Estimation du mouvement par assimilation de données dans des modèles dynamiques d'ordre réduit

Karim Drifi

▶ **To cite this version:**

Karim Drifi. Estimation du mouvement par assimilation de données dans des modèles dynamiques d'ordre réduit. Modélisation et simulation. Université Pierre et Marie Curie - Paris VI, 2013. Français. tel-00846688

HAL Id: tel-00846688

https://tel.archives-ouvertes.fr/tel-00846688

Submitted on 19 Jul 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Estimation du mouvement par assimilation de données dans des modèles dynamiques d'ordre réduit

Karim Drifi[1]

1. INRIA, Paris-Rocquencourt Research Center, France

Table des matières

I Estimation du mouvement par assimilation de données dans des modèles réduits **9**

1 Assimilation d'images pour l'estimation du mouvement **10**
1.1 Introduction . . . 10
1.2 Système d'assimilation . . . 11
1.3 Modèle d'advection du mouvement et des images . . . 13
1.4 Résultats d'assimilation . . . 13
1.5 Projection de Galerkin d'un modèle dynamique . . . 15
1.6 Conclusion . . . 16

2 Apprentissage de base par décomposition orthogonale propre **17**
2.1 Introduction . . . 17
2.2 Projection de Galerkin du modèle d'advection image . . . 18
2.2.1 Rappel des équations . . . 18
2.2.2 Projection des équations . . . 18
2.3 Bases réduites obtenues par décomposition orthogonale propre . . . 21
2.3.1 Simulation du modèle complet . . . 21
2.3.2 Décomposition orthogonale propre . . . 21
2.3.3 Analyse des projections sur les bases Φ et Ψ . . . 24
2.4 Simulation du modèle réduit . . . 26
2.4.1 Simulation du modèle réduit . . . 26
2.4.2 Résultats . . . 27
2.5 Assimilation dans le modèle réduit . . . 27
2.5.1 Système d'assimilation . . . 30
2.5.2 Expérience jumelle d'assimilation . . . 31
2.6 Conclusions . . . 35

3 Réduction du modèle de mouvement à divergence nulle **36**
3.1 Introduction . . . 36
3.2 Modèle complet d'advection image à divergence nulle . . . 36
3.2.1 État du modèle . . . 36
3.2.2 Dynamique du modèle . . . 37
3.2.3 Calcul de $\mathbf{w}$ en fonction de ξ . . . 38
3.2.4 Calcul de φ . . . 38
3.3 Projection de Galerkin pour l'obtention du modèle réduit . . . 44
3.4 Simulation . . . 46
3.4.1 Simulation du modèle complet . . . 46

3.4.2 Simulation du modèle réduit . . . 46
3.4.3 Conclusions . . . 48
3.5 Assimilation dans le modèle réduit . . . 50
3.5.1 Conclusions . . . 51
3.6 Expérience à fenêtre glissante . . . 57
3.6.1 Introduction . . . 57
3.6.2 Description de l'expérience . . . 57
3.6.3 Expérience jumelle par fenêtre glissante . . . 58
3.6.4 Remarques et conclusions . . . 62
3.7 Comparaison avec l'état de l'art . . . 62
3.8 Conclusions . . . 63

II Bases fixes pour l'estimation du mouvement 65

4 Projection du modèle à divergence nulle sur une base de fonctions sinus 66
4.1 Introduction . . . 66
4.2 Rappels du formalisme . . . 67
4.3 Espace réduit – Base de fonctions sinus . . . 68
4.4 Simulation . . . 69
4.4.1 Simulation modèle complet . . . 69
4.4.2 Simulation modèle réduit . . . 69
4.4.3 Conclusions . . . 71
4.5 Expérience jumelle d'assimilation . . . 71
4.5.1 Assimilation dans le modèle réduit MR-AIMIDF_sinus-POD . . . 71
4.5.2 Conclusions . . . 72
4.6 Projection de l'équation du mouvement sur base de fonctions sinus . . . 74
4.6.1 Introduction . . . 74
4.6.2 Assimilation dans le modèle réduit MR-AIMIDF_sinus . . . 74
4.6.3 Conclusions . . . 76
4.7 Comparaison des résultats d'assimilation dans les modèles complet et réduit 78
4.7.1 Expérience d'assimilation sur des données synthétiques . . . 78
4.7.2 Assimilation d'images satellite . . . 80
4.8 Expérience fenêtre glissante . . . 86
4.8.1 Introduction . . . 86
4.8.2 Description de l'expérience . . . 86
4.8.3 Expérience jumelle par fenêtre glissante . . . 87
4.8.4 Conclusions . . . 88

5 Estimation du mouvement sur bases d'ondelettes 91
5.1 Introduction . . . 91
5.2 Bases d'ondelettes . . . 92
5.2.1 Base d'ondelettes pour le mouvement à divergence nulle . . . 92
5.2.2 Base image ondelettes . . . 93
5.3 Expériences synthétiques . . . 94
5.3.1 Advection du mouvement et des images à divergence nulle : projection sur Φ^{div} et Ψ . . . 95

5.3.2 Simulation du modèle réduit . . . 96
5.4 Estimation du mouvement de surface de la mer Noire . . . 98
5.4.1 Bases d'ondelettes pour la mer Noire . . . 99
5.4.2 Données océanographiques utilisées pour l'étude . . . 103
5.4.3 Comparaison des simulations complète et réduite . . . 103
5.4.4 Assimilation dans le modèle réduit à divergence nulle . . . 108
5.5 Conclusions . . . 111

6 Conclusion générale et perspectives 113

III Annexes 116

A Composantes de l'algorithme d'assimilation 4D-Var 117
A.1 Calcul du gradient de la fonction de coût associé au système d'assimilation 117
A.2 Algorithme d'assimilation du modèle réduit . . . 119
A.2.1 Description synthétique de l'algorithme . . . 119
A.2.2 Définition des variables et des dimensions . . . 119
A.2.3 Description des runs Forward et Backward . . . 120

B Calcul des modèles tangent et adjoint dans le cas des modèles réduits 121
B.1 Modèle réduit d'advection . . . 121
B.2 Calcul du tangent et de l'adjoint des équations sur b . . . 122

C Intégration numérique des modèles complets 125
C.1 Modèle d'advection du mouvement et des images (nommé AIMI) . . . 125
C.1.1 Définition du vecteur d'état . . . 125
C.1.2 Dynamique du modèle . . . 125
C.1.3 Split des équations d'évolution . . . 126
C.1.4 Schémas de discrétisation . . . 126
C.1.5 Conditions aux bords . . . 127
C.2 Modèle d'advection du mouvement et des images sous la contrainte d'une divergence nulle (AIMIDF) . . . 128
C.2.1 État . . . 128
C.2.2 Dynamique du modèle . . . 129
C.2.3 Schémas de discrétisation . . . 129
C.2.4 Conditions aux bords . . . 129

D Articles 131
D.1 Complément au chapitre 2 : Assimilation d'images dans un modèle réduit pour l'estimation du mouvement. . . . 131
D.2 Complément au chapitre 3 : Learning reduced models for motion estimation on ocean satellite images. . . . 136
D.3 Complément au chapitre 3 : Learning reduced models for motion estimation on long temporal image sequences. . . . 148
D.4 Complément au chapitre 4 : Coupling Reduced Models for Optimal Motion Estimation. . . . 153

D.5 Complément au chapitre 5 : Motion estimation on ocean satellite images by data assimilation in a wavelets reduced model 158

Introduction

Contexte

L'objectif de cette thèse est de concevoir et développer des méthodes d'estimation du mouvement à partir de séquences d'images. L'estimation du mouvement est une des problématiques fondamentales du traitement d'image. Ce problème a été intensivement étudié ces dernières années [24, 26]. Les applications sont diverses et nombreuses : océanographie, météorologie, robotique, surveillance du trafic routier, compression de vidéo ou imagerie médicale. Dans le contexte de l'environnement, l'estimation du mouvement à partir d'images satellite permet, par exemple, d'améliorer les prévisions météorologiques et océanographiques. L'amélioration de ces prévisions peux avoir des enjeux multiples. On peut ainsi citer les enjeux économiques pour le transport et la pêche, ou encore les enjeux de sécurité pour la prévision de certains phénomènes climatiques, tels que les ouragans ou les inondations.

Dans la littérature du traitement d'image, l'estimation du mouvement est le plus souvent réalisée en utilisant la contrainte du flot optique [24], qui suppose le transport des niveaux de gris par le champ de mouvement. Le problème de l'estimation du mouvement, sous cette unique contrainte, est qu'il est un problème mal posé au sens d'Hadamard [17]. Cette contrainte aboutit en effet à un système d'équations sous déterminé, qui se traduit par le problème d'ouverture. Il est donc indispensable, pour obtenir une solution unique, d'ajouter à l'équation du flot optique des contraintes supplémentaires. Des méthodes de régularisation, connues sous le nom de régularisation de Tikhonov [44], ont été proposées : dans [24] la norme du gradient spatial du champ de mouvement est minimisée, dans [26] c'est l'intégrale de la norme de la divergence et du rotationnel du champ de mouvement qui est minimisée. D'autres références de la littérature s'intéressent aux dérivées de la divergence et du rotationnel.

Nous nous intéressons plus particulièrement dans cette thèse à l'estimation du mouvement à partir d'images satellite d'écoulements géophysiques. Dans ce cadre, les méthodes variationnelles d'assimilation de données ont récemment émergées. L'assimilation de données [3, 30] consiste à trouver un compromis entre un modèle et des observations. Dans notre cas, le modèle est un modèle d'évolution du mouvement et des images issu d'heuristiques, et les observations sont les images satellite. L'utilisation de modèles dynamiques, basés sur des connaissances a priori du mouvement, telles que les lois physiques d'un écoulement, permet d'imposer une cohérence temporelle à la solution calculée [2, 22, 45]. De plus, ces modèles étant issus de lois physiques, la contrainte ajoutée au flot optique est moins arbitraire que de nombreuses régularisations de Tikhonov.

Les principales limitations des méthodes d'assimilation d'images sont le temps de calcul et la mémoire nécessaires à leur mise en oeuvre. En effet, ceux-ci sont proportionnels à la taille des images multipliée par la durée de l'intervalle temporel de la séquence. Dans le cas d'images satellite, cette taille peut être très importante : de l'ordre de 10^6 pixels pour une acquisition océanographique sur le bassin de la mer Noire. Ceci limite fortement la mise en œuvre pratique des algorithmes d'assimilation d'images. Afin de contourner ces problèmes, on se propose d'utiliser une technique dite de réduction. La réduction de modèle constitue l'objectif central de cette thèse, afin de limiter le temps de calcul et la mémoire nécessaires à l'estimation du mouvement.

Des modèles réduits, obtenus par projection de Galerkin sur les sous-espaces obtenus par décomposition orthogonale propre, ont été utilisés par D'Adamo et Papadakis [10, 34] pour estimer la dynamique, à partir de champs de mouvement acquis par « Particle Image Velocimetry » [36]. La décomposition orthogonale propre permet d'obtenir des sous-espaces de très faible dimension (en général inférieure à 10) : le temps de calcul et la mémoire nécessaires à la mise en œuvre des algorithmes d'assimilation est ainsi fortement réduit. Ce type de réduction de la dynamique, combiné aux techniques d'assimilation, peut permettre d'obtenir une solution cohérente avec un temps de calcul et une taille mémoire tout à fait raisonnable. Dans la suite du document, on nommera « modèle complet » le modèle issu d'heuristiques et « modèle réduit » la projection de Galerkin du modèle complet sur des sous-espaces du mouvement et des images.

Description du document

Le document est organisé en 3 parties. La première concerne l'estimation du mouvement par assimilation dans des modèles réduits, sur des bases apprises par décomposition orthogonale propre. La seconde concerne les modèles réduits obtenus par projection sur des bases fixes. Enfin la troisième regroupe les annexes. Elle contient les informations les plus techniques, ainsi que les articles publiés au cours de cette thèse.

Première partie

Cette première partie est consacrée aux modèles réduits obtenus par projection de la dynamique sur des sous-espaces de dimension très faible, appris par décomposition orthogonale propre. Ces modèles réduits permettent d'approximer la solution des méthodes d'assimilation d'images, obtenue par un modèle complet, en diminuant significativement le temps de calcul et la mémoire nécessaires.

- Chapitre 1 :
  Ce chapitre présente tout d'abord les grands principes de l'assimilation de données, ainsi que son application à l'assimilation d'images pour l'estimation du mouvement. Ensuite, sont présentés les aspects théoriques de la réduction de modèles dynamiques par projection de Galerkin.
- Chapitre 2 :
  Un modèle réduit, projection de Galerkin d'un modèle complet sur des bases obtenues par décomposition orthogonale propre (POD), est présenté dans ce chapitre.

Nous montrerons l'intérêt d'utiliser un modèle réduit pour estimer le mouvement par assimilation de séquences d'images en un temps très restreint et en utilisant peu de mémoire. Les résultats exposés ont fait l'objet de la publication D.1 présentée en annexe.

- Chapitre 3 :
  Dans ce chapitre, on présente la réduction d'un modèle dynamique à divergence nulle. Tout comme dans le chapitre précédent, le mouvement et les images sont projetés sur des sous-espaces obtenus par décomposition orthogonale propre. Une méthode dite à « fenêtre glissante », qui consiste à coupler l'assimilation dans des modèles complets et réduits pour estimer le mouvement sur de longues séquences d'images est ensuite présentée. Le contenu de ce chapitre correspond à la publication D.3 présentée en annexe.

Seconde partie

Dans cette partie, des modèles réduits sont obtenus par projection de modèle complets sur des bases fixes. Ces bases sont choisies de façon à respecter les propriétés que l'on souhaite imposer à la solution calculée. En effet, comme on le verra par la suite, les estimations du mouvement, obtenues par assimilation dans des modèles réduits, possèdent les mêmes propriétés que les bases utilisées.

- Chapitre 4 :
  Une base de fonctions sinus, dite "fixe", car elle ne dépend pas des données images, est utilisée dans ce chapitre. Le modèle dynamique à divergence nulle présenté dans le chapitre 3 est projeté sur cette nouvelle base. L'objectif est d'imposer de "bonnes" propriétés à la solution en choisissant une base adéquate. Les résultats obtenus avec ces modèles réduits sont ensuite, comme dans le chapitre 3, couplés avec des modèles réduits POD dans une expérience dite à « fenêtre glissante » afin d'estimer le mouvement sur de longues séquences d'images. Les algorithmes sont testés sur des images satellite et comparés à des méthodes de l'état de l'art.
- Chapitre 5 :
  De nouveau types de bases fixes d'ondelettes sont présentés dans ce chapitre. Celles-ci possèdent des caractéristiques particulièrement adaptées aux images satellite bruitées et à l'estimation du mouvement sur des domaines à géométrie complexe, tel que celui de la mer Noire. Les bases d'ondelettes sont calculées automatiquement à partir du domaine spatial et de contraintes choisies par l'utilisateur, que nous détaillerons dans le document.

Enfin une conclusion générale est consacrée à la synthèse de tous les résultats obtenus et présente quelques perspectives pour de futurs travaux.

