar{w}(t_i, x_{\mathbf{C}}) = \frac{1}{n} \sum_{l \in \mathcal{C}} w(t_i, x_l), \tag{9.9}$$

and δt is the characteristic time step used to regularize the unit of $\boldsymbol{a}$ ($m^2 s^{-1}$). The spatial local quadratic variation calculated here is a function both in time and space. Therefore, it needs to be evaluated at each observation time level. Note that either spatial or temporal empirical variance approaches can be used.

9.2.3 Estimator from realized ensemble variance

The ensemble forecast approach not only gives the deterministic mean state, but also allows the quantitive evaluation of the error variance associated to the forecast state. Based on this idea, we can estimate the quadratic variation tensor from the ensemble space as the background error covariance evaluation is estimated in chapter 5. Imagine we have an ensemble of N members,

$$\boldsymbol{a} = C \frac{(\Delta t_i)^2}{T} \sum_{i=0}^{T} \frac{1}{N-1} \sum_{j=1}^{N} [w^j(t_i) - \bar{w}(t_i)][w^j(t_i) - \bar{w}(t_i)]^T \tag{9.10}$$

Here, $\bar{w}$ stands for the ensemble mean velocity fields.

9.3 Estimation of the quadratic variation tensor from DA process

The previous estimation methods are easy to implement, but these estimations constitute only rough approximations as they are estimated either from the resolved component or from high observations. A more accurate way of estimating the parameters is through DA process. As almost all parameterization are due to the imperfectness of the dynamic model. This imperfectness is often interpreted as model errors. The approach of weak constrain formalism in the variational framework can surely serve this purpose and would worth exploring.

Another way consists to consider the unknown parameter as a control parameter in the variational data assimilation system as proposed by Navon (1998). The basic idea is to consider the quadratic variation tensor a_{ij} as a parameter of the model and to employ an augmented control vector containing both the state variables and parameters. Finally, the object of the DA system is to find the optimal initial condition and the associated optimal parameters for which a system trajectory fits at best to the observations.

In our approach, the cost function can be modified as

$$J(\delta \mathbf{x_0}, \delta \boldsymbol{a}) = \frac{1}{2} \|\delta \mathbf{x_0}(x)\|^2_{\mathbf{B_x}} + \frac{1}{2} \|\delta \boldsymbol{a}\|^2_{\mathbf{B_a}} + \frac{1}{2} \int_{t_0}^{t_f} \|\partial_{\mathbf{x}} \mathbb{H}\, \delta \mathbf{x}(t, x, \boldsymbol{a}) - \mathbf{D}(t, x, \boldsymbol{a})\|^2_{\mathbf{R}} dt, \tag{9.11}$$

where the innovation vector $\mathbf{D}(t, x, \boldsymbol{a})$ is defined as:

$$\mathbf{D}(t, x, \boldsymbol{a}) = \mathcal{Y}(t, x) - \mathbb{H}(\varphi_t(\mathbf{x_0}(x), \boldsymbol{a})), \tag{9.12}$$

and the evolution of increment $\delta \mathbf{x}(t, x, \boldsymbol{a})$ is,

$$\delta \mathbf{x}(t, x, \boldsymbol{a}) = \partial_{\mathbf{x}} \varphi_t(\mathbf{x_0}) \delta \mathbf{x_0}.$$

In this formalism, the second term of the RHS of Eq.(9.11) representing the discrepancy with respect to the background parameters. It acts as a constraint on the parameters. It is often absent when initializing the assimilation procedure, which means that no background parameter is prescribed and the inferred parameters will comply to the observation. In most cases, the model parameters are not observed, so only the observation of the state variables (complete or partial) guarantees the generation of the full analysis increments which minimize Eq.(9.11).

To solve such a problem, a direct application of parameter estimation scheme based on section 2.1.2 is straightforward. Several literatures have been already addressed this issue and its applications in various contexts (Navon, 1998; Zhu and Navon, 1999; Pulido and Thuburn, 2005; Kazantsev, 2012). Nevertheless, the gradient evaluation of the cost function versus parameters depends on the model sensitivity to the parameters, but also on the adjoint variable Λ and the adjoint model $\partial_a \mathbb{M}^*$ (Eq.2.28). Thus the gradient still needs to be calculated built on the adjoint models. This drives us back to a calculation, which we wanted to avoid with our ensemble technique. This is the reason why we shift to an ensemble-based framework to estimate the parameters.

9.3.1 Ensemble-based parameter estimation

Estimating parameter through ensemble-based methods has its advantages and limits. For one thing, the adjoint or the tangent linear model is not needed. In this method, the parameters can be estimated either simultaneously through the same ensemble along with the initial condition or independently. A compete review on this topic can be found in Ruiz et al. (2013a).

The former approach is a direct extension of standard ensemble methods. Most studies have followed this path (Zupanski and Zupanski, 2006; Tong and Xue, 2008; Yang and Delsole, 2009; Kang et al., 2011; Ruiz et al., 2013a). Tong and Xue (2008) used the ensemble square root Kalman filter (EnSRF) to estimate the microphysical parameters. Yang and Delsole (2009) showed the possibility of estimating multiplicative parameters with ensemble Kalman filter. LETKF is also used in Kang et al. (2011) to estimate the carbon flux. Bocquet and Sakov (2013b) devised an iterative ensemble Kalman smoother (IEnKS) which relies on an iterative minimization of a cost function built on ensemble space. Both LETKF and IEnKS allow us to perform the assimilation over a relatively long period containing asynchronous observations.

Koyama and Watanabe (2010) proposed to separate the assimilation process for initial state and parameters estimations. This separation is done through two ensembles: one initialized by both an initial condition and some parameters perturbations, the other ensemble accounting for a temporal averaged state over a relatively long period of time and initialized only by a parameters' perturbation. Such a separation, according to the authors, overcomes the disadvantage of a reduced accuracy of the analysis using simultaneous estimations. Ruiz et al. (2013b) argued as well that using separate ensembles induced either by initial condition or parameter perturbations can better represent the error covariance.

In our approach, we choose to work with the simultaneous approach. The reason is that in our setup we focus on coarse resolution simulation with high resolution data, and the coarse initial state grids occupy exactly those locations of high-resolution observation. Consequently, the error associated with the initial condition is a bit difficult to determine.

This approach is also related to the estimation of bias (Dee, 2005; Baek et al., 2006) from DA approach. Model bias is a specific form of model error. The bias estimation process

treats the bias as an augmented variable. In fact our ensemble-based parameter estimation approach is closer to the bias estimation rather than typical parameter estimation algorithm owing to the similar size of our quadratic variation tensor $\boldsymbol{a}$ and the bias compared to the state variables.

Parameter structures

It is essential to realize that the parameters involved in geophysical models can be global or local in time and/or space. The parameters are generally time-dependent and they are subject to change on the basis of the new observation. However, when the parameters are inferred by the data assimilation process, the temporal density of the parameters is at most equal to those of the observations. This suggests that the parameters being estimated should only show slow temporal variation.

In terms of the spatial structure, both global and local parameters are very common. Due to its spatial structure, it may be necessary to introduce the localization to both the auto-covariance matrix of the parameters and the cross-covariance matrix of the parameters and the state variables.

In our case, the quadratic variation tensor $\boldsymbol{a}$ is a function of both time and space. However, in variational approaches, it is custom to assume a constant value of $\boldsymbol{a}$ during the assimilation window. So in order to illustrate the temporal evolution of $\boldsymbol{a}$, it is necessary to introduce cycling procedure. At the end of the jth assimilation window, the analysis state is evolved to the beginning of the $j+1$th cycle with the optimum parameter $\boldsymbol{a}_j^a$:

$$\mathbf{x}_{j+1}^f = \varphi(\mathbf{x}_j^a, \boldsymbol{a}_j^a), \tag{9.13}$$

Parameter evolution model

Besides the state evolution model, we also need to propagate the estimated parameter to the next cycle in someway. This can be generally classified as parameter model in Wikle and Berliner (2007). A persistent (or stationary) model is often used (Baek et al., 2006). This can be simply put as

$$\boldsymbol{a}_{j+1}^f = \boldsymbol{a}_j^a. \tag{9.14}$$

However, the persistent model can be a source of instability to the model integration with limited ensemble numbers in some cases. Gottwald and Majda (2013) explains this instability by the contribution of the unrealistic large covariance terms. The premise for such a conclusion is made upon sparse observations' distributions and small observations' noise.

A similar phenomenon called model blow-up is discussed in Yang and Delsole (2009), the authors proposed a relaxation parameter evolution model to damp the rapid temporal change of model parameter. Here we adopt the same idea,

$$\boldsymbol{a}_{j+1}^f = \boldsymbol{a}_j^f + \beta\widehat{\delta\boldsymbol{a}_j}, \tag{9.15}$$

where β is a damping coefficient. Note that Yang and Delsole (2009) employed $1-\beta$ in equation (19). The value of β is optimally tuned to fall into the interval $(0.2, 0.5)$. Note that in our ensemble-based 4DVar approach listed in chapter 5, the parameter evolution is not only related to the forecast-analysis cycling process, but also can affect the ensemble fields generated between two consecutive outer loops.

Core algorithm

First we need to define $\mathbf{s}_0$ as the augmented state vector $[\mathbf{x}_0, \boldsymbol{a}_0]$ that has dimension $n+p$ instead of n where p is the degree of freedom of $\boldsymbol{a}$. Note that we leave the initial values of $\boldsymbol{a}_0$ as zero. Then the state initial ensemble can be defined as $\mathbf{S}_0$ in which the jth augmented state ensemble reads $\mathbf{S}_0^j = [\mathbf{x}_0^j, \boldsymbol{a}^j]$. Both the initial state and parameter are perturbed according to their uncertainties. As it is already cumbersome to determine the uncertainty of the background state above, the addition of the parameter uncertainty obviously worse. In our application, we assume the initial parameter error follows a Gaussian distribution whose variance has to be tuned. Remark that the second type ensemble is only a function of the parameters. That is to say the only source that contributes to this ensemble spread is the parameter perturbation.

Such perturbations yield an ensemble perturbation matrix,

$$\mathbf{A}'_s := \frac{1}{\sqrt{N-1}}(\mathbf{S}_0^1 - \langle \mathbf{S}_0 \rangle, \ldots, \mathbf{S}_0^N - \langle \mathbf{S}_0 \rangle). \tag{9.16}$$

Note that $\mathbf{A}'_s \in \mathbb{R}^{(n+p)\times N}$.

If we consider a new minimization problem in terms of the augmented state increment vector $\delta\mathbf{s}_0$, the cost function reads:

$$J(\delta\mathbf{s}_0) = \frac{1}{2}\|\delta\mathbf{s}_0\|^2_{\mathbf{B}_\mathbf{s}^f} + \frac{1}{2}\int_{t_0}^{t_f} \|\widetilde{\partial_\mathbf{x}\mathbb{H}}\, \partial_\mathbf{x}\varphi'_t(\mathbf{s}_0)\delta\mathbf{s}_0 - \mathbf{D}(t,x,\boldsymbol{a})\|^2_\mathbf{R} dt, \tag{9.17}$$

with

$$\mathbf{B}_\mathbf{s}^f = \mathbf{A}'_s{\mathbf{A}'_s}^T = \begin{pmatrix} \mathbf{B}_{xx} & \mathbf{B}_{ax}^T \\ \mathbf{B}_{ax} & \mathbf{B}_{aa} \end{pmatrix}. \tag{9.18}$$

$\mathbf{B}_{xx}$ corresponds to the covariance matrix of state vector $\mathbf{x}$, $\mathbf{B}_{aa}$ is the covariance matrix of parameter vector $\boldsymbol{a}$, and $\mathbf{B}_{ax}$ contains the cross covariance between the state vector and the parameter vector.

We must clarify two points in terms of the cost function (9.17). Firstly, the tangent linear model operator $\varphi'_t (\neq \varphi_t)$ is composed of the parameter evolution model as well as the tangent linear dynamic model operator φ_t. Secondly, in a direct observation case, the tangent observation operator should equal to the identity matrix, but if $\boldsymbol{a}$ is not observed, the tangent observation operator $\widetilde{\partial_\mathbf{x}\mathbb{H}}$ is,

$$\widetilde{\partial_\mathbf{x}\mathbb{H}} = \begin{pmatrix} \mathbb{I}_{m\times n} \\ \mathbf{0}_{m\times p} \end{pmatrix},$$

where m is the size of observation vector. In our case, we can use the direct estimation of $\boldsymbol{a}$ from data as its observed value. Hence the tangent observation operator remains an identity matrix. Those observed parameters are proved to be a good approximation and prevent the system from blowing up.

Preconditioning the above cost function by $\delta\mathbf{s}_0 = \mathbf{A}'_s\delta\mathbf{z}_0$ yields,

$$J(\delta\mathbf{z}_0) = \frac{1}{2}\|\delta\mathbf{z}_0\|^2 + \frac{1}{2}\int_{t_0}^{t_f} \|\widetilde{\partial_\mathbf{x}\mathbb{H}}\, \partial_\mathbf{x}\varphi'_t(\mathbf{s}_0)\mathbf{A}'_s\delta\mathbf{z}_0 - \mathbf{D}(t,x,\boldsymbol{a})\|^2_\mathbf{R} dt. \tag{9.19}$$

The gradient reads,

$$\partial_{\delta\mathbf{z}_0} J = \delta\mathbf{z}_0 + \int_{t_0}^{t_f} \widetilde{\mathbf{B}}_t^{\frac{1}{2}T} \left(\widetilde{\partial_\mathbf{x}\mathbb{H}}\right)^* \mathbf{R}^{-1}(\widetilde{\partial_\mathbf{x}\mathbb{H}}\, \widetilde{\mathbf{B}}_t^{\frac{1}{2}}\delta\mathbf{z}_0 - \mathbf{D}(t,x))dt. \tag{9.20}$$

and $\widetilde{\mathbf{B}}_t^{\frac{1}{2}}$ is defined through,

$$\widetilde{\mathbf{B}}_t^{\frac{1}{2}} = \partial_{\mathbf{x}} \varphi\,'_t(\mathbf{s}_0)\mathbf{A}'_s. \tag{9.21}$$

As discussed in chapter 5, the empirical description of $\widetilde{\mathbf{B}}_t^{\frac{1}{2}}$ from a set of samples is based on an empirical expression of the ensemble members integrated in time. The integration of these samples along the assimilation window thus yield a 'cloud' of particles, from which we can estimate the low-rank approximation of the background covariance trajectory and of its square root. The process can be put as,

$$\partial_{\mathbf{x}} \varphi\,'_t(\mathbf{s}_0)\mathbf{A}'_s \approx \varphi_t(\mathbf{S}_0)(\mathbf{I}_N - \frac{\mathbf{1}\mathbf{1}^T}{N}), \tag{9.22}$$

where $\mathbf{1} = (1, \ldots, 1)^T \in \mathbb{R}^N$.

Once we have the optimizer $\widehat{\delta\mathbf{z}_0}$, the analysis increment can be written as,

$$\widehat{\delta\mathbf{s}_0} = \begin{bmatrix} \widehat{\delta\mathbf{x}_0} \\ \widehat{\delta\boldsymbol{a}_0} \end{bmatrix} = \mathbf{A}'_s \widehat{\delta\mathbf{z}_0}. \tag{9.23}$$

Parameter estimation effect

It should be emphasized that although the estimation of the initial condition and the parameter is done simultaneously. The estimation process of the initial condition is nevertheless not directly related to the auxiliary covariance terms $(\mathbf{B}_{aa}, \mathbf{B}_{ax})$ associated with the parameter $\boldsymbol{a}$. The parameter $\boldsymbol{a}$, however, interferes in the quality of the analysis trajectory in two ways: firstly, it explicitly manifests itself on the model integration in the form of the estimated parameter $\widehat{\boldsymbol{a}}$; secondly, it implicitly affects the analysis state through the ensemble spread from which the propagation of the ensemble perturbation matrix is calculated. Koyama and Watanabe (2010) suggests the second factor is more effective than the former.

Localization technique

The parameters that we are trying to optimize is indeed a function of space. It is important to note that the space parameter needs to be localized as well. As a matter of fact, we find in our applications that adequate localization can alleviate the model blow-up problem observed with too small ensemble size. We recall that in chapter 6, the localization can result in more pertinent ensemble-based covariance since it is less subject to sampling errors. The same effects can also be found from the strategy of increasing ensemble size.

The localization technique applied here is derived from the ones discussed in section 5.1.4. Two possible approaches, localized covariance and local ensemble, can be pursued here with few modifications.

The localized covariance approach has to be extended to the spatial parameter covariance and the cross-covariance between the state variable and the parameter. That is to say, we use

$$\mathbf{P}_s^f = \mathcal{C}_s \odot \mathbf{B}_s^f = \begin{pmatrix} \mathcal{C}_{xx} & \mathcal{C}_{ax}^T \\ \mathcal{C}_{ax} & \mathcal{C}_{aa} \end{pmatrix} \odot \begin{pmatrix} \mathbf{B}_{xx} & \mathbf{B}_{ax}^T \\ \mathbf{B}_{ax} & \mathbf{B}_{aa} \end{pmatrix}, \tag{9.24}$$

to filter out the long range spatial correlations. The spatial correlation matrix $\mathcal{C}_{xx}$ versus state variable is identical to $\mathcal{C}$ used in Eq. (5.27). Such a spatial correlation matrix can set

all correlations to zeros when the distance between two points exceeds the cut-off distance. It is, however, not necessary to introduce the same cut-off distance for the state variables and the parameters, although in our application, the same optimal tuned value of cut-off distance is used.

As for the path of local ensemble, no adjustment is needed.

9.3.2 Parameter identifiability

Before we carry out the experiments, it is important for us to assess the parameter identifiability. This is especially pertinent for the case where the quadratic variation tensor $\boldsymbol{a}$ is non-observable. This concept, following the definition in Navon (1998), can be viewed as a criteria to decide whether or not the parameter of interest can be inferred from data. The evaluation of this attribute is related to an output response function reflecting the sensitivity of the model to the parameters. Such an response function usually assumes the form (Tong and Xue, 2008):

$$J_y(\boldsymbol{a}) = \int_0^T \|\mathcal{Y}(t,x) - \mathbb{H}(\varphi_t(\mathbf{x_0}(x), \boldsymbol{a}))\|^2 dt. \tag{9.25}$$

This function shares the same form as the observation term of the non-incremental 4DVar cost function. Such a function must be convex and continuous to ensure the uniqueness of the identifiability of the parameters. This condition is implicitly satisfied in our system.

With an ensemble-based method, an alternative way of examining the parameter identifiability is to calculate the correlation coefficient (Tong and Xue, 2008).

$$cor(\mathbf{x}, \boldsymbol{a}) = \frac{cov(\mathbf{x}, \boldsymbol{a})}{\sqrt{var(\mathbf{x})var(\boldsymbol{a})}}. \tag{9.26}$$

A strong correlation suggests that any changes in the parameter space will heavily affect the state space through the cross-covariance part of the error covariance matrix. The principle is actually similar to the case where we estimate the unobserved components of the state variables.

9.4 Model and experimental settings

To evaluate the methods proposed, we have first carried out experiments on a one-dimensional shallow water model. This model is described in the following section.

9.4.1 Model numerical scheme

To apply the finite volume methods used in chapter 6, the deterministic 1D shallow water equation under uncertainty in terms of movement quantities (h, hu, hv) can be readily

written as follows:

$$\frac{\partial h}{\partial t} + \frac{\partial (hu)}{\partial x} - \frac{1}{2}\frac{\partial^2 (a_{xx} h)}{\partial x^2} = 0, \tag{9.27a}$$

$$\frac{\partial (hu)}{\partial t} + \frac{\partial (hu^2 + \frac{1}{2} g h^2)}{\partial x} - \frac{1}{2}\Big(a_{xx}\frac{\partial (hu_x)}{\partial x} + 2\partial_x a_{xx}\frac{\partial (hu)}{\partial x} + a_{xx}\frac{\partial (uh_x)}{\partial x}\Big) = -ghS_{bx} + hfv, \tag{9.27b}$$

$$\frac{\partial (hv)}{\partial t} + \frac{\partial (huv)}{\partial x} - \frac{1}{2}\Big(a_{xx}\frac{\partial (hv_x)}{\partial x} + 2\partial_x a_{xx}\frac{\partial (hv)}{\partial x} + a_{xx}\frac{\partial (vh_x)}{\partial x}\Big) = -ghS_{by} - hfu. \tag{9.27c}$$

This system corresponds to the large scale evolution of an average depth. Remark that the subgrid model associated with the quadratic tensor $\boldsymbol{a}$ emerges both in the continuity equation and the momentum conservation equation. The finite volume method discussed in section (6.2.1) is directly applicable to this equations. As a matter of fact, the numerical flux formulation of the new subgrid terms is analog to the viscous terms defined in (6.28).

9.4.2 Experimental settings

Several experiments have been conducted to evaluate the performance of the stochastic model. We first test the 1D stochastic shallow water equation. The 1D domain is of length $L = 6000$ km with the initial surface height $h(x, 0)$.

$$h(x, 0) = H_0 - \frac{f U_0 x}{g} + A\xi, \tag{9.28}$$

where $H_0 = 5000$m, $f = 1.03 \times 10^{-4}\text{s}^{-1}$ is the Coriolis parameter, $U_0 = 40$m/s and g is the gravity acceleration. An additional noise is considered. $A = 10$ is its amplitude. And ξ is a random Gaussian covariance field (with de-correlation length equals to 20%L). The initial velocity field is inferred from the geostrophic relation. The true state used to construct the fine observation is obtained by integrating the standard shallow water equation with the initial condition (9.28) on 401 grid points. The large-scale state with the same initial condition is simulated with the stochastic model on only 101 gird points. The time step Δt is set to 37.5s for the true state model run and to 150s for the simulated state model run in order to satisfy different CFL condition. The synthetic observation is extracted from the true state every 600s and the assimilation window is set to contain 3 observations. In order to maintain the balance of state variables for ensemble fields, the analysis started at 6000s.