Première partie

Estimation du mouvement par assimilation de données dans des modèles réduits

Chapitre 1

Assimilation d'images pour l'estimation du mouvement

1.1 Introduction

On décrit dans ce chapitre les grands principes et le cadre méthodologique utilisé dans cette thèse pour estimer le mouvement $\mathbf{w}(\mathbf{x}, t)$ à partir d'une séquence d'images discrète $I = \{I^z\}_z = \{I(\mathbf{x}, t_z)\}_z$. Comme décrit dans l'introduction, les méthodes d'assimilation de données ont récemment émergées [2, 22, 45] pour l'estimation du mouvement fluide. C'est sur cette approche que repose le contenu de cette thèse. Puisque les problèmes de traitement d'image sont souvent mal posés, c'est le cas de l'estimation du mouvement, l'assimilation de données permet d'ajouter des contraintes supplémentaires au système à résoudre afin d'aboutir à une solution unique. Dans cette thèse, les contraintes concernent l'évolution temporelle des données, qui est prise en compte à travers un système d'évolution issu d'heuristiques.

L'assimilation de données regroupe un ensemble de méthodes et d'algorithmes qui ont pour but de faire un compromis entre un modèle et des observations. Certaines, comme le filtre de Kalman [27] et le filtre de Kalman-Bucy [28], sont séquentielles. D'autres types de méthodes sont variationnelles [13], par exemple les méthodes dites 4D-Var. Ces dernières incorporent des observations pour ajuster l'état du système sur un intervalle temporel plutôt que de corriger cet état à chaque date d'observation. L'avantage qui en résulte est de calculer une solution cohérente sur l'ensemble de l'intervalle plutôt que d'introduire des discontinuités en corrigeant ponctuellement l'état à chaque date où des observations sont disponibles.

Dans la suite du document, afin d'estimer le mouvement à partir d'une séquence d'images, nous utilisons des méthodes variationelles, et particulièrement l'algorithme « 4D-Var » [30].

Un système d'assimilation variationnelle est composé de trois équations :

- une équation d'évolution, qui décrit l'évolution du vecteur d'état en utilisant un opérateur nommé « modèle d'évolution »,
- une équation d'observation, qui lie le vecteur d'état aux observations au moyen d'un « opérateur d'observation »,
- et enfin une équation d'ébauche sur la condition initiale, qui traduit les heuristiques dont on dispose sur cette condition initiale.

La section 1.2 décrit les trois équations du système d'assimilation de données. On précise, dans la section 1.3, le système d'équations dans le cadre de l'estimation du mouvement pour un modèle particulier, utilisé pour estimer le mouvement à partir d'une séquence d'images. On présente brièvement quelques résultats obtenus par assimilation dans la section 1.4. Enfin, la réduction de modèle par projection de Galerkin ainsi qu'un système d'assimilation pour la dynamique réduite est présenté dans la section 1.5.

1.2 Système d'assimilation

Équation d'évolution

Soit $\mathbf{X}$ le vecteur d'état dépendant de $(\mathbf{x}, t)$ et défini sur le domaine spatio-temporel $A = \Omega \times [0, T]$. L'objectif est de déterminer $\mathbf{X}$ sur tout le domaine A. On fait l'hypothèse que $\mathbf{X}$ est solution de l'équation d'évolution :

$$\frac{\partial \mathbf{X}}{\partial t}(\mathbf{x}, t) + \mathbb{M}(\mathbf{X})(\mathbf{x}, t) = 0, \tag{1.2.1}$$

Le modèle d'évolution $\mathbb{M}$, décrivant la dynamique de $\mathbf{X}(\mathbf{x}, t)$, est supposé différentiable. Dans un cadre plus général, on peut aussi considérer que $\mathbb{M}$ n'est qu'une approximation de la dynamique de $\mathbf{X}$ et ajouter à l'équation 1.2.1 un terme d'erreur modèle $\mathcal{E}_m(\mathbf{x}, t)$:

$$\frac{\partial \mathbf{X}}{\partial t}(\mathbf{x}, t) + \mathbb{M}(\mathbf{X})(\mathbf{x}, t) = \mathcal{E}_m(\mathbf{x}, t), \tag{1.2.2}$$

Lorsque l'équation d'évolution ne comporte pas d'erreur modèle (équation 1.2.1), on parle de modèle parfait et de contrainte dynamique forte. Lorsqu'il comporte une erreur modèle (équation 1.2.2), on parle de modèle imparfait et de contrainte dynamique faible. Dans la suite du document, on se focalisera sur les systèmes d'assimilation sous contrainte dynamique forte. Une description plus exhaustive de l'assimilation de données, comprenant le cas de la contrainte dynamique faible, peut être trouvée dans [2].

Équation d'observation

Soit $\mathbf{Y}(\mathbf{x}, t)$ une observation disponible en $\mathbf{x}$ à la date t. Cette observation est liée au vecteur d'état à travers l'équation d'observation :

$$\mathbf{Y}(\mathbf{x}, t) = \mathbb{H}\left(\mathbf{X}\right)(\mathbf{x}, t) + \mathcal{E}_O(\mathbf{x}, t) \tag{1.2.3}$$

$\mathbb{H}$ est l'opérateur d'observation, il est supposé différentiable. $\mathcal{E}_o$ est l'erreur d'observation. Cette erreur représente à la fois l'erreur liée à la mesure $\mathbf{Y}$ et l'erreur associée à l'opérateur $\mathbb{H}$. Dans notre cas, l'opérateur d'observation sera systématiquement linéaire et se réduira à une projection, que l'on notera $\mathbb{H}\mathbf{X}$.

Équation d'ébauche

On suppose avoir une connaissance approximative $\mathbf{X}_b$ de la condition initiale $\mathbf{X}(\mathbf{x}, 0)$. $\mathbf{X}_b$ est appelés ébauche. On considère l'équation d'ébauche :

$$\mathbf{X}(\mathbf{x}, 0) = \mathbf{X}_b(\mathbf{x}) + \mathcal{E}_b(\mathbf{x}) \tag{1.2.4}$$

avec $\mathcal{E}_b$ l'erreur d'ébauche. Cette équation représente la connaissance imparfaite de l'état initial du système dont dispose l'utilisateur.

Le système à résoudre pour estimer $\mathbf{X}(\mathbf{x}, t)$ est composé des équations 1.2.1, 1.2.3 et 1.2.4 et s'écrit :

$$\begin{cases} \dfrac{\partial \mathbf{X}}{\partial t}(\mathbf{x}, t) + \mathbb{M}(\mathbf{X})(\mathbf{x}, t) = 0, \\ \mathbf{Y}(\mathbf{x}, t) = \mathbb{H}\mathbf{X}(\mathbf{x}, t) + \mathcal{E}_O(\mathbf{x}, t) \\ \mathbf{X}(\mathbf{x}, 0) = \mathbf{X}_b(\mathbf{x}) + \mathcal{E}_b(\mathbf{x}) \end{cases} \tag{1.2.5}$$

On fait l'hypothèse que les erreurs $\mathcal{E}_O(\mathbf{x}, t)$ et $\mathcal{E}_b(\mathbf{x})$ sont gaussiennes de moyenne nulle, que $\mathcal{E}_O(\mathbf{x}, t)$ est décorrélée en temps et en espace et que $\mathcal{E}_b(\mathbf{x})$ est décorrélée en espace. Elles sont caractérisées par leurs matrices de covariance respectives $R(\mathbf{x}, t)$ et $B(\mathbf{x})$.

Fonction de coût

La résolution du système 1.2.5 se fait en minimisant les erreurs $\mathcal{E}_O(\mathbf{x}, t)$ et $\mathcal{E}_b(\mathbf{x})$. Pour cela, on minimise la fonction de coût suivante $J[\mathbf{X}(\mathbf{x}, 0)]$:

$$J[\mathbf{X}(\mathbf{x}, 0)] = \frac{1}{2}\int_\Omega (\mathbf{X}(\mathbf{x}, 0) - \mathbf{X}_b(\mathbf{x}))^T B^{-1}(\mathbf{x}) (\mathbf{X}(\mathbf{x}, 0) - \mathbf{X}_b(\mathbf{x})) \, d\mathbf{x} \tag{1.2.6}$$

$$+\frac{1}{2}\int_A [\mathbb{H}\mathbf{X} - \mathbf{Y}]^T(\mathbf{x}, t) R^{-1}(\mathbf{x}, t) [\mathbb{H}\mathbf{X} - \mathbf{Y}](\mathbf{x}, t) d\mathbf{x}dt \tag{1.2.7}$$

La fonction de coût ne dépend que de $\mathbf{X}(\mathbf{x}, 0)$ car le système d'assimilation est sous contrainte dynamique forte : une fois $\mathbf{X}(\mathbf{x}, 0)$ déterminé, toute la trajectoire $\mathbf{X}(\mathbf{x}, t)$ est connue en intégrant l'équation 1.2.1. Le premier terme de J est appelé terme d'ébauche et le second terme d'observation. La minimisation de la fonction de coût se fait en utilisant le minimiseur L-BFGS. Son utilisation requiert le calcul de $J[\mathbf{X}(\mathbf{x}, 0)]$ et de son gradient. Le calcul analytique du gradient par la méthode adjointe ainsi que les détails algorithmiques de la minimisation sont décrit dans l'annexe A. L'expression de ce gradient est la suivante :

$$\nabla J[\mathbf{X}(\mathbf{x}, 0)] = B^{-1}(\mathbf{x}) (\mathbf{X}(\mathbf{x}, 0) - \mathbf{X}_b(\mathbf{x})) + \lambda(\mathbf{x}, 0) \tag{1.2.8}$$

où $\lambda(\mathbf{x}, t)$ est la variable adjointe vérifiant le système suivant :

$$\begin{aligned} -\frac{\partial \lambda}{\partial t}(\mathbf{x}, t) + \left(\frac{\partial \mathbb{M}}{\partial \mathbf{X}}\right)^* \lambda(\mathbf{x}, t) &= \mathbb{H}^T R^{-1}(\mathbf{x}, t) [\mathbb{H}\mathbf{X} - \mathbf{Y}](\mathbf{x}, t) \\ \lambda(\mathbf{x}, T) &= 0 \end{aligned} \tag{1.2.9}$$

Le calcul de ce gradient nécessite l'intégration directe du modèle selon l'équation 1.2.1, puis l'intégration rétrograde de la variable adjointe selon l'équation 1.2.9 afin de calculer la valeur de λ à $t = 0$. Une fois les valeurs de J et ∇J calculées, le minimiseur L-BFGS (Limited memory Broyden-Fletcher-Goldfarb-Shanno [47]) est utilisé afin de mettre à jour $\mathbf{X}(\mathbf{x}, 0)$. Le processus est itéré jusqu'à convergence du minimiseur.

1.3 Modèle d'advection du mouvement et des images

Dans cette section, on explicite, pour un modèle particulier, la méthode d'assimilation présentée dans la section 1.2. On rappelle que l'objectif est d'estimer le mouvement $\mathbf{w}(\mathbf{x}, t)$ à partir d'une séquence d'images $I = \{I^z\}_z = \{I(\mathbf{x}, t_z)\}_z$.

Le modèle utilisé est nommé AIMI, pour « Advection Image Model Intern ». Il s'agit d'un modèle d'advection du mouvement et des images. Le vecteur d'état est composé du mouvement et d'un scalaire $I_s(\mathbf{x}, t)$, définis sur le domaine spatio-temporel $A = \Omega \times [0, T]$. Ce scalaire, I_s nommé « pseudo-image », est une quantité mathématique similaire aux images observées I^z.

Les équations du modèle sont les suivantes :

$$\begin{cases} \dfrac{\partial \mathbf{w}}{\partial t}(\mathbf{x}, t) + (\mathbf{w} \cdot \nabla)\, \mathbf{w}(\mathbf{x}, t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x}, t) + \mathbf{w} \cdot \nabla I_s(\mathbf{x}, t) = 0 \end{cases} \tag{1.3.1}$$

Ce modèle suppose l'advection des images par le mouvement et l'advection du mouvement par lui même. Les qualités du modèle AIMI, les hypothèses sur le mouvement sous-jacent ainsi que le choix d'inclure I_s dans le vecteur d'état sont discutées dans [1, 19] et seront explicitées dans le chapitre 2. Les schémas numériques et les conditions aux bords utilisés pour la simulation de ce modèle sont décrits dans l'annexe C.

1.4 Résultats d'assimilation

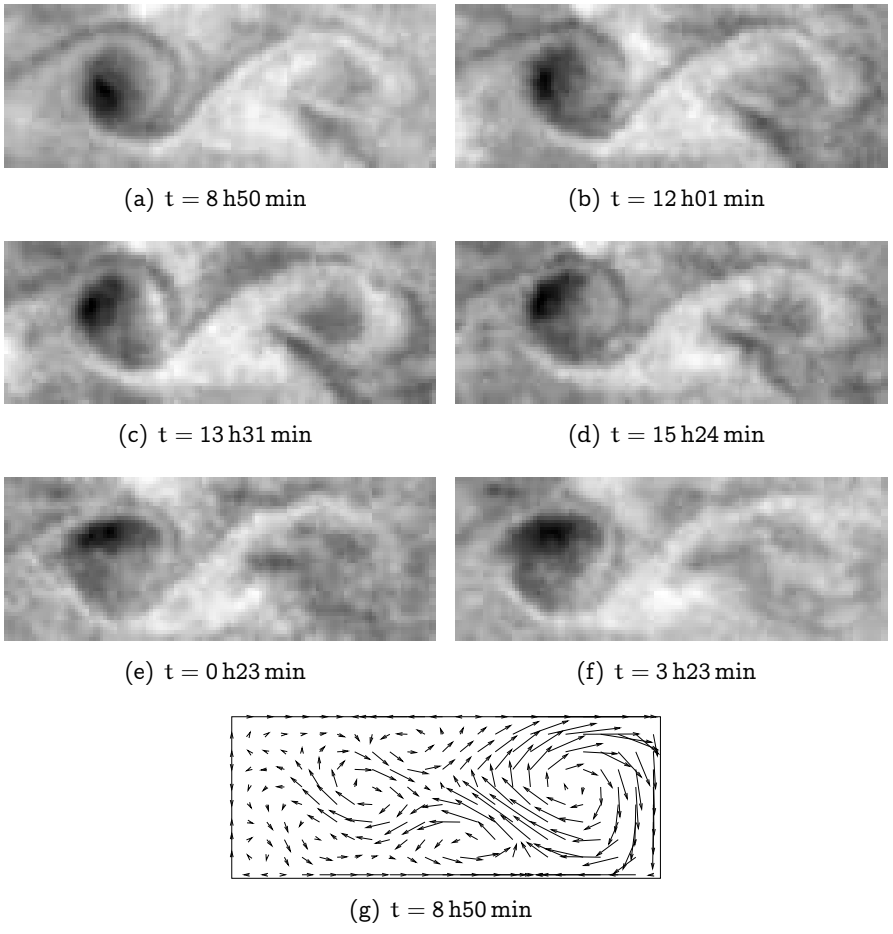
Le système d'assimilation 1.2.5 s'écrit alors pour le modèle AIMI (système 1.3.1) :

$$\begin{cases} \dfrac{\partial \mathbf{w}}{\partial t}(\mathbf{x}, t) + (\mathbf{w} \cdot \nabla)\, \mathbf{w}(\mathbf{x}, t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x}, t) + \mathbf{w} \cdot \nabla I_s(\mathbf{x}, t) = 0 \\ I^z(\mathbf{x}, t_z) = \mathbb{H}\mathbf{X}(\mathbf{x}, t_z) + \mathcal{E}_O(\mathbf{x}, t_z) \\ \mathbf{X}(\mathbf{x}, 0) = \mathbf{X}_b(\mathbf{x}) + \mathcal{E}_b(\mathbf{x}) \end{cases} \tag{1.4.1}$$

Les observations $I^z(\mathbf{x}, t_z)$ sont constituées de la séquence d'images I et l'opérateur $\mathbb{H}$ est le suivant : $\mathbb{H} : \mathbf{X}(\mathbf{x}, t) \rightarrow I_s(\mathbf{x}, t)$. L'ébauche $\mathbf{X}_b$ est nulle pour le mouvement car on ne dispose d'aucune information a priori sur ce mouvement. On choisit la première observation comme ébauche pour I_s, car la fenêtre temporelle choisie pour l'assimilation commence juste avant la première observation. L'ébauche s'écrit alors : $\mathbf{X}_b = \begin{pmatrix} 0 & I^1 \end{pmatrix}^T$ et le système 1.4.1 se réécrit :

$$\begin{cases} \dfrac{\partial \mathbf{w}}{\partial t}(\mathbf{x},t) + (\mathbf{w}\cdot\nabla)\,\mathbf{w}(\mathbf{x},t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x},t) + \mathbf{w}\cdot\nabla I_s(\mathbf{x},t) = 0 \\ I^z(\mathbf{x},t_z) = I_s(\mathbf{x},t_z) + \mathcal{E}_O(\mathbf{x},t_z) \\ I_s(\mathbf{x},0) = I^1(\mathbf{x},t_1) + \mathcal{E}_b(\mathbf{x}) \end{cases} \tag{1.4.2}$$

On présente, sur la figure 1.1, une séquence de 6 images satellite ainsi que le résultat d'assimilation pour le mouvement. La séquence utilisée est composée d'images de température de surface (SST, Sea Surface Temperature), acquises par le satellite NOAA/AVHRR sur la mer Noire les 14 et 15 mai 2005. La résolution spatiale est d'environ 1 km et la fenêtre temporelle est de 18 heures et 33 minutes.