For the 2D case, a very similar setup is implemented. Several differences are needed though. The initial height surface is a 2D field with the extra error term as a 2D random Gaussian field. The simulated state resolution is 21×51 and the observation resolution is 81×401.

For both cases, we tested the cycling windows strategies in order to adapt the temporal variation of parameters. Here the parameters values change at the beginning of every assimilation window. The parameter evolution model is therefore introduced. We introduced 3 windows concerning the 1D case and 5 windows regarding the 2D case.

Note that in these synthetic cases, the coarse gird nodes of the background state used in the DA process belong to the fine grid of synthetic observations.

Finally, we applied this stochastic model to the Kinect-captured image data which are already explored in chapter 7.

We employ different strategies to construct the ensemble in the synthetic case and in the real case. For the former, as there is no initial error associated with the background, the ensemble is initialized as a function of parameters. For the latter, parameters are perturbed in addition to the initial state perturbations to initialize the ensemble. The parameters variances are not easy to determine. In our study, the errors associated with the components of the parameter $\boldsymbol{a}$ are assumed to be zero-mean Gaussian fields, whose variances have to be tuned. Too large variance will lead to model blow-up inevitably while too small variance can hardly make any difference compare to the non-stochastic shallow water model. However, we found experimentally that the variance of the optimum estimated value of $\boldsymbol{a}$ yielded by the DA system is actually not very sensitive of the initial error.

We decide to take the values of the quadratic variation tensor $\boldsymbol{a}$ estimated through the methods described in section (9.2) as the 'observed' values and such values are inserted into the cost function (9.19). This choice is made upon the fact that if $\boldsymbol{a}$ is non-observable and consequently inferred completely from the ensemble-based estimation scheme, the model driven by such diffusive terms exhibits a high instability. The reason for this kind of instability seems to stem from the irregularities of the initial parameter perturbations. It is again related to the filter divergence problem discussed in Gottwald and Majda (2013). As in our case, an empty parameter observation space will lead to unrealistic parameter innovations implied from the ensemble-based error covariance.

For the synthetic data case, we deduce the observation as a combination of spatial and temporal averaged velocity variation according to equation (9.8). For the real data case, since no velocity observation is available, we calculate the observations from the ensemble based on (9.10). If such an observation of parameter is adopted, the error statistics must be imposed. In our applications, the magnitude of the observation is manually prescribed in order to control the variation tendency of the parameter. For example, the estimated parameter is closer to the its observation values when the error is small. On the contrary, it is subject to greater variations if the observation is considered as being less accurate.

9.5 Results and discussions

9.5.1 1D Synthetic Results

In this 1D experiment, before presenting the results, we show the correlation coefficient in figure (9.1) calculated from equation (9.26). The spatial mean of such correlations are $\overline{cor}(h, a_{xx}) = 0.1649$ and $\overline{cor}(u, a_{xx}) = 0.1804$. Such an order of magnitude of the correlation coefficient ensures the identifiability of $\boldsymbol{a}$ through the state variables. The variance and the covariance correspond to the diagonal terms of the full ensemble-based covariance matrix $\mathbf{B}_s$.

The comparison criterion used to evaluate the performance of DA method is still the RMSE equation (7.19). In order to evaluate the effect of $\boldsymbol{a}$ exerted on the model, we set a relatively large observation error to allow significant variation in $\boldsymbol{a}$, ignoring the requirements of stability to some extent. The RMSE results are shown in figures (9.2) and (9.4). In terms of the performance with the model under uncertainty (shown by black lines with suffix 'subgrid-model'), the analysis trajectories shows that too small ensemble members lead to model blow-up while increasing ensemble size allows achieving a considerable

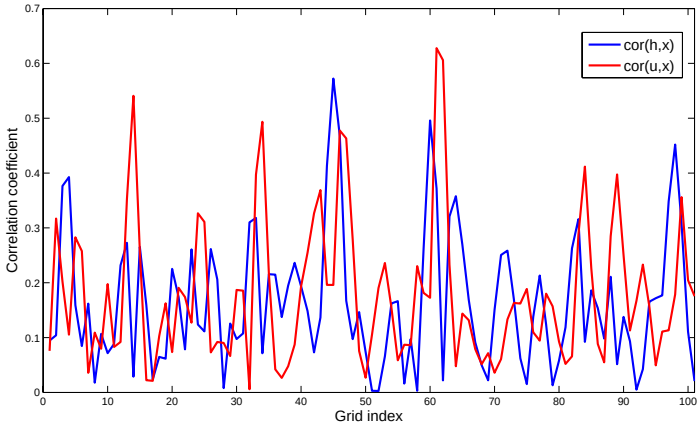


Figure 9.1 – The spatial distribution of the correlation coefficients between the state variable h, u and a: $cor(h, a)$ (blue line) and $cor(u, a)$ (red line).

improvement in the analysis quality in terms of both the initial condition and the parameters. We also find that the localization technique enables improving the result for ensemble of small dimension. We also tested the 'standard-model', which indicates no parameter estimation and the ensemble is generated by perturbing the initial condition state with arbitrary error variance. The results are clearly in favor of our proposed uncertainty model when a sufficient amount of ensemble members or localization technique is employed. We conjecture that such difference actually is related to the ensemble spread. The ensemble spread for both technique is shown in figure 9.3. We observe that the ensemble spread for subgrid model is significantly larger than the spread associated to the standard model. It can be also observed that the ensemble spread is maintained at the same level across the sliding windows.

We have also conducted a study on the different terms composing the subgrid model. It is illustrated in figure (9.4) that the stochastic shallow water model works best with the complete terms presented (denoted as 'a combo 1'). This is also the default case for all the other tests conducted here. We also compared an eddy viscosity model to this subgrid model in which the constant C in equation (9.1) is to be sought from the DA process. Its RMSE curve (denoted by magenta line) in figure 9.4 shows the divergence of the analysis.

In addition to the two configurations, we set two extra cases. Case 'a combo 2' (denoted by red line) does not consider the term associated with the gradient of a_{ij} in the momentum equation. Case 'a combo 3' (denoted by blue line) does not take account of the terms associated with the gradient of a_{ij} in both mass and momentum conservation equations. The different RMSE curves of the two case suggest that the subgrid model term regarding the gradient of a_{ij} in the continuity equation is important. This suggests that the subgrid modeling term presented in the continuity equation is crucial to keep a well estimated model errors and balanced ensemble spread.

9.5.2 2D Synthetic Results

Some results shown in figure (9.5) indicate that this 2D case is obviously much more difficult to handle than the 1D case. Localization is mandatory here. And we tested two localization approaches: LC for localized covariance and LE for local ensemble. We can see that the LE approach needs a large ensemble members to work while the LC approach can

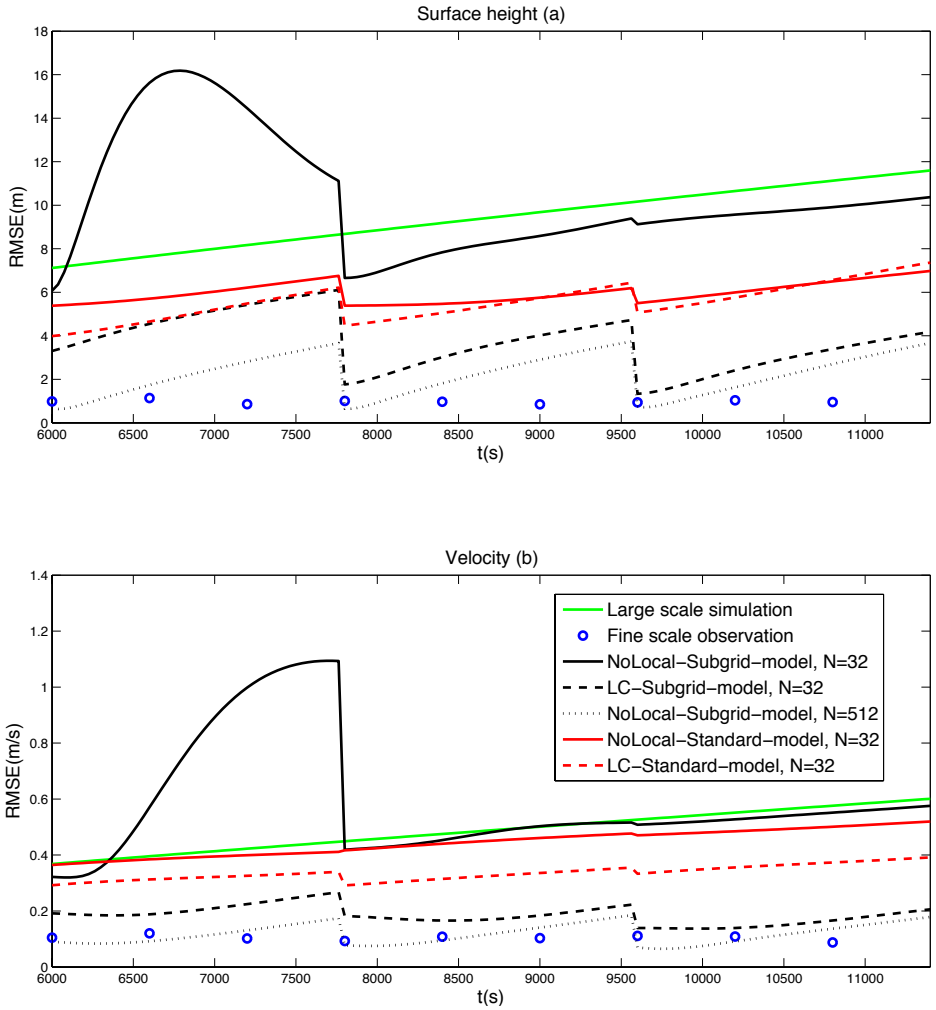


Figure 9.2 – RMSE comparison in terms of free surface height (a) and velocity (b) between various configurations of 4DEnVar. Large scale simulation (green line); fine scale observation (blue points); NoLocal-State: 4DEnVar without localization and with ensemble generated by initial condition perturbation $N = 32$ (red line); LC-State: 4DEnVar with localized covariance and with ensemble generated by initial condition perturbation $N = 32$ (red dashed line); NoLocal-Para: 4DEnVar without localization and with ensemble generated by initial condition perturbation $N = 32$ (black line); LC-Para: 4DEnVar with localized covariance and with ensemble generated by initial condition perturbation $N = 32$ (black dashed line); NoLocal-Para: 4DEnVar without localization and with ensemble generated by parameters perturbation $N = 512$ (black dotted line).

yield comparable result even with a relatively small ensemble. However, the computational cost associated with LC approach of 32 members ensemble is already much more than the cost of LE approach of 512 members. We also notice that the relaxation parameter model is essential to the first cycle. This is because the parameters estimated in the first cycle is not constrained (note that we have a large parameter observation error). Therefore its variation can be dominated by the random error perturbations, which can be very large. However, from the second cycle this effect diminished since $\boldsymbol{a}$ is constrained by its observed values.

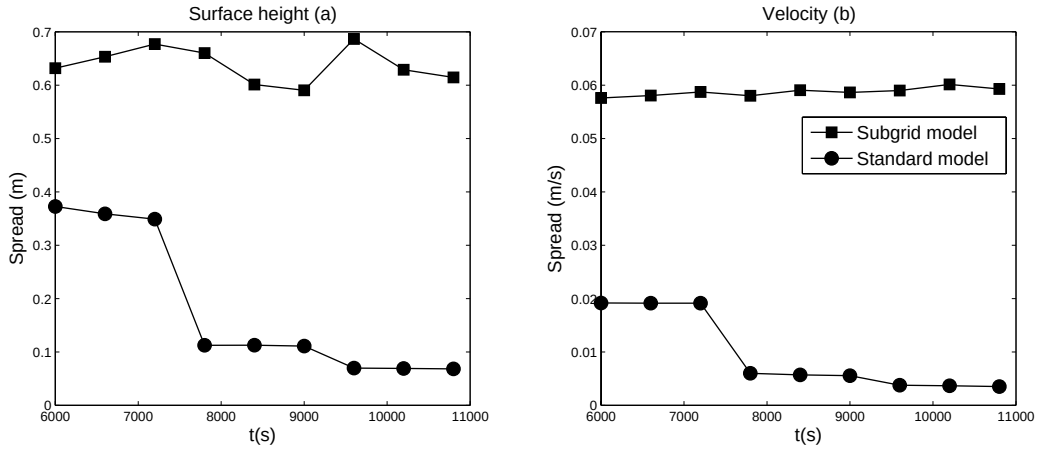


Figure 9.3 – Ensemble spread in terms of free surface height (a) and velocity (b): standard model (circle) with ensemble in function of initial states; subgrid model (square) with ensemble in function of initial parameters.

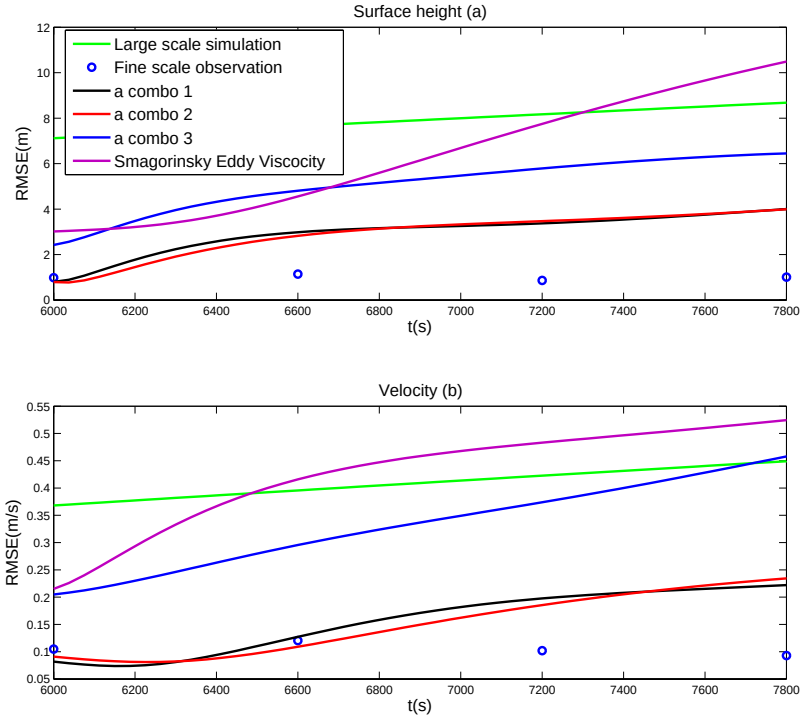


Figure 9.4 – RMSE comparison in terms of free surface height (a) and velocity (b) between various subgrid model configurations of stochastic shallow water model (9.27): Large scale simulation (green line), fine scale observation (blue points), a combo 1 considering full terms (black line), a combo 2 only considering terms associated with a_{xx} in Eq.(9.27b) (red line), a combo 3 only considering terms associated with a_{xx} (blue line), Eddy viscosity model (magenta line).

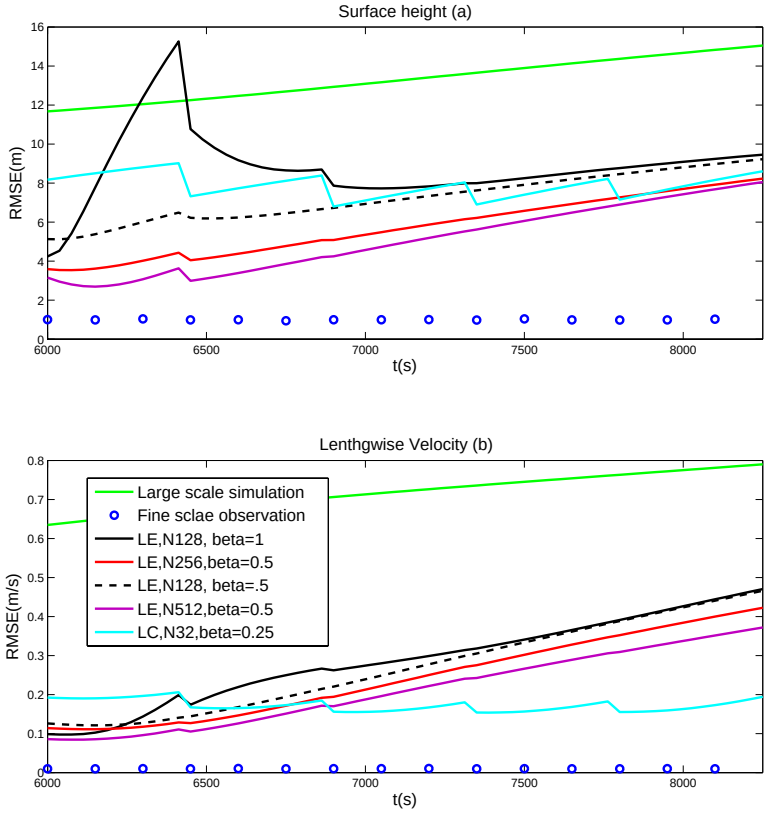


Figure 9.5 – RMSE comparison in terms of free surface height (a) and lengthwise velocity (b) between various configurations of 4DEnVar: Large scale simulation (green line), fine scale observation (blue points), 4DEnVar-LE (local ensemble) with ensemble members $N = 128$ and $\beta = 1$ (black line), 4DEnVar-LE with ensemble members $N = 128$ and $\beta = 0.5$ (black dashed line), 4DEnVar-LE with ensemble members $N = 256$ and $\beta = 0.5$ (red line), 4DEnVar-LE with ensemble members $N = 512$ and $\beta = 0.5$ (magenta line), 4DEnVar-LC (localized covariance) with ensemble members $N = 32$ and $\beta = 0.25$ (cyan line).

9.5.3 Results related to real Kinect-captured data

In terms of the real image data depicting the free surface height of a laboratory tank, we tested the influences of observation errors connected to the parameters. The parameter observations are estimated from ensemble fields generated along an assimilation window. We found that the model runs are very sensitive to the parameters. They need to be set to a relatively small values. Figure 9.6 shows the evolution of the mean surface height of the wave crest region using different methods. Note that all lines except the black ones are identical to the ones of Figure 7.3. We have conducted three cases of local ensemble strategy in terms of different observation error magnitude. The suffix 'case1' (identified by dashed black line) symbolizes the case with small parameters observation errors. The suffix 'cases' (identified by dash-dot black line) is linked to the case with medium observation errors. At last, the suffix 'case3' (identified by dotted black line) represents the case with large observation errors. We can conclude that the quality of the initial analysis state is remotely related to the magnitude of observation errors. Nevertheless, the quadratic variation tensor, $\boldsymbol{a}$, estimated in the different cases generates quite distinctive trajectories.

The trajectory with small observation errors is closer to the background since the parameter estimated here is not very different from the one deduced from its ensemble estimation. The trajectory with large observation errors tends to over correct the surface height. This is realized by a stronger dissipative energy terms. An optimal value can be found by applying a medium value which allows the parameter space to be subject to extra degrees of freedom but in the meantime does not deviate from the observed value too much. We have also plotted the ensemble spread versus time in Figures 9.7. Like the synthetic case, the ensemble spread associated with subgrid model is relatively larger than those of the standard model in the first assimilation window. However, the ensemble spread still attenuates significantly during cycling procedure.

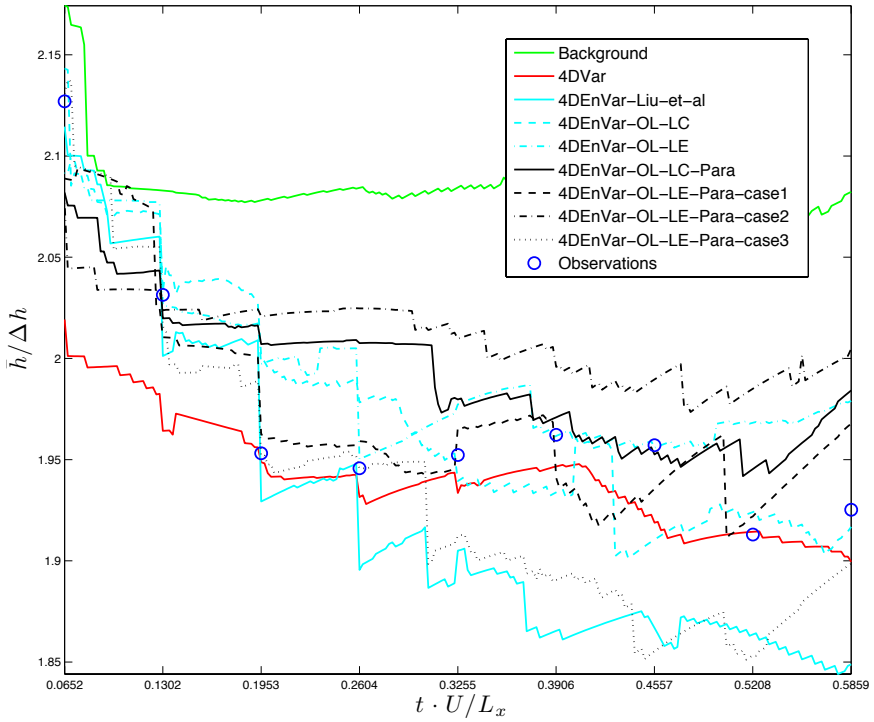


Figure 9.6 – Mean surface height of the wave crest region as a function of time - comparison of various configurations of 4DVar: Large scale simulation (green line), fine scale observation (blue points), 4DEnVar-Liu-et-al (cyan line), 4DEnVar-OL-LC (cyan dashed line), 4DEnVar-OL-LE (cyan dash-dot line), 4DEnVar-OL-LC-Para: shallow water under uncertainty with localized covariance (black line), 4DEnVar-OL-LE-Para-case1: shallow water under uncertainty with local ensemble and low noise on the parameter observation (black dashed line), 4DEnVar-OL-LE-Para-case2: shallow water under uncertainty with local ensemble and medium noise on the parameter observation (black dash-dot line), 4DEnVar-OL-LE-Para-case3: shallow water under uncertainty with local ensemble and high noise on the parameter observation (black dotted line).