(a) t = 8 h50 min (b) t = 12 h01 min

(c) t = 13 h31 min (d) t = 15 h24 min

(e) t = 0 h23 min (f) t = 3 h23 min

(g) t = 8 h50 min

FIGURE 1.1 – Observations et résultat de mouvement obtenu au début de la fenêtre temporelle.

Comme on peut le constater, l'estimation du mouvement par assimilation d'images permet d'obtenir de bons résultats. Les avantages de ces algorithmes sont discutés dans [2, 6, 39, 45].

Ces méthodes souffrent cependant d'inconvénients majeurs : le temps de calcul et la mémoire nécessaires à leur mise en œuvre sont proportionnels à la taille des images multipliée par la durée de l'intervalle temporel. Dans le cas d'images satellite SST pour le bassin de la mer Noire, par exemple, l'ordre de grandeur est de 10^6 pixels, ce qui rend très difficile l'estimation du mouvement par assimilation de données.

Un moyen de contourner ces inconvénients est d'appliquer une technique de réduction du modèle. La dynamique est projetée sur un sous-espace de dimension très inférieure à la taille de l'espace image. Cela permet d'assimiler les images en un temps de calcul beaucoup plus raisonnable et en utilisant beaucoup moins de mémoire. On présente dans la section 1.5 le principe de la projection de Galerkin d'un modèle sur un sous-espace. La projection de Galerkin sera utilisée dans la suite du document pour définir des méthodes d'estimation du mouvement.

1.5 Projection de Galerkin d'un modèle dynamique

Dans cette section, on présente le principe de la projection de Galerkin d'un modèle d'évolution $\mathbb{M}$ sur un sous-espace. On nomme dorénavant $\mathbb{M}$ le modèle complet, et sa projection de Galerkin $\mathbb{M}_R$ est nommée modèle réduit.

Le modèle complet $\mathbb{M}$ décrit l'évolution du vecteur d'état $\mathbf{X}(x,t)$ (selon l'équation 1.5.1). $\mathbf{X}$ est défini sur le domaine spatio-temporel $A = \Omega \times [0,T]$ et vérifie l'équation d'évolution :

$$\frac{\partial \mathbf{X}}{\partial t}(x,t) + \mathbb{M}(\mathbf{X})(x,t) = 0 \tag{1.5.1}$$

Soient :
- $\Phi = \{\phi_k(x)\}_{k=1\ldots K}$ une base orthogonale de dimension K,
- P^K la projection orthogonale :

$$P^K : \mathbf{X}(t) \quad \rightarrow P^K(\mathbf{X})(t) = \sum_{i=1}^{K} \frac{\langle X(t), \phi_i \rangle}{\langle \phi_i, \phi_i \rangle} \phi_i$$

avec $< .,. >$ le produit scalaire défini par $E : < f,g >= \int_\Omega f(x)g(x)dx$

On note $a_i(t) = \dfrac{\langle \mathbf{X}(t), \phi_i \rangle}{\langle \phi_i, \phi_i \rangle}$ le coefficient de projection de $\mathbf{X}(t)$ sur ϕ_i.

On a donc :

$$P^K(\mathbf{X})(x,t) = \sum_{i=1}^{K} a_i(t)\phi_i(x), \quad \forall (x,t) \in A \tag{1.5.2}$$

La projection de Galerkin $\mathbb{M}_R$ de $\mathbb{M}$ sur Φ est définie par l'équation 1.5.3 :

$$P^K \left[\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}\left(P^K(\mathbf{X})\right) \right] = 0 \tag{1.5.3}$$

On peut donc, à partir d'un modèle complet $\mathbb{M}$ et d'une base orthogonale Φ, définir un modèle réduit. Afin que le modèle réduit soit une bonne approximation du modèle complet $\mathbb{M}$, il est nécessaire que la trajectoire du vecteur d'état $\mathbf{X}(x,t)$ soit proche du sous-espace défini par la base Φ. Il faut donc avoir des informations sur les trajectoires $\mathbf{X}(x,t)$ pour définir un bon modèle réduit.

1.6 Conclusion

La projection de Galerkin du modèle AIMI (système 1.3.1) sur un sous-espace engendré par une base Φ, ainsi que l'utilisation du modèle réduit obtenu pour l'estimation du mouvement seront présentées de façon plus complète au chapitre 2. Dans ce chapitre 1, nous avons présenté les grands principes de l'estimation du mouvement par assimilation d'images. Les avantages et inconvénients de ces méthodes ont étés décrits. Nous avons présenté succinctement une technique de réduction de modèle afin de contourner les principales difficultés de l'assimilation d'images, à savoir le temps de calcul et la mémoire nécessaires. Les chapitres 2 à 5 sont consacrés à l'utilisation de la réduction de modèle pour l'estimation du mouvement par assimilation de données sur des données synthétiques et réelles.

Chapitre 2

Apprentissage de base par décomposition orthogonale propre

2.1 Introduction

L'estimation du mouvement $\mathbf{w}(\mathbf{x}, t)$, à partir d'une séquence d'images $I(\mathbf{x}, t)$, par des algorithmes d'assimilation de données a fait l'objet d'un important effort de recherche ces dernières années [2, 6, 45]. Ces méthodes, que nous avons brièvement présentées dans le chapitre 1, ont pour principales limitations le temps de calcul et la mémoire nécessaire à leur utilisation. La complexité de ces algorithmes est proportionelle à la taille des images multipliée par la durée de la fenêtre temporelle. Un moyen de contourner ces difficultés d'utilisation est d'utiliser une technique dite de réduction : la projection du modèle complet $\mathbb{M}$ sur un sous-espace de faible dimension permet de réduire considérablement le temps de calcul et la mémoire nécessaire.

Le sous-espace de projection peut être, par exemple, obtenu par décomposition orthogonale ("Proper Orthogonal decomposition" ou POD). Également connue sous le nom d'Analyse en Composantes Principales (ACP) ou décomposition de Karhunen-Loève [31], cette technique a été en premier lieu introduite pour la mécanique des fluides par Lumley [32]. Elle a depuis été largement utilisée pour approximer la description de divers écoulements [11, 23, 37, 38]. Un modèle réduit, obtenu par projection de Galerkin, a ainsi été utilisé par D'Adamo et Papadakis dans [10, 35] pour estimer la dynamique réduite d'un écoulement, par assimilation de données, à partir d'observations du champ de mouvement acquises par « Particle Image Velocimetry » [36]. Ce modèle dynamique réduit décrit l'évolution temporelle des cœfficients de projection $a_i(t)$ du mouvement $\mathbf{w}(\mathbf{x}, t)$ sur une base $\Phi = \{\phi_i(\mathbf{x})\}_{i=1\ldots K}$ afin d'approximer l'évolution temporelle du champ de vitesse $\mathbf{w}(\mathbf{x}, t)$.

Dans ce chapitre, on souhaite estimer un mouvement $\mathbf{w}(\mathbf{x}, t)$, à partir d'une séquence discrète d'images $I = \{I^z\}_z = \{I(\mathbf{x}, t_z)\}_z$. On considère un modèle complet $\mathbb{M}$ décrivant les lois d'évolution du mouvement et des images. Un modèle réduit $\mathbb{M}_R$ est défini par projection de Galerkin de $\mathbb{M}$ sur des sous-espaces obtenus par POD. Une base Φ est définie pour le mouvement et une base Ψ pour les images. Le mouvement $\mathbf{w}(\mathbf{x}, t)$ est représenté par ses cœfficients de projection $a_i(t)$ sur la base Φ, et les images par leurs cœfficients de projections $b_j(t)$ sur la base Ψ. Pour que le modèle réduit soit une bonne approximation du modèle complet, il faut que les sous-espaces de projection soient tels

que :

- l'erreur de projection de la solution est faible, et
- la dynamique du modèle complet est bien approximée.

Après avoir défini les sous-espaces, pour le mouvement et les images, un algorithme d'assimilation dans le modèle réduit $\mathbb{M}_R$ est conçu et permet d'assimiler les cœfficients des images afin d'estimer les cœfficients du mouvement.

Dans la section 2.2, la projection du modèle complet $\mathbb{M}$ d'advection du mouvement et des images, que nous appelons AIMI, est explicitée, afin d'illustrer la démarche de projection. De même, la détermination des sous-espaces de projection par POD est décrite dans la section 2.3 afin de permettre au lecteur d'appréhender la globalité de la démarche. Des expériences de simulation et d'assimilation dans le modèle réduit sont présentées dans les sections 2.4 et 2.5.

2.2 Projection de Galerkin du modèle d'advection image

2.2.1 Rappel des équations

On rappelle les équations 2.2.1 du modèle AIMI, qui suppose l'advection de la vitesse $\mathbf{w}$ et de la pseudo-image I_s par la vitesse :

$$\begin{cases} \dfrac{\partial \mathbf{w}}{\partial t}(\mathbf{x},t) + (\mathbf{w}\cdot\nabla)\,\mathbf{w}(\mathbf{x},t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x},t) + \mathbf{w}\cdot\nabla I_s(\mathbf{x},t) = 0 \end{cases} \tag{2.2.1}$$

$\mathbf{X} = \begin{pmatrix} \mathbf{w} \\ I_s \end{pmatrix}$ est le vecteur d'état du modèle et le système 2.2.1 se réécrit :

$$\frac{\partial \mathbf{X}}{\partial t}(\mathbf{x},t) + \mathbb{M}(\mathbf{X})(\mathbf{x},t) = 0 \tag{2.2.2}$$

2.2.2 Projection des équations

On approxime $\mathbf{w}$ et I_s par leurs projections sur les sous-espaces engendrés par les bases orthogonales $\Phi = \{\phi_i(\mathbf{x})\}_{i=1\ldots K}$ et $\Psi = \{\psi_j(\mathbf{x})\}_{j=1\ldots L}$:

$$\begin{cases} \mathbf{w}(\mathbf{x},t) & \approx & \displaystyle\sum_{i=1}^{K} a_i(t)\phi_i(\mathbf{x}) \\ I_s(\mathbf{x},t) & \approx & \displaystyle\sum_{j=1}^{L} b_j(t)\psi_j(\mathbf{x}) \end{cases} \tag{2.2.3}$$

Il s'agit de calculer la projection de Galerkin de $\mathbb{M}$ sur le sous-espace engendré par le produit des bases $\Phi \times \Psi$. Pour cela, on remplace $\mathbf{w}$ et I_s par leurs approximations (selon le système 2.2.3) dans le système 2.2.1 :

$$\begin{cases} \dfrac{\partial}{\partial t}\left(\displaystyle\sum_{i=1}^{K} a_i(t)\phi_i(x)\right) + \left[\left(\displaystyle\sum_{i=1}^{K} a_i(t)\phi_i(x)\right)\cdot\nabla\right]\left(\displaystyle\sum_{j=1}^{K} a_j(t)\phi_j(x)\right) \approx 0 \\ \dfrac{\partial}{\partial t}\left(\displaystyle\sum_{j=1}^{L} b_j(t)\psi_j(x)\right) + \left(\displaystyle\sum_{i=1}^{K} a_i(t)\phi_i(x)\right)\cdot\nabla\left(\displaystyle\sum_{j=1}^{L} b_j(t)\psi_j(x)\right) \approx 0 \end{cases} \tag{2.2.4}$$

On obtient par linéarité :

$$\begin{cases} \displaystyle\sum_{i=1}^{K} \frac{da_i}{dt}(t)\phi_i(x) + \sum_{i=1}^{K} a_i(t)\left[(\phi_i(x)\cdot\nabla)\left(\sum_{j=1}^{K} a_j(t)\phi_j(x)\right)\right] \approx 0 \\ \displaystyle\sum_{j=1}^{L} \frac{db_j}{dt}(t)\psi_j(x) + \sum_{i=1}^{K} a_i(t)\left[\phi_i(x)\cdot\nabla\left(\sum_{j=1}^{L} b_j(t)\psi_j(x)\right)\right] \approx 0 \end{cases} \tag{2.2.5}$$

$$\begin{cases} \displaystyle\sum_{i=1}^{K} \frac{da_i}{dt}(t)\phi_i(x) + \sum_{i=1}^{K}\sum_{j=1}^{K} a_i(t)a_j(t)\left(\phi_i(x)\cdot\nabla\right)\phi_j(x) \approx 0 \\ \displaystyle\sum_{j=1}^{L} \frac{db_j}{dt}(t)\psi_j(x) + \sum_{i=1}^{K}\sum_{j=1}^{L} a_i(t)b_j(t)\phi_i(x)\cdot\nabla\psi_j(x) \approx 0 \end{cases} \tag{2.2.6}$$

Ensuite, on projette la première équation du système 2.2.6 sur ϕ_k pour $k = 1 \dots K$ et la seconde sur ψ_l pour $l = 1 \dots L$:

$$\begin{cases} \left\langle \displaystyle\sum_{i=1}^{K} \frac{da_i}{dt}(t)\phi_i + \sum_{i=1}^{K}\sum_{j=1}^{K} a_i(t)a_j(t)\left(\phi_i\cdot\nabla\right)\phi_j, \phi_k \right\rangle = 0, \quad k = 1 \dots K \\ \left\langle \displaystyle\sum_{j=1}^{L} \frac{db_j}{dt}(t)\psi_j + \sum_{i=1}^{K}\sum_{j=1}^{L} a_i(t)b_j(t)\phi_i\cdot\nabla\psi_j, \psi_l \right\rangle = 0, \quad l = 1 \dots L \end{cases} \tag{2.2.7}$$

En utilisant la linéarité du produit scalaire on obtient :

$$\begin{cases} \displaystyle\sum_{i=1}^{K} \frac{da_i}{dt}(t)\left\langle\phi_i, \phi_k\right\rangle + \sum_{i=1}^{K}\sum_{j=1}^{K} a_i(t)a_j(t)\left\langle\left(\phi_i\cdot\nabla\right)\phi_j, \phi_k\right\rangle = 0, \quad k = 1 \dots K \\ \displaystyle\sum_{j=1}^{L} \frac{db_j}{dt}(t)\left\langle\psi_j, \psi_l\right\rangle + \sum_{i=1}^{K}\sum_{j=1}^{L} a_i(t)b_j(t)\left\langle\phi_i\cdot\nabla\psi_j, \psi_l\right\rangle = 0 \quad l = 1 \dots L \end{cases} \tag{2.2.8}$$

Les bases Φ et Ψ étant orthogonales, les équations se simplifient en :

$$\begin{cases} \dfrac{da_k}{dt}(t)\,\langle \phi_k, \phi_k\rangle + \displaystyle\sum_{i=1}^{K}\sum_{j=1}^{K} a_i(t)a_j(t)\,\langle (\phi_i\cdot\nabla)\,\phi_j, \phi_k\rangle = 0, \quad k = 1\dots K \\ \dfrac{db_l}{dt}(t)\,\langle \psi_l, \psi_l\rangle + \displaystyle\sum_{i=1}^{K}\sum_{j=1}^{L} a_i(t)b_j(t)\,\langle \phi_i\cdot\nabla\psi_j, \psi_l\rangle = 0 \quad l = 1\dots L \end{cases} \tag{2.2.9}$$

ou encore, en divisant respectivement les deux équations par $\langle \phi_k, \phi_k\rangle$ et $\langle \psi_l, \psi_l\rangle$ afin de les normaliser :

$$\begin{cases} \dfrac{da_k}{dt}(t) + \displaystyle\sum_{i=1}^{K}\sum_{j=1}^{K} a_i(t)a_j(t)\dfrac{\langle (\phi_i\cdot\nabla)\,\phi_j, \phi_k\rangle}{\langle \phi_k, \phi_k\rangle} = 0, \quad k = 1\dots K \\ \dfrac{db_l}{dt}(t) + \displaystyle\sum_{i=1}^{K}\sum_{j=1}^{L} a_i(t)b_j(t)\dfrac{\langle \phi_i\cdot\nabla\psi_j, \psi_l\rangle}{\langle \psi_l, \psi_l\rangle} = 0 \quad l = 1\dots L \end{cases} \tag{2.2.10}$$

On définit les grandeurs suivantes :

- $a(t) = \begin{pmatrix} a_1(t) & \dots & a_K(t)\end{pmatrix}^T$ et $b(t) = \begin{pmatrix} b_1(t) & \dots & b_L(t)\end{pmatrix}^T$ les vecteurs de cœfficients de taille respective K et L,
- $B(k)$ la matrice de taille $K \times K$ dont l'élément (i,j) est :

$$B(k)_{i,j} = \frac{\langle (\phi_i\cdot\nabla)\phi_j, \phi_k\rangle}{\langle \phi_k, \phi_k\rangle},$$

- $G(l)$ la matrice de taille $K \times L$ dont l'élément (i,j) est :

$$G(l)_{i,j} = \frac{\langle \phi_i\cdot\nabla\psi_j, \psi_l\rangle}{\langle \psi_l, \psi_l\rangle}.$$

Le système 2.2.10 se réécrit alors :

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T B(k)a = 0, \quad k = 1\dots K \\ \dfrac{db_l}{dt}(t) + a^T G(l)b = 0, \quad l = 1\dots L \end{cases} \tag{2.2.11}$$

On définit aussi $\mathbf{X}_R(t) = \begin{pmatrix} a(t) \\ b(t)\end{pmatrix}$ le vecteur d'état du modèle réduit et le système d'évolution 2.2.11 devient :

$$\frac{dX_R}{dt}(t) + \mathbb{M}_R(X_R)(t) = 0. \tag{2.2.12}$$

L'opérateur $\mathbb{M}_R$, qui est ainsi défini, est la **projection de Galerkin** de $\mathbb{M}$ sur les bases Φ et Ψ.

2.3 Bases réduites obtenues par décomposition orthogonale propre

On présente, dans cette section, l'obtention des bases Φ et Ψ au moyen de la décomposition orthogonale propre (encore appelée « Proper Orthogonal Decomposition »ou POD dans ce document). Dans ce cas, on suppose qu'on dispose de séquences discrètes de champs de mouvement, $W = \{\mathbf{w}^z(\mathbf{x})\}_z$, et d'images, $I = \{I_s^z(\mathbf{x})\}_z$.

Dans la sous-section 2.3.1, les séquences de mouvement et d'images sont obtenues par une simulation du modèle complet $\mathbb{M}$ à partir des conditions initiales, selon l'équation 2.2.2. On décrit le calcul des bases dans la sous-section 2.3.2, et on compare ensuite les résultats de simulation du modèle complet à sa projection sur les bases Φ et Ψ dans la sous-section 2.3.3.

2.3.1 Simulation du modèle complet

On considère le système d'évolution du modèle complet AIMI :

$$\begin{cases} \dfrac{\partial \mathbf{w}}{\partial t}(\mathbf{x},t) + (\mathbf{w}\cdot\nabla)\,\mathbf{w}(\mathbf{x},t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x},t) + \mathbf{w}\cdot\nabla I_s(\mathbf{x},t) = 0 \\ \mathbf{w}(\mathbf{x},0) = \mathbf{w}_0(\mathbf{x}), \\ I_s(\mathbf{x},0) = I_0(\mathbf{x}) \end{cases} \tag{2.3.1}$$

Le modèle complet AIMI, décrit par le système 2.3.1, est simulé, à partir des conditions initiales $\mathbf{w}_0$ et I_0 représentées sur la figure 2.1. $\mathbf{w}_0$ est représenté en pseudo-couleurs à l'aide de l'outil de représentation de la « Middlebury database [1] ». En chaque pixel, la teinte représente la direction du vecteur vitesse et son intensité représente la norme du vecteur vitesse.

Pour réaliser cette simulation, le modèle 2.3.1 est discrétisé et les paramètres de la simulation sont les suivants :

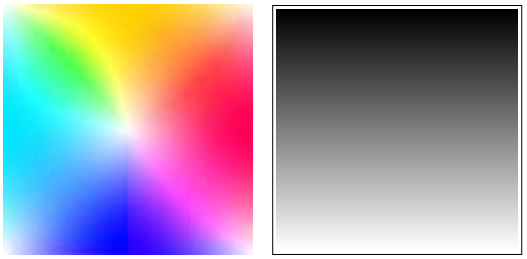
- $dx = dy = 1\,m$,
- $dt = 0.01\,s$,
- $Nx = Ny = 128$ soit $128\,$mètres,
- $Nt = 700$ soit $7\,$secondes.

La simulation fournit les séquences $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$, dont quelques images sont présentées sur la figure 2.2. On choisit les dates $t_z = 0\,s, 0.5\,s \ldots, 7\,s$ pour former les séquences discrètes $W = \{\mathbf{w}^z\}_z = \{\mathbf{w}(\mathbf{x},t_z)\}_z$ et $I = \{I^z\}_z = \{I(\mathbf{x},t_z)\}_z$, qui ont donc 13 éléments. L'obtention des bases Φ et Ψ au moyen de la POD à partir des séquences W et I est décrite dans la sous-section 2.3.2.

2.3.2 Décomposition orthogonale propre

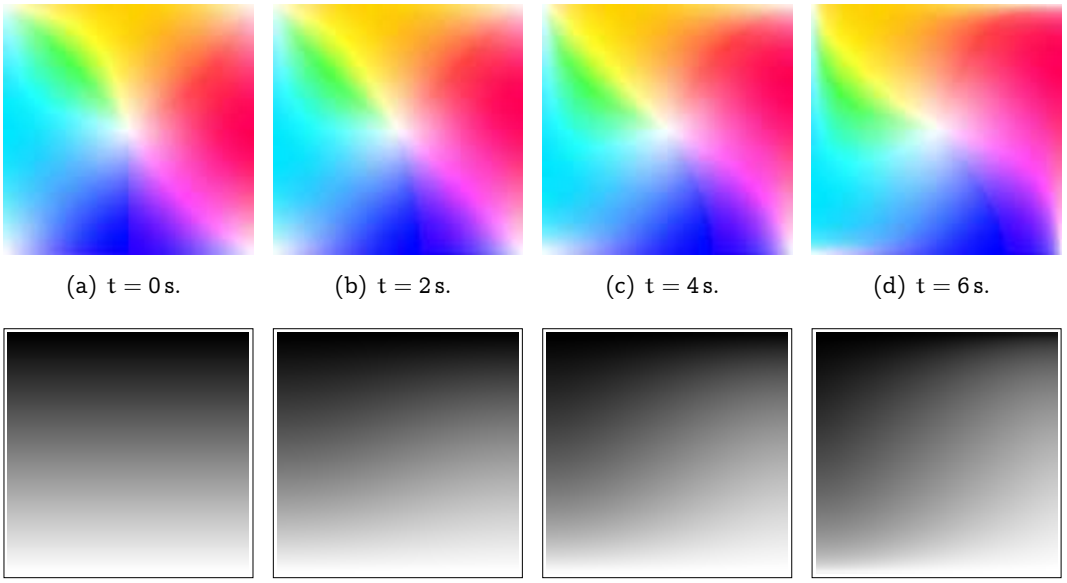
On décrit, dans cette sous-section, l'obtention des bases réduites $\Phi = \{\phi_i(\mathbf{x})\}_{i=1\ldots K}$, pour la représentation de la séquence de mouvement $\mathbf{w}$, et $\Psi = \{\psi_j(\mathbf{x})\}_{j=1\ldots L}$, pour la séquence d'images I. Dans un cadre général, on considère une séquence discrète $E =$

1. http://vision.middlebury.edu/flow/



(a) $\mathbf{w}_0$. (b) I_0.

FIGURE 2.1 – Conditions initiales de la simulation. Gauche : représentation du mouvement en pseudo-couleurs. Droite : image initiale.



(a) $t = 0\,s$. (b) $t = 2\,s$. (c) $t = 4\,s$. (d) $t = 6\,s$.

(e) $t = 0\,s$. (f) $t = 2\,s$. (g) $t = 4\,s$. (h) $t = 6\,s$.

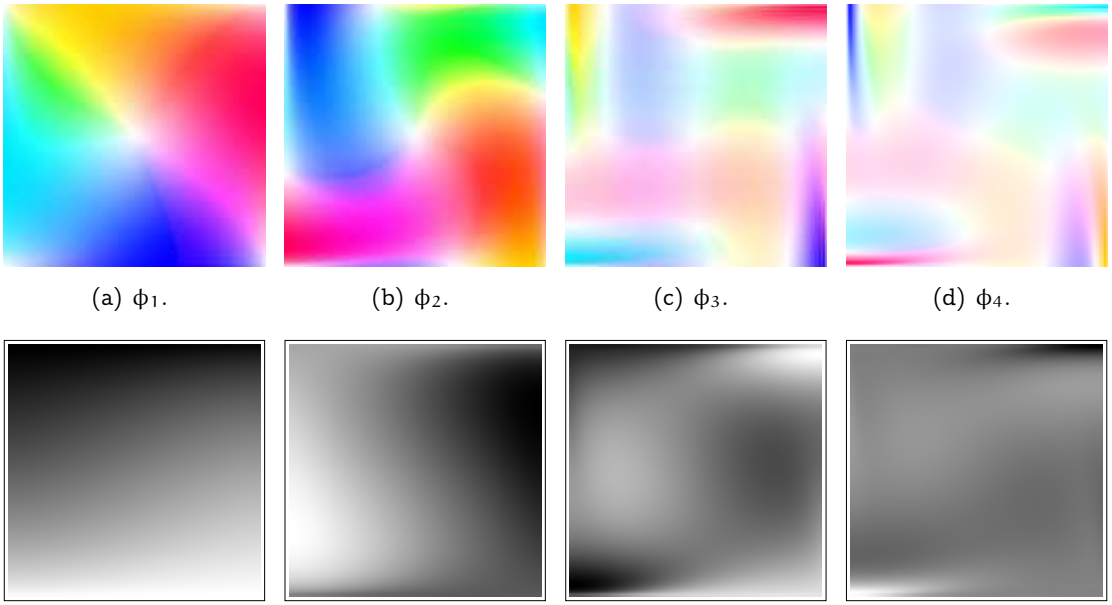
FIGURE 2.2 – Simulation du modèle complet. Haut : représentation en pseudo-couleurs de $\mathbf{w}(\mathbf{x}, t)$. Bas : pseudo-images $I_s(\mathbf{x}, t)$.

$\{E^m\}_{m\in\{1,\dots,M\}}$. E peut être scalaire, dans le cas $E = I$, ou vectorielle dans le cas $E = W$. Les données sont discrètes et représentées sur une grille de pixels. Un élément E^m s'écrit sur la base canonique $\{e_n\}_{n=1\dots N}$ des pixels :

$$\forall m \in \{1,\dots,M\}, \qquad E^m = \sum_{n=1}^{N} E_n^m e_n \tag{2.3.2}$$

avec N le nombre de pixels du domaine. Considérons la matrice $\underline{E}$ de taille $N \times M$, tel que l'élément de la $n^{\text{ième}}$ ligne et $m^{\text{ième}}$ colonne est E_n^m. Soit $G = \underline{E}\underline{E}^T$ la matrice de Gram de taille $N \times N$. Une décomposition orthogonale propre est appliquée à la matrice G. Les vecteurs propres obtenus, associés aux valeurs propres non nulles, forment une base $\Gamma = \{\gamma_m\}_{m=1\dots M}$ de l'espace Vect(E) engendré par E.
Les P premiers vecteurs propres, expliquant 99% de la variance de E, sont choisis pour constituer la base réduite que l'on note également $\Gamma = \{\gamma_p\}_{p=1\dots P}$ afin de simplifier les notations.

Bases obtenues. La décomposition orthogonale propre, appliquée aux séquence W et I, fournit respectivement les bases réduites Φ pour le mouvement et Ψ pour les images. Les bases obtenues sont représentées sur la figure 2.3. En choisissant de limiter à 99% de la variance, les bases sont toutes deux constituées de 4 éléments : $K = L = 4$.



(a) ϕ_1. (b) ϕ_2. (c) ϕ_3. (d) ϕ_4.

(e) ψ_1. (f) ψ_2. (g) ψ_3. (h) ψ_4.

FIGURE 2.3 – Bases Φ (haut) et Ψ (bas), obtenues par décomposition orthogonale propre de $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$.

Afin d'évaluer la qualité de ces bases pour représenter le mouvement et les images, on réalise, dans la sous-section 2.3.3, la projection des séquences $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ sur les bases Φ et Ψ.

2.3.3 Analyse des projections sur les bases Φ et Ψ

Afin de bâtir un modèle réduit $\mathbb{M}_R$, qui soit une bonne approximation du modèle complet $\mathbb{M}$, la première étape consiste à estimer un sous-espace pour le mouvement et les images. C'est ce que nous avons fait dans la sous-section précédente. Une fois ce sous-espace obtenu, il faut s'assurer que l'erreur de projection des séquences est faible, car le mouvement sera estimé dans ce sous-espace de projection.