9.6 Summary

In this chapter, we have presented the principles of the ensemble-based data assimilation scheme allowing a joint estimation of a parameter and of the initial condition. Results

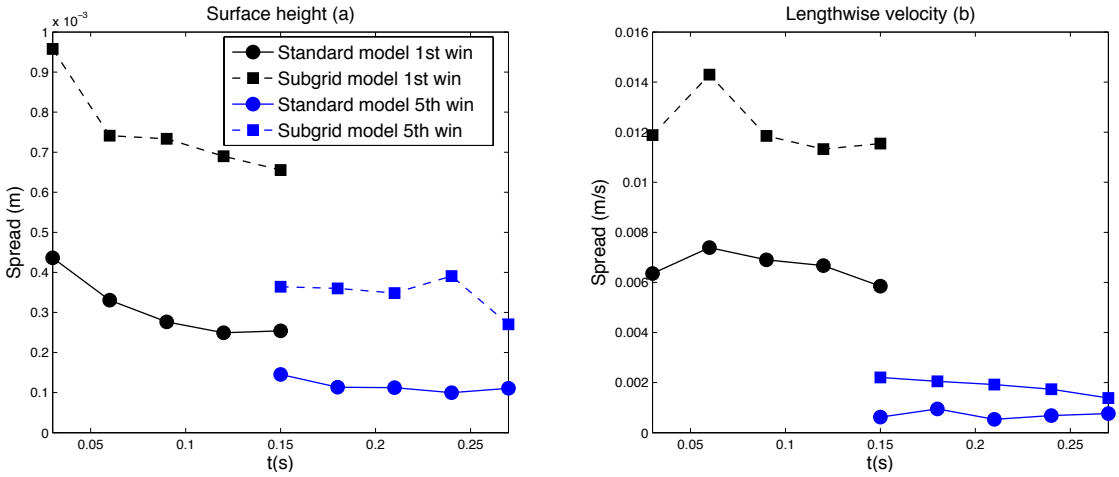


Figure 9.7 – Ensemble Spread in terms of free surface height (a) and lengthwise velocity (b): standard model (circle) with ensemble in function of initial states; subgrid model (square) with ensemble in function of initial parameters.

on both synthetic and real image data have been presented. Our stochastic shallow water model indeed shows some potential in the quantification of the unknown small scale physical processes or numerical artifacts. This is still an on-going research and we are planning to employ complex data sets and construct a similar parameter estimation scheme based on adjoint models. Such adjoint models will allow us to grasp clearer understanding of the interaction between the resolved state variables and the unresolved uncertainty terms.

Part V

Conclusions and Perspectives

Chapter 10

Conclusion and Perspectives

Conclusion

In the course of this thesis, we have explored several data assimilation strategies for fluid flow analysis. We recalled in the beginning of this thesis that a typical DA system consists of three aspects: the model, the observations and the assimilation method. Our work aimed at exploring these three topics in terms of particular ensemble technique.

The backbone of this thesis relies on an ensemble-based variational method. This method falls into the category of hybrid method in which the goal is to benefit from the advantages of variational assimilation techniques (4DVar) and ensemble Kalman filter (EnKF) while bypassing their weaknesses. A comprehensive description of the proposed strategy is listed in the chapter 5 followed by a validation in the chapter 6 on the shallow water model. Our method includes several enhancements compared to the other existing methods. We proposed a new nested loop scheme in which the background error covariance matrix is updated for each outer loop. We also devised different ensemble update schemes together with two localization schemes. In terms of the direct update approach, we exploited the links between the analysis error covariance matrix and the inverse Hessian of our 4D cost function to propose a quasi-Newton minimization procedure relying on an approximation of the inverse Hessian.

The primal objective of this thesis also takes place within the study of efficient assimilation techniques for image data observations. To that end, the proposed ensemble methods have been assessed on synthetic and real world data. Their performances have been compared to a standard 4DVar method and to several ensemble methods proposed in the literature. We found that the ensemble-based method constitutes an effective solution to handle incomplete data, which constitutes the standard situation associated with image observations. With partial observed data, the ensemble-based methods outperform the standard 4DVar in terms of unobserved component reconstruction. We also observed that the ensemble generated by a parameter perturbation provides a more pertinent ensemble spread and allows to better approximate the background error statistics when the parameter of interest is related to physical effects. The computational cost (CPU time and memory demands) of ensemble-based methods are significantly lower than standard 4DVar if a proper parallel calculation technique is deployed.

The method proposed have been also evaluated in the context of noisy experimental image data of a free surface flow supplied by a Kinect sensor (chapter 7). These observations exhibits large region of missing data. Our methods yield better result to track the surface height and show advantages in dealing with discontinuities.

A direct nonlinear image operator based on the image reconstruction error has been assessed for such ensemble techniques on a Surface Quasi Geostrophic model of oceanic flows. The ensemble methods proposed in this thesis have shown to constitute interesting techniques in the general context of a nonlinear image observation operator.

In order to deal with a strong scale discrepancy between the image data and the dynamical model's resolution grid we explored the performance of a representation of the shallow water model under uncertainty. The principal motivation comes here from the assimilation of high resolution images into large scale dynamical models. This constitutes a standard situation in geophysics. This model introduces a subgrid model encoding the effects of physical processes observed on the high resolution observation grid. Such model allows a cheaper computational cost as well. We have shown how to estimate the subgrid model parameter from the data directly and from our ensemble-based variational method. The evaluations have been carried out with 1D synthetic, 2D synthetic and 2D real image data, respectively. The results are encouraging and are showing a great potential to deal with high-resolution image data.

Perspective and future work

The future work can be composed of following subjects:

SST image application For the time being, we use the SQG model to simulate the ocean surface temperature fields. It would be worth assessing the direct image assimilation operator with the SQG model on real-world satellite images. However, the evaluation becomes a difficult issue in such a case.

Complex model application It is straightforward to consider a more complex model. Even though our simple models' results have shown great advantages of the proposed ensemble-based method. It is still unclear if this method still works with a more realistic model setup. Since the ensemble-based method are deprived of adjoint calculation, the performance of an ensemble related strategy hinges on the qualities of ensemble states. Unfortunately, there is no common laws prescribing the way of generating these ensemble. We can only rely on *ad hoc* studies performed on different case.

Uncertainty subgrid estimation We have estimated the uncertainty subgrid parameters from a point through state vector augmentation. This strategy is easy to manipulate but suffers from a severe drawback: the parameter, if unobserved, varies too freely according to the cross-covariance between the parameter and the state variables. Other methods are possible, as discussed in this thesis. It is possible to calculate the sensitivity of the model with respect to the parameter out of the adjoint operator. This is expensive but it offers an accurate way of evaluating the effect of the parameter exerted to the model trajectory. Another perspective would consists in estimating the uncertainty subgrid term, which represents actually a model error in this context, through the weak constraint variational approach as well.

Appendix A

Stochastic Reynolds transport theorem

This section cites the derivation process of the stochastic Reynolds transport theorem in Mémin (2014). We start from the Ito-Wentzell formula (8.10).

$$d\phi(\mathbf{X},t) = d_t\phi + \boldsymbol{\nabla}\phi \cdot d\mathbf{X} + \frac{1}{2}\sum_{i,j} d\langle X_i, X_j\rangle \frac{\partial^2\phi}{\partial x_i \partial x_j}dt + \sum_i d\langle \frac{\partial\phi}{\partial x_i}, (\boldsymbol{\sigma}\boldsymbol{B_t})^i\rangle dt. \tag{A.1}$$

in which ϕ is a sufficiently spatially regular enough scalar function of compact support, transported by the flow within control volume $\mathcal{V}(t)$ and on $\partial\mathcal{V}(t)$. And ϕ is a random function of the stochastic flow. For a fixed point $\boldsymbol{x}$, ϕ should satisfy a stochastic differential equation:

$$d_t\phi = v(\boldsymbol{x},t)dt + \boldsymbol{f}(\boldsymbol{x},t)\cdot d\boldsymbol{B_t}, \tag{A.2}$$

With the Ito-Wenzell formula, we can regroup the expressions for v and $\boldsymbol{f}$,

$$v(\boldsymbol{x},t) = -\boldsymbol{\nabla}\phi\cdot\boldsymbol{w} + \sum_{i,j}\frac{1}{2}a_{ij}\frac{\partial^2\phi}{\partial x^2} + \boldsymbol{\nabla}\phi\cdot\frac{\partial\boldsymbol{\sigma}_{\bullet j}}{\partial x_i}\boldsymbol{\sigma}_{ij}, \tag{A.3a}$$

$$\boldsymbol{f}(\boldsymbol{x},t) = -\boldsymbol{\nabla}\phi(\boldsymbol{x},t)^T\boldsymbol{\sigma}(\boldsymbol{x},t). \tag{A.3b}$$

Finally we have,

$$d_t\phi = \mathcal{L}\phi dt - \boldsymbol{\nabla}\phi\cdot\boldsymbol{\sigma}d\boldsymbol{B_t}, \tag{A.4a}$$

$$\mathcal{L}\phi = -\boldsymbol{\nabla}\phi\cdot\mathbf{w} + \sum_{i,j}\frac{1}{2}a_{ij}\frac{\partial^2\phi}{\partial x^2} + \boldsymbol{\nabla}\phi\cdot\frac{\partial\boldsymbol{\sigma}_{\bullet j}}{\partial x_i}\boldsymbol{\sigma}_{ij}. \tag{A.4b}$$

Such differential at a fixed point $\boldsymbol{x}$ actually corresponds to the material derivative of a function transported by the flow in the deterministic case. Suppose that q a scalar function transported by the stochastic flow as well, we can write the differential of the integral over a material volume of the product $q\phi$:

$$\begin{aligned} d\int_{\mathcal{V}(t)} q\phi(\mathbf{X}_t,t)d\boldsymbol{x} &= d\int_\Omega q\phi d\boldsymbol{x} \\ &= \int_\Omega (d_t q\phi + q d_t\phi_d t\langle q,\phi\rangle)d\boldsymbol{x}, \end{aligned} \tag{A.5}$$