Soient $\mathbf{w}^K(\mathbf{x},t)$ et $I_s^L(\mathbf{x},t)$ les projections de $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ sur Φ et Ψ :

$$\begin{aligned}\mathbf{w}^K(\mathbf{x},t) &= \sum_{i=1}^{K} \frac{\langle \mathbf{w}(t), \phi_i \rangle}{\langle \phi_i, \phi_i \rangle} \phi_i(\mathbf{x}) \\ I_s^L(\mathbf{x},t) &= \sum_{j=1}^{L} \frac{\langle I_s(t), \psi_j \rangle}{\langle \psi_j, \psi_j \rangle} \psi_j(\mathbf{x})\end{aligned}$$

Les séquences obtenues sont représentées sur les figures 2.4 et 2.5. Visuellement, les séquences de mouvement et d'images sont semblables. Quelques statistiques d'erreur de projection sont fournies dans le tableau 2.1. Les statistiques d'erreurs sont très bonnes : les corrélations sont très proches de 1 et la NRMSE est de 0.08% pour le mouvement et 0.06% pour les images.

Les formules utilisées pour calculer ces statistiques sont les suivantes :

- Corrélation :

$$\text{Corrélation} = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N} (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^{N} (y_i - \bar{y})^2}} \tag{2.3.3}$$

  avec $\bar{x} = \frac{1}{N} \sum_{i=1}^{N} x_i$ et $\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i$,

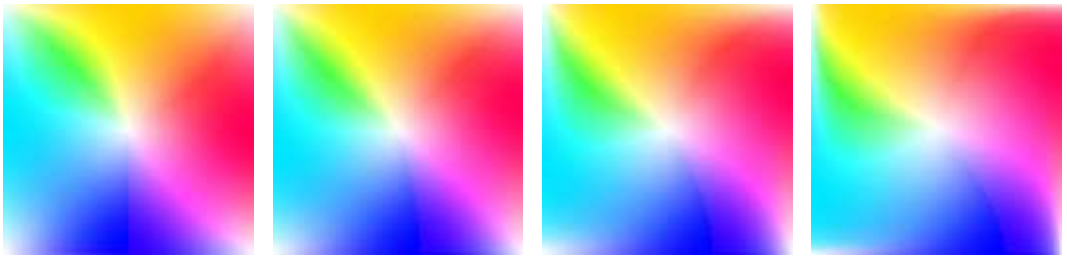
- RMSE (Root-Mean-Square Error) :

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (x_i - y_i)}{N}} \tag{2.3.4}$$

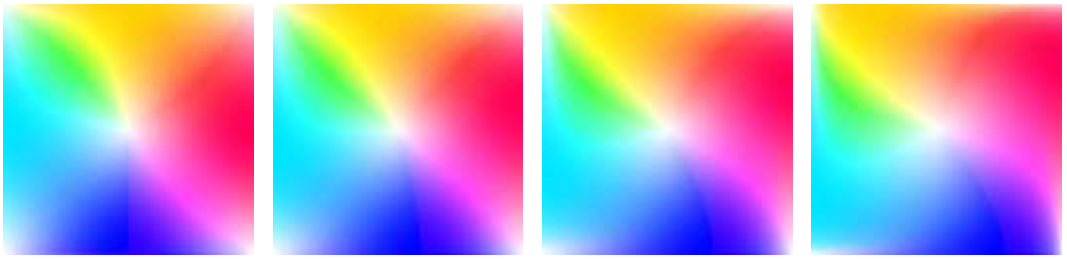
- NRMSE (Normalized Root-Mean-Square Error) :

$$\text{NRMSE} = \frac{\text{RMSE}}{x_{max} - x_{min}}. \tag{2.3.5}$$

Ces résultats montrent que Φ et Ψ permettent de bien approximer $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ sur des sous-espaces de dimension très faible, puisque la dimension des sous-espaces est de 4. Dans la sous-section suivante, on réalise une simulation du modèle réduit $\mathbb{M}_R$ qui est obtenu par projection de Galerkin de $\mathbb{M}$ sur les bases Φ et Ψ, selon le principe expliqué dans la sous-section 2.2.2.

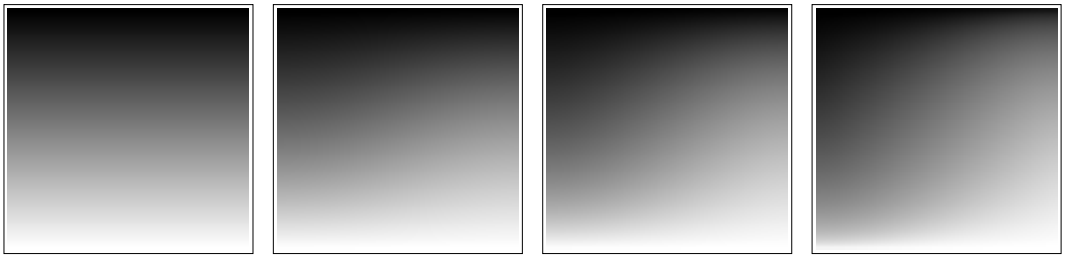


(a) $t = 0\,s$. (b) $t = 2\,s$. (c) $t = 4\,s$. (d) $t = 6\,s$.

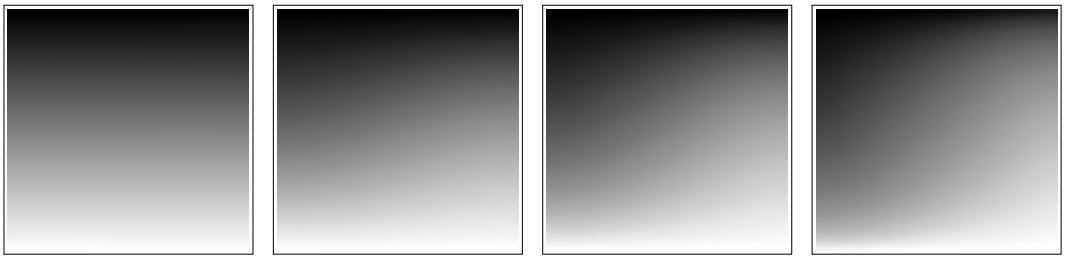


(e) $t = 0\,s$. (f) $t = 2\,s$. (g) $t = 4\,s$. (h) $t = 6\,s$.

FIGURE 2.4 – Haut : séquence $\mathbf{w}(\mathbf{x}, t)$ obtenue par simulation du modèle complet. Bas : séquence $\mathbf{w}^K(\mathbf{x}, t)$ obtenue par projection de $\mathbf{w}(\mathbf{x}, t)$ sur la base Φ.



(a) $t = 0\,s$. (b) $t = 2\,s$. (c) $t = 4\,s$. (d) $t = 6\,s$.



(e) $t = 0\,s$. (f) $t = 2\,s$. (g) $t = 4\,s$. (h) $t = 6\,s$.

FIGURE 2.5 – Haut : séquence $I_s(\mathbf{x}, t)$ obtenue par simulation du modèle complet. Bas : séquence $I_s^L(\mathbf{x}, t)$ obtenue par projection de $I_s(\mathbf{x}, t)$ sur la base Ψ.

Mouvement	valeur min	moyenne	valeur max
$\mathbf{w}$	-5.75	9.47×10^{-3}	5.75
$\mathbf{w}^K$	-5.78	9.435×10^{-3}	5.78

Images	valeur min	moyenne	valeur max
I_s	-49.23	-0.4251	48.46
I_s^L	-49.90	-0.4250	48.72

	$\mathbf{w}$	I_s
RMSE	9.37×10^{-3}	6.27×10^{-2}
NRMSE (%)	8.14×10^{-2}	6.42×10^{-2}
Corrélation	0.99	0.99

TABLE 2.1 – Statistiques d'erreur entre les séquences $\mathbf{w}(\mathbf{x}, t)$ et $I_s(\mathbf{x}, t)$ et leurs projections sur les bases Φ et Ψ.

2.4 Simulation du modèle réduit

Dans cette section, on présente une simulation du modèle réduit $\mathbb{M}_R$, obtenu par projection de Galerkin du modèle complet AIMI sur les bases POD décrites dans la section 2.3. On nomme ce modèle réduit MR-AIMI-POD-POD, afin de signifier que le modèle AIMI est projeté sur deux bases POD pour obtenir le modèle réduit. Les simulations des modèles complet et réduit sont ensuite comparées afin de s'assurer que le modèle réduit approxime convenablement le modèle complet.

2.4.1 Simulation du modèle réduit

Comme décrit dans la section précédente, les bases Φ et Ψ sont chacune constituées de quatre éléments ($K = L = 4$). On considère le système d'évolution du modèle réduit :

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T B(k) a = 0, \quad k = 1 \dots K \\ \dfrac{db_l}{dt}(t) + a^T G(l) b = 0, \quad l = 1 \dots L \\ a(0) = a^0 \\ b(0) = b^0 \end{cases} \tag{2.4.1}$$

avec a^0 et b^0 conditions initiales de la simulation.

Afin de simplifier l'interprétation des résultats, les normes des ϕ_j et ψ_j sont choisies telles que $a^0 = \begin{pmatrix} 1 & \dots & 1 \end{pmatrix}^T$, $b^0 = \begin{pmatrix} 1 & \dots & 1 \end{pmatrix}^T$. Pour cela, on divise ϕ_i et ψ_j par les cœfficients de projection des champs $\mathbf{w}_0(\mathbf{x})$ et $I_0(\mathbf{x})$ (présentés sur la figure 2.1) sur ϕ_i et ψ_j. Les gradients spatiaux intervenant dans l'expression des matrices $B(k)$ et $G(l)$ sont calculés en utilisant le filtre récursif de Canny-Deriche [4, 12] avec la valeur 1.9 pour le paramètre α du noyau $f(x) = Sxe^{-\alpha|x|}$.

La simulation est effectuée sur 700 pas de temps avec un $dt = 0.01\,s$, soit une fenêtre temporelle de 7 secondes.

2.4.2 Résultats

Les courbes de simulation obtenues sont notées a_i^s et b_j^s (l'exposant s pour le mot simulation), les courbes de projection des séquences $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ sont notées $a_i^p(t)$ et $b_j^p(t)$ (l'exposant p pour le mot projection). Ces courbes sont présentées sur les figures 2.6 et 2.7. Les courbes simulées et les courbes de projection sont, à l'exception de b_4, pratiquement identiques. La projection sur Φ et Ψ du mouvement et des images est bien approximée par le modèle réduit. Il reste à vérifier que les champs de mouvement et les images obtenus à partir des cœfficients a_i^s et b_j^s sont également proches des résultats de simulation du modèle complet.

On reconstruit les champs de mouvement $\mathbf{w}^r(\mathbf{x},t)$ et les images $I_s^r(\mathbf{x},t)$ (l'exposant r pour le mot reconstruction), correspondant aux courbes a_i^s et b_j^s, afin de quantifier les performances de l'approximation du modèle complet AIMI par le modèle réduit MR-AIMI-POD-POD. Le résultat obtenu par simulation du modèle complet et la reconstruction de la simulation du réduit sont présentés sur les figures 2.8 et 2.9. Visuellement, les séquences de mouvement et les séquences d'images des modèles complet et réduit sont semblables. Des statistiques d'erreur sont fournies dans le tableau 2.2. Ces statistiques sont très bonnes : les corrélations sont très proches de 1 et la NRMSE est de 0.13% pour le mouvement et 0.82% pour les images.

Mouvement	valeur min	moyenne	valeur max
$\mathbf{w}$	-5.75	9.47×10^{-3}	5.75
$\mathbf{w}^r$	-5.80	9.42×10^{-3}	5.80

Images	valeur min	moyenne	valeur max
I_s	49.23	-0.425	48.46
I_s^r	-53.05	-0.443	52.22

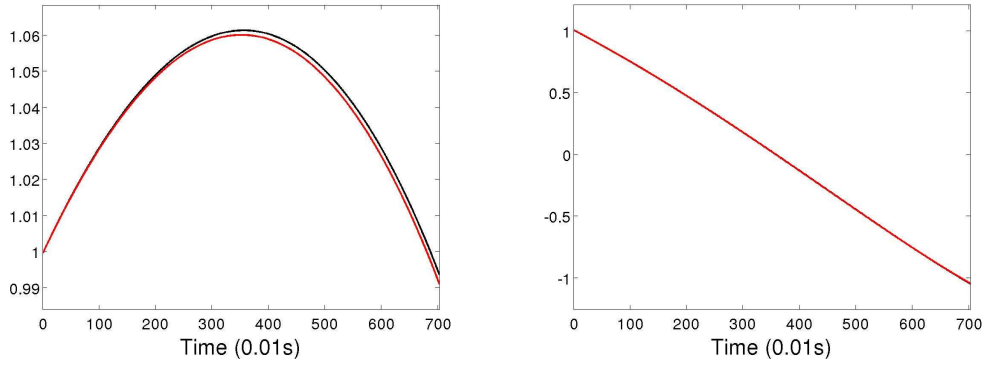
	$\mathbf{w}$	I_s
RMSE	1.46×10^{-2}	0.806
NRMSE (%)	0.13	0.825
Corrélation	0.99	0.99

TABLE 2.2 – Statistiques d'erreur entre les séquences $(\mathbf{w}(\mathbf{x},t), I_s(\mathbf{x},t))$ et $(\mathbf{w}^r(\mathbf{x},t), I_s^r(\mathbf{x},t))$.

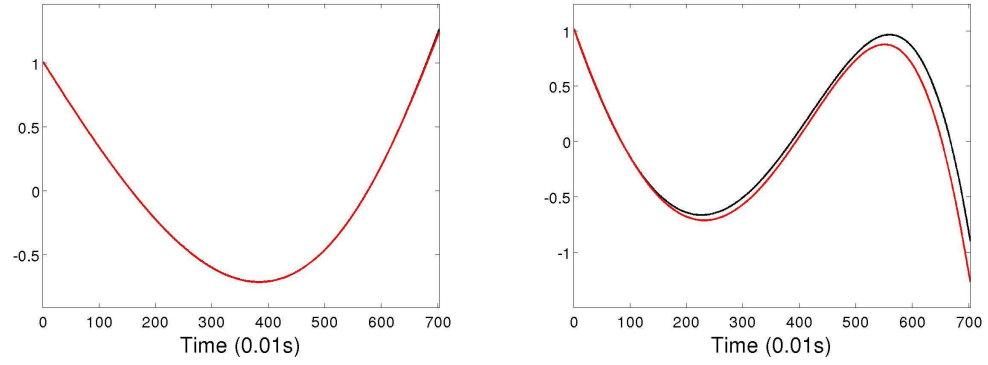
Conclusions. Les statistiques d'erreurs entre les séquences simulées montrent que le modèle réduit MR-AIMI-POD-POD constitue une très bonne approximation du modèle complet. On utilise donc, dans la section 2.5, ce modèle réduit, afin estimer le mouvement au moyen d'une méthode d'assimilation de données.

2.5 Assimilation dans le modèle réduit

On souhaite, à partir d'une séquence discrète d'images $I = \{I(\mathbf{x},t_k)\}_k$, estimer le mouvement $\mathbf{w}(\mathbf{x},t)$. Comme décrit dans le chapitre 1, cette estimation peut se faire par assimilation des images I dans un modèle complet, par exemple le modèle AIMI. Toutefois, afin

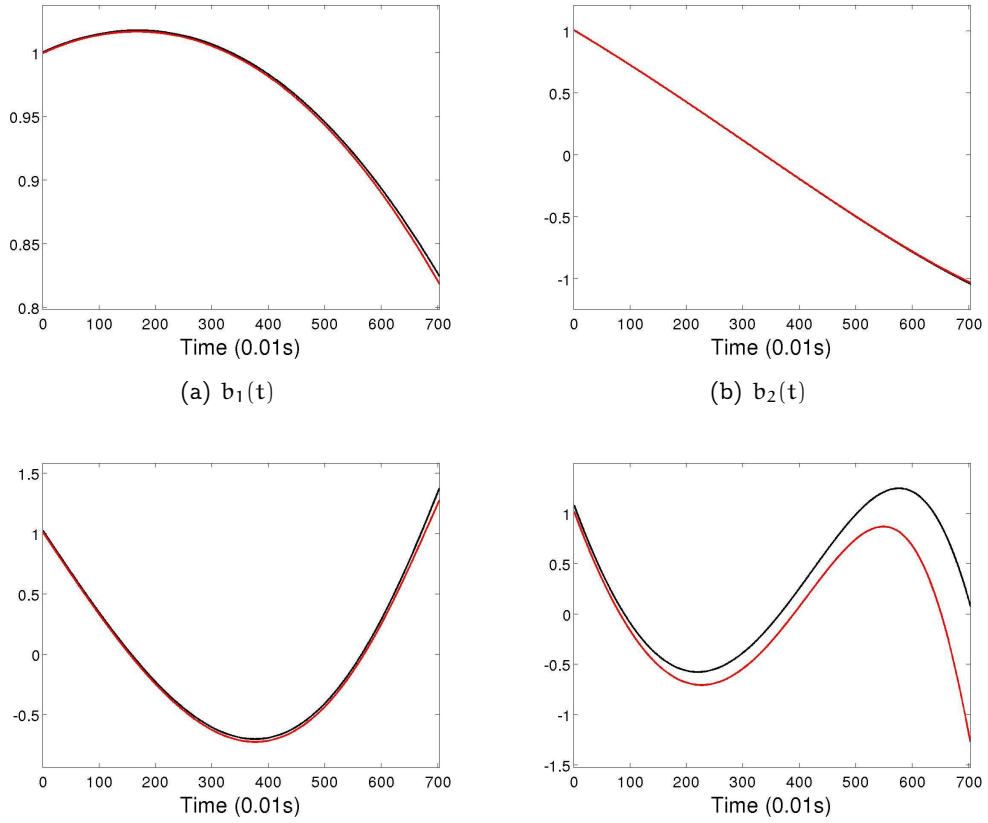


(a) $a_1(t)$ (b) $a_2(t)$



(c) $a_3(t)$ (d) $a_4(t)$

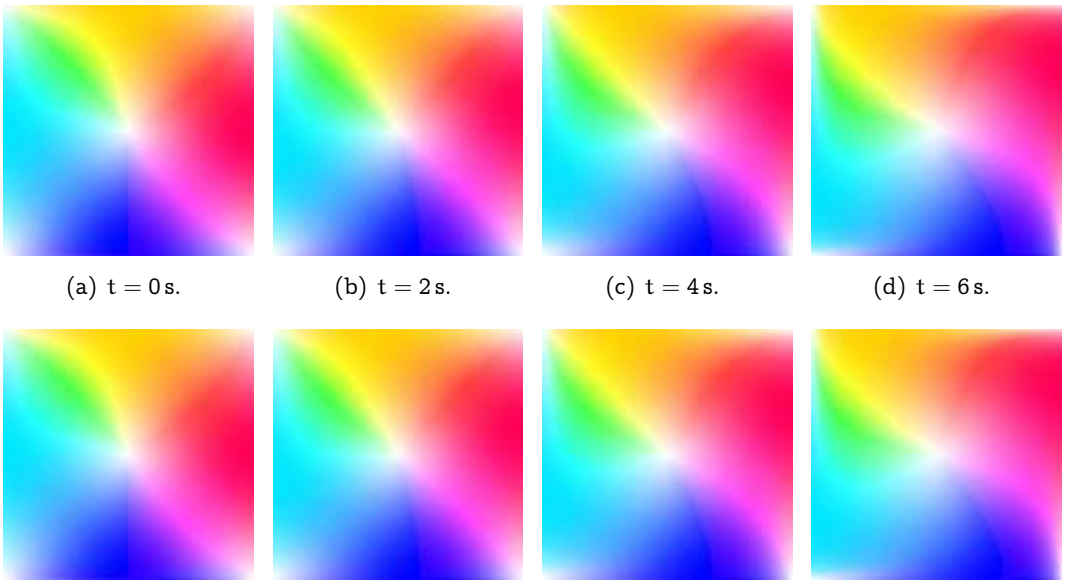
FIGURE 2.6 – Cœfficients de projection du modèle complet $a_i^p(t)$ en rouge et cœfficients simulés par le modèle réduit a_i^s en noir.



(a) $b_1(t)$ (b) $b_2(t)$

(c) $b_3(t)$ (d) $b_4(t)$

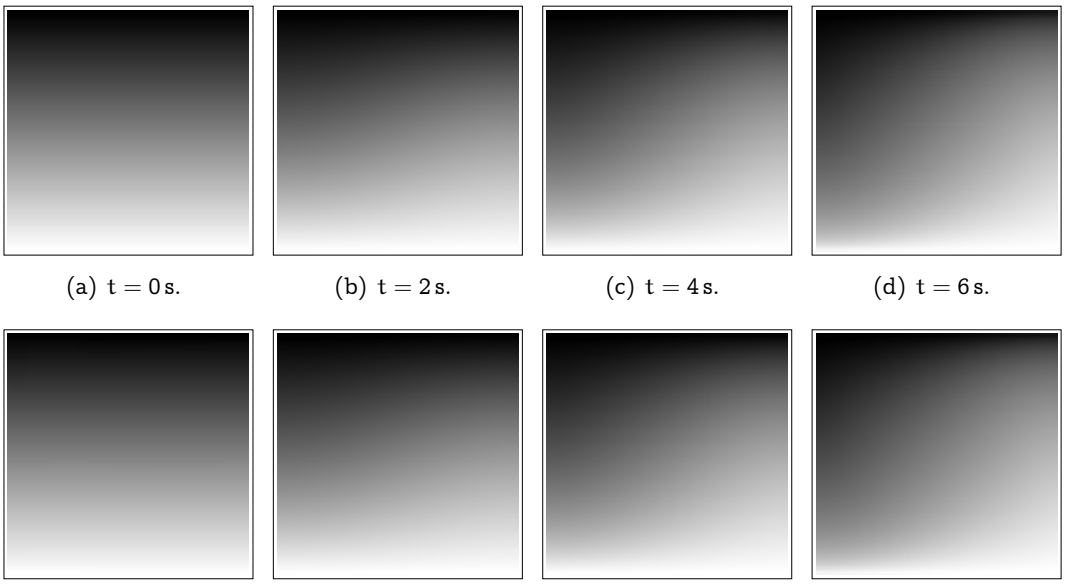
FIGURE 2.7 – Cœfficients de projection du modèle complet $b_i^p(t)$ en rouge et cœfficients simulés par le modèle réduit b_i^s en noir.



(a) $t = 0\,s$. (b) $t = 2\,s$. (c) $t = 4\,s$. (d) $t = 6\,s$.

(e) $t = 0\,s$. (f) $t = 2\,s$. (g) $t = 4\,s$. (h) $t = 6\,s$.

FIGURE 2.8 – Haut : séquence $\mathbf{w}(\mathbf{x}, t)$ obtenue par simulation du modèle complet. Bas : séquence $\mathbf{w}^r(\mathbf{x}, t)$ obtenue à partir de la simulation du modèle réduit.



(a) $t = 0\,s$. (b) $t = 2\,s$. (c) $t = 4\,s$. (d) $t = 6\,s$.

(e) $t = 0\,s$. (f) $t = 2\,s$. (g) $t = 4\,s$. (h) $t = 6\,s$.

FIGURE 2.9 – Haut : séquence $I_s(\mathbf{x}, t)$ obtenue par simulation du modèle complet. Bas : séquence $I_s^r(\mathbf{x}, t)$ obtenue à partir de la simulation du modèle réduit.

de réduire le temps de calcul, on utilise le modèle MR-AIMI-POD-POD pour réaliser l'estimation du mouvement. On assimile donc les cœfficients de projection $b_j(t_k)$ des images de la séquence I dans le modèle réduit MR-AIMI-POD-POD, afin d'estimer les cœfficients $a_i^a(t)$ (l'exposant a pour le mot assimilation) du mouvement. Ces cœfficients permettent alors de reconstruire le mouvement et d'obtenir l'estimation $\mathbf{w}^a(\mathbf{x}, t)$.

Le système d'assimilation, ainsi que l'algorithme conçu, sont présentés dans la sous-section 2.5.1 et une expérience jumelle est décrite dans la sous-section 2.5.2, afin d'évaluer la qualité de la méthode.

2.5.1 Système d'assimilation

Le vecteur d'état réduit $\mathbf{X}_R(t) = \begin{pmatrix} a(t)^T & b(t)^T \end{pmatrix}^T$ est défini sur $[0, T]$, et vérifie l'équation d'évolution :

$$\frac{d\mathbf{X}_R}{dt}(t) + \mathbb{M}_R(\mathbf{X}_R)(t) = 0. \tag{2.5.1}$$

On considère des observations, ou plutôt des cœfficients observés, $b^{obs}(t_k) = \begin{pmatrix} b_1(t_k) & \dots & b_L(t_k) \end{pmatrix}$ liés à $\mathbf{X}_R$ par l'équation d'observation :

$$b^{obs}(t_k) = \mathbb{H}\mathbf{X}_R(t_k) + \mathcal{E}_O(t_k) \tag{2.5.2}$$

$\mathbb{H}$ est la matrice de la projection $\begin{pmatrix} a(t)^T & b(t)^T \end{pmatrix}^T \to b(t)$.
$\mathcal{E}_O(t)$ représente l'erreur liée à l'observation.

On considère une estimation $\mathbf{X}_{R,b}$ du vecteur d'état à la date $t = 0$, appelée ébauche :

$$\mathbf{X}_R(0) = \mathbf{X}_{R,b} + \mathcal{E}_b \tag{2.5.3}$$

$\mathcal{E}_b$ est l'erreur associée à cette ébauche.

On fait l'hypothèse que les erreurs $\mathcal{E}_O(t)$ et $\mathcal{E}_b$ sont gaussiennes de moyenne nulle et que $\mathcal{E}_O(t)$ est décorrélée en temps. Elles sont caractérisées par leurs matrices de covariance respectives $R(t)$ et B.

On cherche à estimer le vecteur d'état $\mathbf{X}_R$ vérifiant le système d'assimilation :

$$\begin{aligned} \frac{d\mathbf{X}_R}{dt}(t) + \mathbb{M}_R(\mathbf{X})(t) &= 0 && (2.5.4)\\ \mathbf{X}_R(0) &= \mathbf{X}_{R,b} + \mathcal{E}_b && (2.5.5)\\ b^{obs}(t_k) &= \mathbb{H}\mathbf{X}_R(t_k) + \mathcal{E}_o(t_k), && (2.5.6) \end{aligned}$$

Comme décrit dans Bérèziat et al. [2] pour le cas général, et dans l'annexe A pour le cas particulier d'un modèle réduit, le système est résolu en minimisant la fonctionnelle J :

$$\begin{aligned} J(\mathbf{X}_R(0)) = & \frac{1}{2}\left(\mathbf{X}_R(0) - \mathbf{X}_{R,b}\right)^T B^{-1}\left(\mathbf{X}_R(0) - \mathbf{X}_{R,b}\right) \\ & + \frac{1}{2}\int_{[0,T]} \left(b^{obs}(t) - \mathbb{H}\mathbf{X}_R(t)\right)^T R^{-1}(t)\left(b^{obs}(t) - \mathbb{H}\mathbf{X}_R(t)\right) dt \end{aligned}$$

La minimisation de la fonctionnelle J permet de calculer $\mathbf{X}_R(0)$ puis $\mathbf{X}_R(t)$ par par intégration de l'équation 2.5.4 sur l'intervalle temporel $[0, T]$.

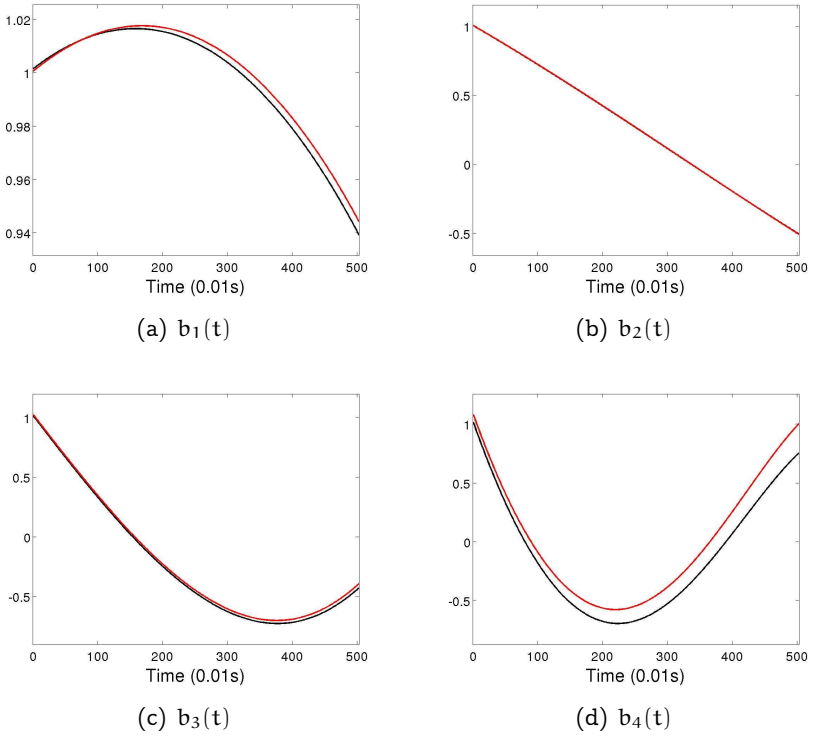
2.5.2 Expérience jumelle d'assimilation

Une simulation $\mathbf{X}(\mathbf{x}, t) = \begin{pmatrix} \mathbf{w}(\mathbf{x}, t) \\ I_s(\mathbf{x}, t) \end{pmatrix}$ du modèle complet AIMI, à partir de conditions initiales $\mathbf{X}(0)$ sur une fenêtre $[0, T]$, est présentée dans la sous-section 2.3.1. Des dates d'observation t_z sont choisies pour former une séquence discrète de 11 observations $I = \{I_s^z\}_z = \{I_s(\mathbf{x}, t_z)\}_z : t_z = 0.01\,s, 0.51\,s, \ldots, 5.01\,s$. Les cœfficients de projection $\{b^z\}_{z=1\ldots Z}$ des images I_s^z sont assimilés dans le modèle réduit, pour estimer les cœfficients $a_i^a(t)$ et les comparer à la vérité terrain.

La variable de contrôle est la condition initiale $\mathbf{X}_R(0) = \begin{pmatrix} a(0) \\ b(0) \end{pmatrix}$. Lors de l'assimilation, la valeur de l'ébauche est 0 pour les cœfficients a_i et on utilise les cœfficients de projection de la première image I_s^1 comme valeur de l'ébauche sur b. Les cœfficients $a_i(0)$ et $b_j(0)$ sont, comme décrit dans la sous-section 2.4.1, égaux à 1. On considère donc que l'ordre de grandeur des cœfficients $a_i(t)$ et $b_j(t)$ est de 1 et on choisit des écarts-types de 10^{-1} pour les erreurs d'ébauche et d'observation. Les matrices de covariance $R(t)$ et B sont diagonales avec la valeur 10^{-2} sur la diagonale. La fenêtre d'assimilation comporte 502 pas de temps avec un $dt = 0.01\,s$, soit une fenêtre temporelle de 5.02 secondes.