From (A.4b), this differential can be written as,

$$\int_{\Omega}(d_t q\phi + q\mathcal{L}\phi + \boldsymbol{\nabla}\phi\cdot\boldsymbol{a}\boldsymbol{\nabla}q)dtd\boldsymbol{x} - \int_{\Omega} q\boldsymbol{\nabla}\phi\cdot\boldsymbol{\sigma}d\boldsymbol{B_t}. \tag{A.6}$$

Owing to the adjoint operator $\mathcal{L}^*$ with Dirichlet boundary conditions:

$$\int_{\Omega}\Big(\big(d_t q + \mathcal{L}^* q + \boldsymbol{\nabla}\cdot(\boldsymbol{a}\boldsymbol{\nabla}q)\big)dt + \boldsymbol{\nabla}\cdot(q\phi\boldsymbol{\sigma}d\boldsymbol{B_t})\Big)\phi d\boldsymbol{x}. \tag{A.7}$$

By inserting a complete expression $\mathcal{L}^*$, we have:

$$\begin{aligned}\int_{\mathcal{V}}[d_t q + \Big(\boldsymbol{\nabla}\cdot(q\mathbf{w})dt - \sum_{ij}\boldsymbol{\nabla}\cdot(q\sigma^{ij}\frac{\partial\sigma^{j\bullet}}{\partial x_i}) + \frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i\partial x_j}(a_{ij}q)\\ - \sum_{ij}\partial_{x_i}a^{ij}\partial_{x_j}q - \sum_{ij}a^{ij}\partial^2_{x_j}q\Big)dt + \boldsymbol{\nabla}\cdot(q\boldsymbol{\sigma}d\boldsymbol{B_t})]d\boldsymbol{x},\end{aligned} \tag{A.8}$$

The term $-\frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i\partial x_j}(a_{ij}q)|_{\boldsymbol{\nabla}\cdot\sigma=0}dt + \frac{1}{2}\parallel\boldsymbol{\nabla}\cdot\sigma\parallel^2 qdt$ can be written as, omitting dt:

$$\begin{aligned}&-\frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i\partial x_j}(a_{ij}q)|_{\boldsymbol{\nabla}\cdot\sigma=0} + \frac{1}{2}\parallel\boldsymbol{\nabla}\cdot\sigma\parallel^2 q =\\ &-\frac{1}{2}\sum_{ij}\big(a_{ij}\frac{\partial^2 q}{\partial x_i\partial x_j} + \frac{\partial a_{ij}}{\partial x_i}\frac{\partial q}{\partial x_j} + \frac{\partial a_{ij}}{\partial x_j}\frac{\partial q}{\partial x_i}\big) + \frac{1}{2}\big(\sum_{ijk}\partial_{x_i}\sigma^{ik}\partial_{x_j}\sigma^{jk}q\big) =\\ &-\frac{1}{2}\sum_{ijk}\big(\sigma_{ik}\sigma_{kj}\frac{\partial^2 q}{\partial x_i\partial x_j} + \frac{\partial a_{ij}}{\partial x_i}\frac{\partial q}{\partial x_j} + \frac{\partial a_{ij}}{\partial x_j}\frac{\partial q}{\partial x_i} - \partial_{x_i}\sigma^{ik}\partial_{x_j}\sigma^{jk}q\big),\end{aligned} \tag{A.9}$$

which can be simplified to,

$$\begin{aligned}\int_{\mathcal{V}}[d_t q + \boldsymbol{\nabla}\cdot(q\mathbf{w})dt - \frac{1}{2}\sum_{ijk}(\sigma^{ik}\sigma^{kj}\frac{\partial^2 q}{\partial x_i\partial x_j} + \partial_{x_j}\sigma^{ik}\partial_{x_i}\sigma^{kj}q\\ +2\sigma^{ik}\partial_{x_i}\sigma^{kj}\partial_{x_j}q - \partial_{x_i}\sigma^{ik}\partial_{x_j}\sigma^{kj}q) + \boldsymbol{\nabla}\cdot(q\boldsymbol{\sigma}d\boldsymbol{B_t})]d\boldsymbol{x}.\end{aligned} \tag{A.10}$$

Above equation can be compactly written as

$$d\int_{\mathcal{V}} qdx = \int_{\mathcal{V}}[d_t q + \boldsymbol{\nabla}\cdot(q\mathbf{w})dt - \frac{1}{2}\sum_{ij}\frac{\partial^2}{\partial x_i\partial x_j}(a_{ij}q)|_{\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}=0}dt + \frac{1}{2}\parallel\boldsymbol{\nabla}\cdot\boldsymbol{\sigma}\parallel^2 qdt + \boldsymbol{\nabla}\cdot(q\boldsymbol{\sigma}d\boldsymbol{B_t})]dx. \tag{A.11}$$

Bibliography

Allen, E. (2007). *Modeling with Itô stochastic differential equations*, volume 22. Springer.

Ambrosi, D. (1995). Approximation of shallow water equations by roe's riemann solver. *International journal for numerical methods in fluids*, 20(2):157–168.

Anderson, B. D. O. and Moore, J. B. (1979). *Optimal Filtering*. Englewood Cliffs, NJ : Prentice Hall.

Anderson, J. L. (2001). An ensemble adjustment Kalman filter for data assimilation. *Mon. Wea. Rev.*, 129(12):2884–2903.

Anderson, J. L. (2003). A local least squares framework for ensemble filtering. *Mon. Wea. Rev.*, 131(4):634–642.

Baek, S.-J., Hunt, B. R., Kalnay, E., Ott, E., and Szunyogh, I. (2006). Local ensemble Kalman filtering in the presence of model bias. *TellusA*, 58(3):293–306.

Bannister, R. N. (2008a). A review of forecast error covariance statistics in atmospheric variational data assimilation. I: Characteristics and measurements of forecast error covariances. *Quart. J. Roy. Meteor. Soc.*, 134(637):1951–1970.

Bannister, R. N. (2008b). A review of forecast error covariance statistics in atmospheric variational data assimilation. II: Modelling the forecast error covariance statistics. *Quart. J. Roy. Meteor. Soc.*, 134(637):1971–1996.

Bergthorsson, P. and Döös, B. R. (1955). Numerical weather map analysis. *Tellus*, 7(3):329–340.

Beyou, S. (2013). *Estimation de la vitesse des courants marins à partir de séquences d'images satellitaires* . PhD thesis, Université Rennes 1.

Beyou, S., Corpetti, T., Gorthi, S., and Mémin, E. (2013a). Fluid flow estimation with multiscale ensemble filters based on motion measurements under location uncertainty. *Numerical Mathematics: Theory, Methods and Applications*, 6:21–46.

Beyou, S., Cuzol, A., Gorthi, S., and Mémin, E. (2013b). Weighted ensemble transform kalman filter for image assimilation. *TellusA*, 65:18803.

Bishop, C. H., Etherton, B. J., and Majumdar, S. J. (2001). Adaptive sampling with the ensemble transform Kalman filter. part I: Theoretical aspects. *Mon. Wea. Rev.*, 129(3):420–436.

Bishop, C. H., Hodyss, D., Steinle, P., Sims, H., Clayton, A. M., Lorenc, A. C., Barker, D. M., and Buehner, M. (2011). Efficient Ensemble Covariance Localization in Variational Data Assimilation. *Mon. Wea. Rev.*, 139(2):573–580.

Blum, J., Le Dimet, F.-X., and Navon, I. M. (2009). Data Assimilation for Geophysical Fluids. In *Handbook of Numerical Analysis, Vol. XIV*, pages 385–441. Elsevier.

Bocquet, M. and Sakov, P. (2012). Combining inflation-free and iterative ensemble Kalman filters for strongly nonlinear systems. *Nonlinear Processes in Geophysics*, 19(3):383–399.

Bocquet, M. and Sakov, P. (2013a). An iterative ensemble Kalman smoother. *Quarterly Journal of the Royal Meteorological Society*, 140(682):1521–1535.

Bocquet, M. and Sakov, P. (2013b). Joint state and parameter estimation with an iterative ensemble Kalman smoother. *Nonlinear Processes in Geophysics*, 20(5):803–818.

Buehner, M. (2005). Ensemble-derived stationary and flow-dependent background-error covariances: Evaluation in a quasi-operational NWP setting. *Quart. J. Roy. Meteor. Soc.*, 131(607):1013–1043.

Buehner, M., Houtekamer, P. L., Charette, C., Mitchell, H. L., and He, B. (2010a). Intercomparison of Variational Data Assimilation and the Ensemble Kalman Filter for Global Deterministic NWP. Part I: Description and Single-Observation Experiments. *Mon. Wea. Rev.*, 138(5):1550–1566.

Buehner, M., Houtekamer, P. L., Charette, C., Mitchell, H. L., and He, B. (2010b). Intercomparison of Variational Data Assimilation and the Ensemble Kalman Filter for Global Deterministic NWP. Part II: One-Month Experiments with Real Observations. *Mon. Wea. Rev.*, 138(5):1567–1586.

Buehner, M., Morneau, J., and Charette, C. (2013). Four-dimensional ensemble-variational data assimilation for global deterministic weather prediction. *Nonlinear Processes in Geophysics*, 20(5):669–682.

Bueno-Orovio, A., Kay, D., and Burrage, K. (2012). Fourier spectral methods for fractional-in-space reaction-diffusion equations. *BIT Numerical Mathematics*, pages 1–18.

Burgers, G., van Leeuwen, P., and Evensen, G. (1998). Analysis scheme in the ensemble Kalman filter. *Mon. Wea. Rev.*, 126:1719–1724.

Chehab, J.-P., Moalla, M. T., et al. (2012). Numerical simulations of a 2d quasi geostrophic equation.

Clayton, A. M., Lorenc, A. C., and Barker, D. M. (2012). Operational implementation of a hybrid ensemble/4D-Var global data assimilation system at the Met Office. *Quart. J. Roy. Meteor. Soc.*, 139(675):1445–1461.

Combes, B., Guibert, A., Mémin, E., and Heitz, D. (2011). Free-surface flows from Kinect: Feasibility and limits. *FVR2011*.

Combes, B., Heitz, D., Guibert, A., and Mémin, E. (2015). A particle filter to reconstruct a free-surface flow from a depth camera. *Fluid Dynamics Research*. minor revision.

Constantin, P. and Iyer, G. (2008). A stochastic lagrangian representation of the three-dimensional incompressible navier-stokes equations. *Communications on Pure and Applied Mathematics*, 61(3):330–345.

Constantin, P. and Wu, J. (1999). Behavior of solutions of 2d quasi-geostrophic equations. *SIAM journal on mathematical analysis*, 30(5):937–948.

Corpetti, T., Héas, P., Mémin, E., and Papadakis, N. (2009). Pressure image assimilation for atmospheric motion estimation. *TellusA*, 61(1):160–178.