Résultats. On obtient une séquence $\begin{pmatrix} a^a(t) \\ b^a(t) \end{pmatrix}$ (l'exposant a signifie que ce sont des résultats d'assimilation), à partir de laquelle on reconstruit le champ de mouvement $\mathbf{w}^a(\mathbf{x}, t)$

et l'image $I_s^a(x,t)$, pour $t \in [0\,s, 5.02\,s]$. Les courbes obtenues sont présentées sur les figures 2.10 et 2.11. L'évolution temporelle des cœfficients b_j^a est très proche de l'évolution temporelle des cœfficients de projection de la simulation du modèle complet. Les courbes a_1^a, a_2^a et a_3^a obtenues sont elles aussi proches de la vérité terrain. Seules la courbe a_4^a n'est pas correctement estimée (biais constant de l'ordre de 1). Afin de quantifier l'écart entre les résultats d'assimilation et la vérité terrain, les champs obtenus à partir des courbes sont présentés sur les figures 2.12 et 2.13 et quelques statistiques d'erreur sont fournies sur le tableau 2.3. Les images I_s^a obtenues sont visuellement très proches des observations I_s^z, la corrélation entre les deux séquences est de 99% et la NRMSE inférieure à 0.1%. On observe, sur la figure 2.12 que les séquences de mouvement sont également semblables. La NRMSE est inférieure à 1% sur l'ensemble de la séquence. La corrélation est égale à 0.99.

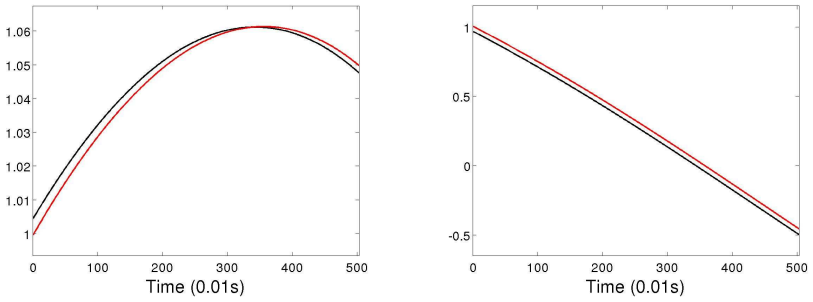


(a) $b_1(t)$ (b) $b_2(t)$

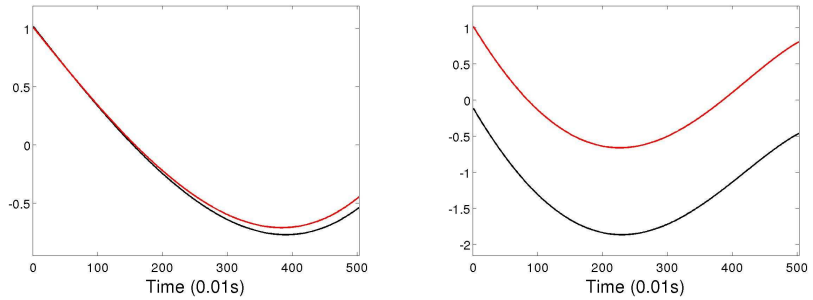
(c) $b_3(t)$ (d) $b_4(t)$

FIGURE 2.10 – Cœfficients $b_i^p(t)$ de projection du modèle complet en rouge et cœfficients b_i^a obtenus par assimilation dans le modèle réduit en noir.

Conclusions. Les résultats obtenus dans cette expérience jumelle montrent qu'il est possible d'utiliser un modèle réduit POD pour estimer correctement le mouvement par assimilation de données. L'objectif de réduction du temps de calcul et de mémoire utilisée est parfaitement atteint : le vecteur d'état de ce modèle complet comporte est de 12288 composantes et celui du modèle réduit n'en comporte que 8. Le temps de calcul pour une itération de minimisation est de l'ordre de la seconde pour le modèle complet et de la milliseconde pour le modèle réduit.

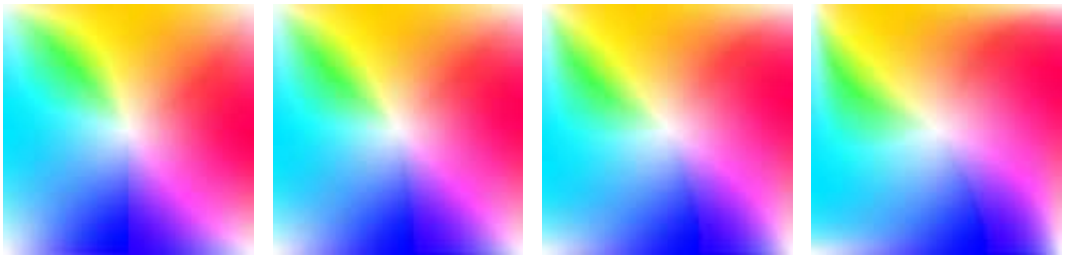


(a) $a_1(t)$ (b) $a_2(t)$



(c) $a_3(t)$ (d) $a_4(t)$

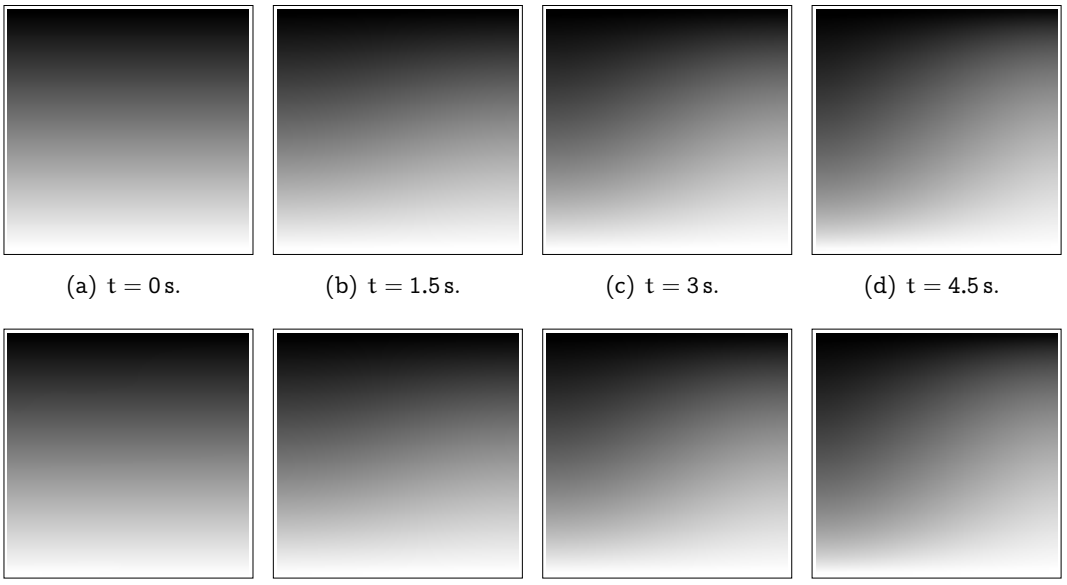
FIGURE 2.11 – Cœfficients $a_i^p(t)$ de projection du modèle complet en rouge et cœfficients a_i^a obtenus par assimilation dans le modèle réduit en noir.



(a) $t = 0\,s$. (b) $t = 1.5\,s$. (c) $t = 3\,s$. (d) $t = 4.5\,s$.

(e) $t = 0\,s$. (f) $t = 1.5\,s$. (g) $t = 3\,s$. (h) $t = 4.5\,s$.

FIGURE 2.12 – Haut : séquence $\mathbf{w}(\mathbf{x}, t)$ obtenue par simulation du modèle complet. Bas : séquence $\mathbf{w}^a(\mathbf{x}, t)$ obtenue par assimilation dans le modèle réduit.



(a) $t = 0\,s$. (b) $t = 1.5\,s$. (c) $t = 3\,s$. (d) $t = 4.5\,s$.

(e) $t = 0\,s$. (f) $t = 1.5\,s$. (g) $t = 3\,s$. (h) $t = 4.5\,s$.

FIGURE 2.13 – Haut : séquence $I_s(\mathbf{x}, t)$ obtenue par simulation du modèle complet. Bas : séquence $I_s^a(\mathbf{x}, t)$ obtenue par assimilation dans le modèle réduit.

Mouvement	valeur min	moyenne	valeur max
$\mathbf{w}$	-5.83	4.9e-5	5.75
$\mathbf{w}^a$	-5.83	5.4e-5	5.75

Images	valeur min	moyenne	valeur max
I_s	-49.22	1.71e-2	49.22
I_s^a	-49.70	1.72e-2	49.82

	$\mathbf{w}$	I_s
RMSE	6.95e-2	6.73e-2
NRMSE (%)	0.60	6.83e-2
Corrélation	0.99	0.99

TABLE 2.3 – Statistiques d'erreur entre les séquences $(\mathbf{w}(\mathbf{x}, t), I_s(\mathbf{x}, t))$ et $(\mathbf{w}^a(\mathbf{x}, t), I_s^a(\mathbf{x}, t))$.

2.6 Conclusions

Dans ce chapitre, nous avons conçu et utilisé une méthode d'assimilation de données dans un modèle dynamique d'ordre réduit. Cette approche permet d'estimer des champs de vitesse à partir d'une séquence discrète d'images. Une expérience jumelle, réalisée sur des données synthétiques, démontre l'intérêt de la méthode. Les champs de vitesse sont correctement estimés à partir de vecteurs d'état à 8 composantes.

L'obtention d'un sous-espace POD pour le mouvement nécessite, comme nous l'avons déjà évoqué, l'estimation préalable d'un champ de vitesse initial $\mathbf{w}_0$. Ce champ est intégré en temps, et la décomposition orthogonale propre est appliquée à la séquence obtenue. La qualité de la base POD pour le mouvement repose exclusivement sur la qualité d'estimation de $\mathbf{w}_0$. En effet, dans l'expérience jumelle présentée dans la section 2.5, le champs $\mathbf{w}_0$ est parfaitement connu et permet d'obtenir Φ. Dans un cas d'une application plus réaliste, un prétraitement de la séquence d'images I est indispensable pour calculer une estimation $\tilde{\mathbf{w}_0}$ de $\mathbf{w}_0$. Cette estimation est nécessairement entachée d'une incertitude.
La prise en compte de cette incertitude semble donc fondamentale pour s'assurer de la qualité de la base Φ. Une manière de prendre en compte cette incertitude serait de bruiter N fois $\tilde{\mathbf{w}_0}$, d'intégrer temporellement chaqun de ces champs et de réaliser la décomposition orthogonale propre sur l'ensemble des séquences obtenues. Cela permettrait d'estimer une base POD robuste au bruit. La prise en compte de cette incertitude pose un certain nombre de questions parmi lesquelles :

- Quelle valeur choisir pour N ?
- Quel type de bruitage choisir selon le cadre applicatif ?
- Faut-il spatialiser les incertitudes ?

Quelques tests préliminaires de POD robuste ont étés effectués et montrent que l'on peut correctement estimer la base mouvement avec des incertitudes sur $\tilde{\mathbf{w}_0}$. Un champ $\tilde{\mathbf{w}_0}$ a été obtenu à partir de $\mathbf{w}_0$ en ajoutant un bruit gaussien dont l'écart-type était égal à 10% de l'écart-type de $\mathbf{w}_0$. Les 4 éléments de la base Φ ont été alors correctement estimés par POD robuste à partir de $\tilde{\mathbf{w}_0}$.

Chapitre 3

Réduction du modèle de mouvement à divergence nulle

3.1 Introduction

Ce chapitre est consacré à la description d'un modèle réduit d'advection sous l'hypothèse d'un mouvement à divergence nulle. Dans le chapitre 2, nous avons présenté un modèle d'advection du mouvement et des images (AIMI), ainsi qu'une méthode de réduction permettant une estimation rapide du mouvement par assimilation de données. Le modèle AIMI n'est cependant pas adapté au traitement de tous les types de séquences d'images.
Dans certains cas, des hypothèses supplémentaires sont disponibles. Comme décrit dans [20], la circulation océanique de surface sur la mer Noire est essentiellement horizontale : les courants verticaux sont très faibles. Dans ce cadre, l'hypothèse d'un mouvement 2D à divergence nulle est approximativement justifiée. On se propose donc, dans ce chapitre, d'utiliser un modèle d'évolution caractérisant un mouvement à divergence nulle, afin estimer le mouvement par assimilation de données, et de définir un modèle réduit pour réduire le temps de calcul et la mémoire nécessaire.
La section 3.2 est consacrée à la présentation du modèle complet, nommé AIMIDF. Ce modèle, tout comme AIMI dans le chapitre précédent, suppose l'advection de la vitesse et des images par la vitesse. Mais il contraint l'évolution à se faire en satisfaisant l'hypothèse d'une divergence nulle : DF (Divergence Free). On présente, dans la section 3.3, la réduction du modèle. Les sections 3.4 à 3.7 sont ensuite consacrées aux expériences numériques.

3.2 Modèle complet d'advection image à divergence nulle

3.2.1 État du modèle

Cette sous-section concerne un modèle 2D d'advection, dont l'état est défini sur le domaine spatio-temporel $A = \Omega \times [0, T]$.
Les variables d'état sont les composantes u et v du vecteur vitesse $\mathbf{w}$ et, comme dans le chapitre 1 pour le modèle AIMI, une pseudo-image $\mathbf{I}_s$. Le vecteur $\mathbf{X}(\mathbf{x}, t) = (u(\mathbf{x}, t), v(\mathbf{x}, t), I_s(\mathbf{x}, t))^T$ est à valeur dans $\mathbb{R}^3$.
On suppose une divergence nulle du mouvement :

$$\begin{aligned} div(\mathbf{w}) &= 0 \\ \Leftrightarrow \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} &= 0 \end{aligned}$$

3.2.2 Dynamique du modèle

Les équations d'évolution des composantes u et v du mouvement s'écrivent :

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = 0 \tag{3.2.1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = 0 \tag{3.2.2}$$

On dérive l'équation 3.2.1 par rapport à y et l'équation 3.2.2 par rapport à x :

$$\frac{\partial v}{\partial t\partial x} + \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + u\frac{\partial^2 v}{\partial x^2} + \frac{\partial v}{\partial x}\frac{\partial v}{\partial y} + v\frac{\partial^2 v}{\partial x\partial y} = 0 \tag{3.2.3}$$

$$\frac{\partial u}{\partial t\partial y} + \frac{\partial u}{\partial y}\frac{\partial u}{\partial x} + u\frac{\partial^2 u}{\partial x\partial y} + \frac{\partial u}{\partial y}\frac{\partial v}{\partial y} + v\frac{\partial^2 u}{\partial y^2} = 0 \tag{3.2.4}$$

On fait la différence des équations 3.2.3 et 3.2.4 :

$$\begin{aligned} &\frac{\partial}{\partial t}\left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right) + u\frac{\partial}{\partial x}\left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right) + v\frac{\partial}{\partial y}\left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right) \\ &\qquad + \frac{\partial u}{\partial x}\left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right) + \frac{\partial v}{\partial y}\left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right) = 0 \end{aligned} \tag{3.2.5}$$

Soit la vorticité $\xi = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$. On réécrit l'équation 3.2.5 sous la forme :

$$\frac{\partial \xi}{\partial t} + u\frac{\partial \xi}{\partial x} + v\frac{\partial \xi}{\partial y} + \xi\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = 0 \tag{3.2.6}$$

La divergence est nulle ($\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$), donc le système 3.2.6 s'écrit :

$$\begin{aligned} \frac{\partial \xi}{\partial t} + u\frac{\partial \xi}{\partial x} + v\frac{\partial \xi}{\partial y} &= 0 \\ \frac{\partial \xi}{\partial t} + \mathbf{w}\nabla\xi &= 0 \end{aligned}$$

Cette équation peut aussi être écrite sous la forme :

$$\frac{\partial \xi}{\partial t} + div(\xi\mathbf{w}) = 0$$

On considère alors un nouveau modèle, dont l'état s'écrit : $\mathbf{X}(\mathbf{x}, t) = \begin{pmatrix} \xi(\mathbf{x}, t) & I_s(\mathbf{x}, t) \end{pmatrix}^T$, et le système d'évolution :

$$\begin{cases} \dfrac{\partial \xi}{\partial t}(\mathbf{x},t) + \mathrm{div}\left(\mathbf{w}\xi\right)(\mathbf{x},t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x},t) + \mathrm{div}\left(\mathbf{w}I_s\right)(\mathbf{x},t) = 0 \\ \xi(\mathbf{x},0) = \xi_0(\mathbf{x}) \\ I_s(\mathbf{x},0) = I_0(\mathbf{x}) \end{cases} \tag{3.2.7}$$

Pour discrétiser spatialement ces deux équations, on utilise un centré décrit dans l'annexe C. Les conditions aux bords utilisés pour la simulation de ce système sont également décrites dans l'annexe C. Pour intégrer ξ et I_s, il est nécessaire, comme le montre le système 3.2.7, de calculer $\mathbf{w}$ à chaque pas de temps. $\mathbf{w}$ est calculé à partir de ξ selon la méthode présentée dans la sous-section 3.2.3.