Courtier, P., Thépaut, J.-N., and Hollingsworth, A. (1994). A strategy for operational implementation of 4d-var, using an incremental approach. *Quart. J. Roy. Meteor. Soc.*, 120(519):1367–1387.

Cressman, G. P. (1959). An operational objective analysis system. *Monthly Weather Review*, 87(10):367–374.

Cushman-Roisin, B. and Beckers, J.-M. (2011). Introduction to Geophysical Fluid Dynamics.

Deardorff, J. W. (1970). A numerical study of three-dimensional turbulent channel flow at large reynolds numbers. *Journal of Fluid Mechanics*, 41(02):453–480.

Dee, D. P. (2005). Bias and data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 131(613):3323–3343.

Derber, J. and Bouttier, F. (1999). A reformulation of the background error covariance in the ecmwf global data assimilation system. *Tellus A*.

Desroziers, G., Camino, J.-T., and BERRE, L. (2014). 4DEnVar: link with 4D state formulation of variational assimilation and different possible implementations. *Quart. J. Roy. Meteor. Soc.*, pages n/a–n/a.

Evensen, G. (1994). Sequential data assimilation with a non linear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.*, 99 (C5)(10):143–162.

Evensen, G. (2003). The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dynamics*, 53:343–367.

Evensen, G. (2004). Sampling strategies and square root analysis schemes for the EnKF. *Ocean dynamics*, 54(6):539–560.

Evensen, G. and van Leeuwen, P. J. (2000). An ensemble Kalman smoother for nonlinear dynamics. *Mon. Wea. Rev.*, 128(6):1852–1867.

Fairbairn, D., Pring, S. R., Lorenc, A. C., and Roulstone, I. (2013). A comparison of 4DVar with ensemble data assimilation methods. *Quart. J. Roy. Meteor. Soc.*, 140(678):281–294.

Fertig, E. J., Harlim, J., and Hunt, B. R. (2007). A comparative study of 4D-VAR and a 4D Ensemble Kalman Filter: perfect model simulations with Lorenz-96. *Tellus A*, 59(1):96–100.

Frederiksen, J. S., O'Kane, T. J., and Zidikheri, M. J. (2013). Subgrid modelling for geophysical flows. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1982):20120166.

Gandin, L. (1965). Objective analysis of meteorological fields: Gidrometeorologicheskoe izdatelâĂŹstvo (gimiz), leningrad. *Translated by Israel Program for Scientific Translations, Jerusalem*.

Gaspari, G. and Cohn, S. E. (1999). Construction of correlation functions in two and three dimensions. *Quart. J. Roy. Meteor. Soc.*, 125(554):723–757.

Gelb, A., editor (1974). *Applied Optimal Estimation*. The MIT Press.

Gordon, N., Salmond, D., and Smith, A. (1993). Novel approach to non-linear/non-gaussian bayesian state estimation. *IEEE Processing-F*, 140(2).

Gottwald, G. A. and Majda, A. (2013). A mechanism for catastrophic filter divergence in data assimilation for sparse observation networks. *Nonlinear Processes in Geophysics*, 20(5):705–712.

Haben, S. A. (2011). *Conditioning and Preconditioning of the Minimization Problem in Variational Data Assimilation*. PhD thesis, The University of Reading.

Haben, S. A., Lawless, A. S., and Nichols, N. K. (2011a). Conditioning and preconditioning of the variational data assimilation problem. *Computers and Fluids*, 46(1):252–256.

Haben, S. A., Lawless, A. S., and Nichols, N. K. (2011b). Conditioning of incremental variational data assimilation, with application to the Met Office system. *Tellus A*, 63(4):782–792.

Hamill, T. M. and Snyder, C. (2000). A Hybrid Ensemble Kalman Filter–3D Variational Analysis Scheme. *Mon. Wea. Rev.*, 128(8):2905–2919.

Harlim, J. and Hunt, B. R. (2007). Four-dimensional local ensemble transform kalman filter: numerical experiments with a global circulation model. *Tellus A*, 59(5):731–748.

Held, I. M., Pierrehumbert, R. T., Garner, S. T., and Swanson, K. L. (1995). Surface quasi-geostrophic dynamics. *Journal of Fluid Mechanics*, 282:1–20.

Higham, N. J. (2002). *Accuracy and Stability of Numerical Algorithms*. Second Edition. SIAM.

Houtekamer, P. L. and Mitchell, H. L. (1998). Data assimilation using an ensemble Kalman filter technique. *Mon. Wea. Rev.*, 126(3):796–811.

Houtekamer, P. L. and Mitchell, H. L. (2001). A sequential ensemble Kalman filter for atmospheric data assimilation. *Mon. Wea. Rev.*, 129(1):123–137.

Houtekamer, P. L. and Mitchell, H. L. (2005). Ensemble Kalman filtering. *Quart. J. Roy. Meteor. Soc.*, 131(613):3269–3289.

Hunt, B. R., Kalnay, E., Kostelich, E. J., Ott, E., Patil, D. J., Sauer, T., Szunyogh, I., Yorke, J. A., and Zimin, A. V. (2004). Four-dimensional ensemble Kalman filtering. *Tellus A*, 56(4):273–277.

Hunt, B. R., Kostelich, E. J., and Szunyogh, I. (2007). Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. *Physica D*, 230:112–126.

Kadri Harouna, S. and Mémin, E. (2014). Stochastic reynolds transport theorem and navier-stokes equations under uncertainty: numerical simulation and assessment. *Inria Research Report*.

Kalnay, E., Li, H., Miyoshi, T., and Yang, S. (2007). 4-D-Var or ensemble Kalman filter? *TellusA*, 59:758–773.

Kang, J.-S., Kalnay, E., Liu, J., Fung, I., Miyoshi, T., and Ide, K. (2011). âĂIJvariable localizationâĂİ in an ensemble kalman filter: Application to the carbon cycle data assimilation. *Journal of Geophysical Research: Atmospheres (1984–2012)*, 116(D9).

Kazantsev, E. (2012). Sensitivity of a shallow-water model to parameters. *Nonlinear Analysis: Real World Applications*, 13(3):1416–1428.

Koyama, H. and Watanabe, M. (2010). Reducing forecast errors due to model imperfections using ensemble kalman filtering. *Monthly Weather Review*, 138(8):3316–3332.

Krysta, M., Blayo, E., Cosme, E., and Verron, J. (2011). A Consistent Hybrid Variational-Smoothing Data Assimilation Method: Application to a Simple Shallow-Water Model of the Turbulent Midlatitude Ocean. *Mon. Wea. Rev.*, 139(11):3333–3347.

Lawson, W. G. and Hansen, J. A. (2004). Implications of Stochastic and Deterministic Filters as Ensemble-Based Data Assimilation Methods in Varying Regimes of Error Growth. *Mon. Wea. Rev.*, 132:1966–1981.

Le Dimet, F.-X. and Talagrand, O. (1986). Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus*, 38A:97–110.

Li, Z. and Navon, I. M. (2001). Optimality of variational data assimilation and its relationship with the Kalman filter and smoother. *Quart. J. Roy. Meteor. Soc.*, 127(572):661–683.

Lions, J. L. (1971). *Optimal control of systmes governed by PDEs*. Springer-Verlag, New-York.

Liu, C., Xiao, Q., and Wang, B. (2008). An Ensemble-Based Four-Dimensional Variational Data Assimilation Scheme. Part I: Technical Formulation and Preliminary Test. *Mon. Wea. Rev.*, 136(9):3363–3373.

Liu, C., Xiao, Q., and Wang, B. (2009). An Ensemble-Based Four-Dimensional Variational Data Assimilation Scheme. Part II: Observing System Simulation Experiments with Advanced Research WRF (ARW). *Mon. Wea. Rev.*, 137(5):1687–1704.

Lorenc, A., Bell, R., and Macpherson, B. (1991). The meteorological office analysis correction data assimilation scheme. *Quarterly Journal of the Royal Meteorological Society*, 117(497):59–89.

Lorenc, A. C. (1986). Analysis methods for numerical weather prediction. *Quart. J. Roy. Meteor. Soc.*, 112:1177–1194.

Lorenc, A. C. (2003). The potential of the ensemble kalman filter for nwp — a comparison with 4d-var. *Quart. J. Roy. Meteor. Soc.*, 129:3183–3203.

Lorenz, E. N. (1963). Deterministic Nonperiodic Flow. *J. Atmos. Sci.*, 58:130–141.

Mémin, E. (2014). Fluid flow dynamics under location uncertainty. *Geophysical and Astrophysical Fluid Dynamics*, 108(2):119–146.

Mikulevicius, R. and Rozovskii, B. (2004). Stochastic navier–stokes equations for turbulent flows. *SIAM Journal on Mathematical Analysis*, 35(5):1250–1310.

Miller, R. N., Carter, E. F., and Blue, S. T. (1999). Data assimilation into nonlinear stochastic models. *Tellus A*, 51(2):167–194.

Mirouze, I. and Weaver, A. T. (2010). Representation of correlation functions in variational assimilation using an implicit diffusion operator. *Quart. J. Roy. Meteor. Soc.*, 136:1421–1443.

Moeng, C.-H. (1984). A large-eddy-simulation model for the study of planetary boundary-layer turbulence. *Journal of the Atmospheric Sciences*, 41(13):2052–2062.

Navon, I. (1998). Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography. *Dynamics of Atmospheres and Oceans*, 27(1):55–79.

Nocedal, J. and Wright, S. J. (2004). *Numerical Optimization*. Springer.

Ott, E., Hunt, B. R., Szunyogh, I., Zimin, A. V., E. J. Kostelich, M. C., Kalnay, E., Patil, D. J., and Yorke, J. A. (2004). A local ensemble Kalman filter for atmospheric data assimilation. *Tellus*, 56A:415–428.

Papadakis, N. and Mémin, E. (2008). Variational Assimilation of Fluid Motion from Image Sequence. *SIAM Journal on Imaging Sciences*, 1(4):343–363.

Papadakis, N., Mémin, E., Cuzol, A., and Gengembre, N. (2010). Data assimilation with the weighted ensemble kalman filter. *Tellus-A*, 62(5):673–697.

Parrish, D. F. and Derber, J. C. (1992). The National Meteorological Center's spectral statistical-interpolation analysis system. *Mon. Wea. Rev.*, 120(8):1747–1763.

Priestley, A. (1987). *Numerical solution of the shallow-water equations*. PhD thesis, Delft Univ. of Tech.

Pulido, M. and Thuburn, J. (2005). Gravity wave drag estimation from global analyses using variational data assimilation principles. i: Theory and implementation. *Quarterly Journal of the Royal Meteorological Society*, 131(609):1821–1840.

Qiu, C. and Chou, J. (2005). Four-dimensional data assimilation method based on SVD: Theoretical aspect. *Theoretical and Applied Climatology*, 83(1-4):51–57.

Rabier, F. and Courtier, P. (1992). Four-dimensional assimilation in the presence of baroclinic instability. *Quart. J. Roy. Meteor. Soc.*, 118:649–672.

Roe, P. L. (1981). Approximate riemann solvers, parameter vectors, and difference schemes. *J. Comp. Phys.*, 43:357–372.

Ruiz, J. J., Pulido, M., and Mitoshi, T. (2013a). Estimating model parameters with ensemble-based data assimilation: A review. *Journal of the Meteorological society of Japan*, 91(2):79–99.

Ruiz, J. J., Pulido, M., and Miyoshi, T. (2013b). Estimating model parameters with ensemble-based data assimilation: Parameter covariance treatment. *Journal of the Meteorological society of Japan*, 91(4):453–469.

Sakov, P., Evensen, G., and Bertino, L. (2010). Asynchronous data assimilation with the EnKF. *Tellus A*, 62(1):24–29.

Sakov, P. and Oke, P. (2007). Implications of the form of the ensemble transformation in the ensemble square root filter. *Mon. Wea. Rev.*, 136:1042–1053.

Sakov, P. and Oke, P. R. (2008). A deterministic formulation of the ensemble Kalman filter: an alternative to ensemble square root filters. *Tellus A*, 60(2):361–371.

Sakov, P., Oliver, D. S., and Bertino, L. (2012). An Iterative EnKF for Strongly Nonlinear Systems. *Mon. Wea. Rev.*, 140(6):1988–2004.

Sasaki, Y. (1958). An objective analysis based on the variational method. *J. Meteor. Soc. Japan*, 36(3):77–88.

Sasaki, Y. (1970). Some Basic Formalisms in Numerical Variational Analysis. *Mon. Wea. Rev.*, 98:875–883.

Smagorinsky, J. (1963). General circulation experiments with the primitive equations: I. the basic experiment*. *Monthly weather review*, 91(3):99–164.

Solonen, A., Haario, H., Hakkarainen, J., Auvinen, H., Amour, I., and Kauranne, T. (2012). Variational ensemble Kalman filtering using limited memory BFGS. *Electronic Transactions on Numerical Analysis*, 39:271–285.

Souopgui, I. (2010). *Assimilation d'images pour les fluides géophysiques*. PhD thesis, Université Joseph-Fourier-Grenoble I.

Talagrand, O. (1997). Assimilation of observations, an introduction. *J. Meteor. Soc. Japan*.

Tian, X. J., Xie, Z. H., and Dai, A. G. (2008). An ensemble-based explicit four-dimensional variational assimilation method. *J. Geophys. Res.*, 113(D21):D21124.

Tippett, M., Anderson, J., Craig, C. B., Hamill, T., and Whitaker, J. (2003). Ensemble square root filters. *Mon. Wea. Rev.*, 131(7):1485–1490.

Titaud, O., Vidard, A., Souopgui, I., and Le Dimet, F.-X. (2010). Assimilation of image sequences in numerical models. *Tellus A*, 62(1):30–47.

Tong, M. and Xue, M. (2008). Simultaneous estimation of microphysical parameters and atmospheric state with simulated radar data and ensemble square root kalman filter. part i: Sensitivity analysis and parameter identifiability. *Monthly weather review*, 136(5):1630–1648.

Toro, E. F. (2009). *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer.

Van Leeuwen, P. J. (2009). Particle filtering in geophysical systems. *Monthly Weather Review*, 137(12):4089–4114.

van Leeuwen, P. J. (2010). Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quart. J. Roy. Meteor. Soc.*, 136(653):1991–1999.

van Leeuwen, P. J. and Evensen, G. (1996). Data assimilation and inverse methods in terms of a probabilistic formulation. *Mon. Wea. Rev.*, 124:2898–2913.

Vreugdenhil, C. B. (1994). *Numerical methods for shallow-water flow*. Water Science and Technology Library.

Wang, X., Barker, D. M., Snyder, C., and Hamill, T. M. (2008a). A Hybrid ETKF–3DVAR Data Assimilation Scheme for the WRF Model. Part I: Observing System Simulation Experiment. *Mon. Wea. Rev.*, 136(12):5116–5131.

Wang, X., Barker, D. M., Snyder, C., and Hamill, T. M. (2008b). A Hybrid ETKF–3DVAR Data Assimilation Scheme for the WRF Model. Part II: Real Observation Experiments. *Mon. Wea. Rev.*, 136(12):5132–5147.

Wang, X., Bishop, C. H., and Julier, S. J. (2004). Which is better, an ensemble of positive-negative pairs or a centered simplex ensemble? *Mon. Wea. Rev.*, 132:1590–1605.

Wang, X., Snyder, C., and Hamill, T. M. (2007). On the Theoretical Equivalence of Differently Proposed Ensemble–3DVAR Hybrid Analysis Schemes. *Mon. Wea. Rev.*, 135(1):222–227.

Weaver, A. and Courtier, P. (2001). Correlation modelling on the sphere using a generalized diffusion equation. *Quart. J. Roy. Meteor. Soc.*, 127:1815–846.

Weaver, A. T., Vialard, J., and Anderson, D. L. T. (2003). Three- and Four-Dimensional Variational Assimilation with a General Circulation Model of the Tropical Pacific Ocean. Part I: Formulation, Internal Diagnostics, and Consistency Checks. *Mon. Wea. Rev.*, 131:1360–1378.

Wei, M., Jacobs, G., Rowley, C., Barron, C. N., Hogan, P., Spence, P., Smedstad, O. M., Martin, P., Muscarella, P., and Coelho, E. (2013). The impact of initial spread calibration on the RELO ensemble and its application to Lagrangian dynamics. *Nonlinear Processes in Geophysics*, 20(5):621–641.

Whitaker, J. S. and Hamill, T. M. (2002). Ensemble data assimilation without perturbed observations. *Mon. Wea. Rev.*, 130(7):913–1924.

Wikle, C. K. and Berliner, L. M. (2007). A Bayesian tutorial for data assimilation. *Physica D: Nonlinear Phenomena*, 230(1):1–16.

Wu, L., Mallet, V., Bocquet, M., and Sportisse, B. (2008). A comparison study of data assimilation algorithms for ozone forecasts. *Journal of Geophysical Research*, 113(D20).

Yang, S.-C., Corazza, M., Carrassi, A., Kalnay, E., and Miyoshi, T. (2009). Comparison of Local Ensemble Transform Kalman Filter, 3DVAR, and 4DVAR in a Quasigeostrophic Model. *Mon. Wea. Rev.*, 137(2):693–709.

Yang, X. and Delsole, T. (2009). Using the ensemble Kalman filter to estimate multiplicative model parameters. *TellusA*, 61(5):601–609.

Zhang, F. Q., Zhang, M., and Hansen, J. A. (2009). Coupling ensemble Kalman filter with four-dimensional variational data assimilation. *Adv. Atmos. Sci.*, 26(1):1–8.

Zhang, M. and Zhang, F. (2012). E4DVar: Coupling an Ensemble Kalman Filter with Four-Dimensional Variational Data Assimilation in a Limited-Area Weather Prediction Model. *Mon. Wea. Rev.*, 140(2):587–600.

Zhu, Y. and Navon, I. (1999). Impact of parameter estimation on the performance of the fsu global spectral model using its full-physics adjoint. *Monthly Weather Review*, 127(7):1497–1517.

Zou, X., Navon, I. M., and LeDimet, F. X. (1992). An optimal nudging data assimilation scheme using parameter estimation. *Quart. J. Roy. Meteor. Soc.*, 118(508):1163–1186.

Zupanski, D. and Zupanski, M. (2006). Model error estimation employing an ensemble data assimilation approach. *Monthly Weather Review*, 134(5):1337–1354.

Zupanski, M. (2005). Maximum likelihood ensemble filter: theoretical aspects. *Mon. Wea. Rev.*, 130:1710–1726.

Abstract: The hybrid methods combing the 4D variational method and the ensemble Kalman filter provide a flexible framework. In such framework the potential advantages with respect to each method can be retained. These advantages include, for example, the flow-dependent background error covariance, the ability to explicitly get the analysis error covariance matrix, the iterative minimization procedure and the simultaneously assimilation of all observations with in a time span *etc.*

In this thesis, an enhanced ensemble-based 4DVar scheme is proposed and has been analyzed in detail in the case of the 2D shallow water model. Several variations related to this method are introduced and tested. We proposed a new nested loop scheme in which the background error covariance matrix is updated for each outer loop. We also devised different ensemble update schemes together with two localization schemes. And we exploited the links between the analysis error covariance matrix and the inverse Hessian of our 4D cost function. All these variants have been tested with the real Kinect-captured image data and synthetic image data associated with a SQG (Surface Quasi-Geostrophic) model, respectively.

Our proposed ensemble-based variational method is then used to devise a parameter estimation scheme. Such formulation allows the estimation of an uncertainty subgrid stress tensor in the context of ensemble estimation. And this uncertainty subgrid stress tensor is derived from a perspective of flow phenomenon driven by a stochastic process. Finally, a first effort is made to assimilation high-resolution image data with the dynamical model running on a much coarser grid.
Keywords: data assimilation, variational methods, Kalman filter, fluid dynamics, subgrid stress modeling
Résumé: Les méthodes hybrides combinant les méthodes de 4D variationnelles et le filtre de Kalman d'ensemble fournissent un cadre flexible. Dans ce cadre, les avantages potentiels par rapport à chaque méthode peuvent être conservé. Ces avantages comprennent, par exemple, la matrice de covariance d'erreur d'ébauche dépendant d'écoulement, la capacité d'obtenir explicitement la matrice de covariances d'erreur d'analyse, la procédure de minimisation itérative et l'assimilation simultanée de toutes les observations dans un intervalle de temps *etc.*

Dans cette thèse, un système d'ensemblist-4DVar renforcé a été proposé et a été analysé en détail dans le cas du modèle de 2D shallow-water. Nous avons proposé un nouveau schéma de boucle imbriquée dans laquelle la matrice de covariance d'erreur d'ébauche est mis à jour pour chaque boucle externe. Nous avons aussi élaboré différents schémas de mise à jour d'ensemble avec deux stratégies de localisation différentes. Dans l'approche de transformation directe, nous avons exploité les liens entre la matrice de covariances d'erreur d'analyse et la matrice hessienne de la fonction coût. Toutes ces variantes ont été testées avec les données réelles de l'image capturées par Kinect et de l'image associées à un modèle de Surface quasi-géostrophique, respectivement.

Notre méthodes variationnelles d'ensemble proposées sont ensuite utilisées pour la concevoir un système d'estimation des paramètres. Cette formulation nous permet d'estimer des paramètres du stress tenseur de l'incertitude. Ce stress tenseur est dérivé d'un point de vue de phénomène d'écoulement modélisé par un processus stochastique. Enfin, un premier effort est fait pour l'assimilation des données d'image à haute résolution avec le modèle dynamique sur une grille plus grossière.
Mots-clés: assimilation de donnée, méthodes variationnelle, filtrage de Kalman, dynamique des fluides, modélisation de stress à échelle grille-fine