3.2.3 Calcul de w en fonction de ξ

Si le mouvement est à divergence nulle, le calcul de $\mathbf{w}$ à partir de sa vorticité ξ peut se faire dans l'espace de Fourier [34], ou par une méthode algébrique [20]. C'est cette seconde méthode que l'on décrit dans cette section. Ce calcul passe par le calcul d'une variable intermédiaire. Soit φ une fonction de courant telle que $\mathbf{w} = \begin{pmatrix} \dfrac{\partial \varphi}{\partial y} & -\dfrac{\partial \varphi}{\partial x} \end{pmatrix}^T = \nabla^{\perp}\varphi$.
φ vérifie alors l'équation de Poisson : $-\Delta\varphi = \xi$, avec Δ l'opérateur Laplacien :
$\Delta = \left(\dfrac{\partial^2}{\partial x^2} + \dfrac{\partial^2}{\partial y^2}\right)$. On peut alors obtenir le mouvement à partir de la vorticité selon :

$$\mathbf{w}\left(\xi\right) = -\nabla^T \Delta^{-1}\left(\xi\right) \tag{3.2.8}$$

3.2.4 Calcul de φ

Il s'agit, dans cette sous-section, de déterminer les valeurs propres et vecteurs propres de l'opérateur $-\Delta$. Ce calcul est explicité en continu puis en discret. Afin de clarifier l'exposé de la méthode, on commence par effectuer les calculs dans le cas simplifié d'un domaine spatial à une dimension.

Cas 1D en continu. On cherche les valeurs propres λ de $-\Delta = -\dfrac{\partial^2}{\partial x^2}$. Soit :

$$U_K = \sin(\pi K x),$$

alors :

$$\begin{aligned} U_{Kx} &= -\pi K \cos(\pi K x) \\ U_{Kxx} &= -\pi^2 K^2 \sin(\pi K x) \\ -U_{Kxx} &= \pi^2 K^2 \sin(\pi K x) \end{aligned}$$

Donc $U_K = \sin(\pi K x)$ est un vecteur propre de $-\Delta$ associé à la valeur propre $\lambda_K = \pi^2 K^2$.

Cas 2D en continu. On cherche les valeurs propres λ de $-\Delta = -\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2}$. Soit :

$$U_{KL} = \sin(\pi K x)\sin(\pi L y),$$

alors :

$$\begin{aligned}
U_{KLx} &= -\pi K \cos(\pi K x)\sin(\pi L y) \\
U_{KLxx} &= -\pi^2 K^2 \sin(\pi K x)\sin(\pi L y) \\
U_{KLy} &= -\pi L \sin(\pi K x)\cos(\pi L y) \\
U_{KLyy} &= -\pi^2 L^2 \sin(\pi K x)\sin(\pi L y) \\
\Rightarrow -U_{KLxx} - U_{KLyy} &= \pi^2 K^2 \sin(\pi K x)\sin(\pi L y) + \pi^2 L^2 \sin(\pi K x)\sin(\pi L y) \\
&= \left[\pi^2 K^2 + \pi^2 L^2\right] U_{KL}
\end{aligned}$$

Donc $U_{KL} = \sin(\pi K x)\sin(\pi L y)$ est un vecteur propre de $-\Delta$ associé à la valeur propre $\lambda_{KL} = \left[\pi^2 K^2 + \pi^2 L^2\right]$.

Résolution du problème discret en 1D.
Soit un vecteur F de taille N_x représentant les pixels d'une image 1D. On ajoute, aux deux extrémités, un pixel auquel on attribue la valeur 0. La taille du vecteur étendu est $N_x 2b = N_x + 2$. On définit h_x la taille normalisée d'un pixel : $h_x = \frac{1}{N_x + 1} = \frac{1}{N_x 2b - 1}$. Soit dx le pas de discrétisation spatial. La figure 3.1 illustre ces explications.

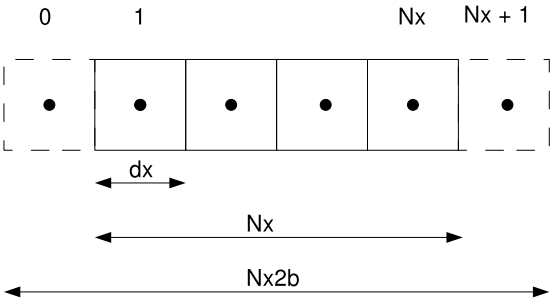


FIGURE 3.1 – Illustration des notations N_x, $N_x 2b$, h_x, dx.

On souhaite discrétiser l'équation $-\frac{\partial^2 \varphi}{\partial x^2} = \xi$ afin de calculer φ en fonction de ξ. L'opérateur $-\frac{\partial^2}{\partial x^2}$ est approximé par $\boxed{\frac{-1}{d_x^2} \mid \frac{2}{d_x^2} \mid \frac{-1}{d_x^2}}$.
Les valeurs de la vorticité ξ sont représentées par le vecteur F et celles de φ par le vecteur U. Connaissant F, on cherche donc la valeur de U. On peut écrire :

$$\frac{1}{dx^2} \times K \times U = F \tag{3.2.9}$$

avec

$$K = \begin{pmatrix} 2 & -1 & 0 & ... & ... & 0 \\ -1 & 2 & -1 & ... & ... & 0 \\ 0 & -1 & 2 & -1 & ... & 0 \\ ... & & & & & \\ ... & & & & & \\ ... & & & & & 0 \\ ... & & & -1 & 2 & -1 \\ ... & & & 0 & -1 & 2 \end{pmatrix}$$

A la $k^{ième}$ ligne de la matrice K on a :

- -1 à la colonne $k-1$
- 2 à la colonne k
- -1 à la colonne $k+1$

On cherche les valeurs propres λ_i et les vecteurs propres ϕ_i de K ($i = 1 \dots N_x$). Soit :

$$\phi_i = \begin{pmatrix} sin(\pi i h_x) \\ sin(2\pi i h_x) \\ ... \\ sin(N_x \pi i h_x) \end{pmatrix} \tag{3.2.10}$$

Montrons que les ϕ_i ainsi définis sont des vecteurs propres de K. On a :

$$K\phi_i = \begin{pmatrix} ... \\ ... \\ -sin[(k-1)\pi i h_x] + 2sin(k\pi i h_x) - sin[(k+1)\pi i h_x] \\ ... \end{pmatrix} \tag{3.2.11}$$

Or :

$$\begin{aligned} sin(a+b) &= sin(a)cos(b) + sin(b)cos(a) \\ sin(a-b) &= sin(a)cos(b) - sin(b)cos(a) \\ sin(a+b) + sin(a-b) &= 2sin(a)cos(b) \end{aligned}$$

donc :

$$-sin[(k-1)\pi i h_x] - sin[(k+1)\pi i h_x] = -2sin(k\pi i h_x)cos(\pi i h_x) \tag{3.2.12}$$

et donc :

$$K\phi_i = \begin{pmatrix} ... \\ ... \\ 2[1 - cos(\pi i h_x)]sin(k\pi i h_x) \\ ... \end{pmatrix}$$

On a donc $K\phi_i = \lambda_i \phi_i$ avec $\lambda_i = 2[1 - cos(\pi i h_x)]$.

On connaît les valeurs propres λ_i et les vecteurs propres ϕ_i de K. Soit $\phi_i^{'}$ la normalisation

de ϕ_i ($\phi_i' = \frac{1}{\langle \phi_i, \phi_i \rangle}\phi_i$). Pour résoudre l'équation 3.2.9, on décompose F dans la base des ϕ_i' :

$$F = \sum_i \left\langle F, \phi_i' \right\rangle \phi_i' = \sum_i a_i \phi_i' \tag{3.2.13}$$

On rappelle que l'on cherche à définir U tel que $\frac{1}{dx^2}KU = F$. Il suffit en fait d'écrire :

$$U = dx^2 \sum_i \frac{a_i}{\lambda_i}\phi_i' \tag{3.2.14}$$

En effet, on vérifie :

$$\begin{aligned} \frac{1}{dx^2}KU &= \sum_i \frac{a_i}{\lambda_i} K\phi_i' \\ &= \sum_i \frac{a_i}{\lambda_i}\lambda_i\phi_i' \\ &= \sum_i a_i\phi_i' \\ &= F \end{aligned}$$

Donc connaissant les ϕ_i' et λ_i, on peut calculer U tel que $\frac{1}{dx^2}KU = F$:

$$U = dx^2 \sum_i \frac{\left\langle F, \phi_i' \right\rangle}{\lambda_i}\phi_i' \tag{3.2.15}$$

Une fois le calcul de U effectué, on obtient $\mathbf{w}$ par l'équation : $\mathbf{w} = \nabla^T U$.

Résolution du problème discret en 2D.
Soit une image G de taille $N_x \times N_y$ représentant les pixels d'une image 2D. On ajoute un pixel de valeur nulle sur tout le bord de l'image. La taille de l'image étendue est donc $N_x2b \times N_y2b$ avec $N_x2b = N_x + 2$ et $N_y2b = N_y + 2$. On définit h_x et h_y les tailles normalisées de l'intervalle horizontal et vertical comme : $h_x = \frac{1}{N_x + 1} = \frac{1}{N_x2b - 1}$ et $h_y = \frac{1}{N_y + 1} = \frac{1}{N_y2b - 1}$. Soient dx et dy les pas de discrétisation dans les directions x et y. La figure 3.2 illustre ces explications.

On souhaite discrétiser l'équation $-\Delta\varphi = \xi$.
On approxime $-\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2}$ par :

0	$\frac{-1}{d_y^2}$	0
$\frac{-1}{d_x^2}$	$\frac{2}{d_x^2} + \frac{2}{d_y^2}$	$\frac{-1}{d_x^2}$
0	$\frac{-1}{d_y^2}$	0

On suppose $dx = dy$, puisque c'est la cas de toutes les images réelles, que nous avons

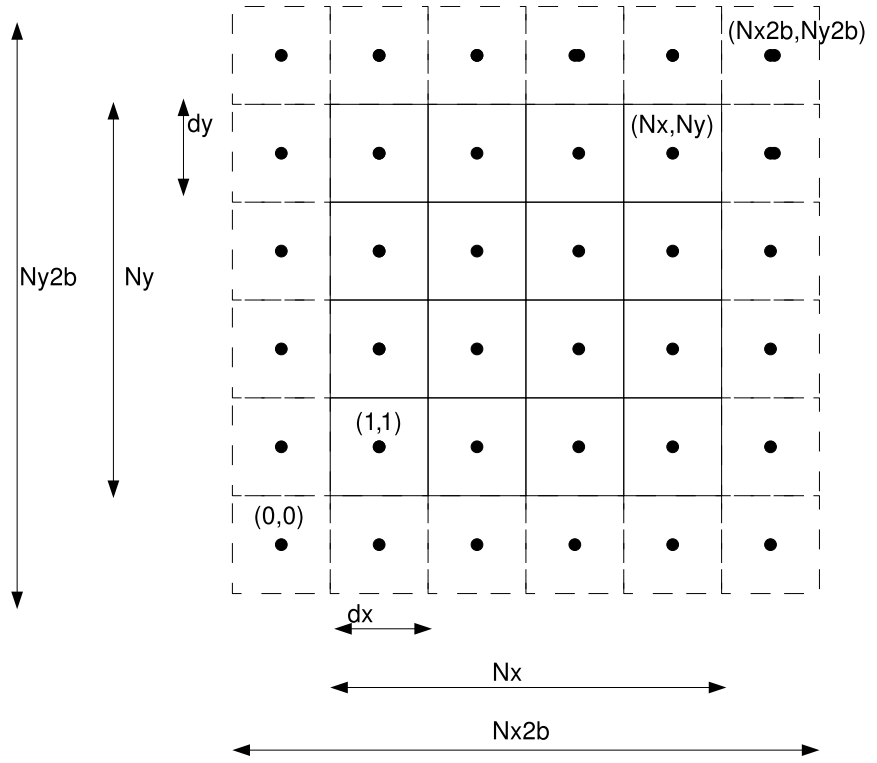


FIGURE 3.2 – Illustration des notations N_x, N_y, N_x2b, N_y2b, h_x, h_y, dx, dy.

traitées dans cette thèse. On a alors : $-\frac{\partial^2}{\partial x^2} - \frac{\partial^2}{\partial y^2}$ approximé par $\frac{1}{dx^2}$

0	-1	0
-1	4	-1
0	-1	0

.

G est représenté par le vecteur F obtenu en mettant les valeurs ligne par ligne et colonne par colonne.

On note F_{xy} la valeur de G à la colonne x et à la ligne y on a :

$$F = \begin{pmatrix} F_{11} \\ F_{21} \\ \dots \\ F_{N_x 1} \\ F_{12} \\ F_{22} \\ \dots \\ F_{N_x 2} \\ \dots \\ F_{1N_y} \\ F_{2N_y} \\ \dots \\ F_{N_x N_y} \end{pmatrix}$$

U est l'approximation discrétisée de φ. On cherche U tel que $\frac{1}{dx^2} K_{2D} U = F$ avec K_{2D} matrice de taille $N_x N_y \times N_x N_y$. Sur la ligne k de K_{2D} on trouve :

- 4 sur la colonne k
- -1 sur la colonne $k-1$
- -1 sur la colonne $k+1$
- -1 sur la colonne $k-N_x$
- -1 sur la colonne $k+N_x$

On cherche les N_xN_y valeurs propres λ_{ij} et les N_xN_y vecteurs propres ϕ_{ij}, de taille N_xN_y, de K_{2D}.
Soient :

$$\begin{aligned}\phi_{ij} &= \phi_j \otimes \phi_i \\ \lambda_{ij} &= \lambda_i + \lambda_j\end{aligned}$$

où ϕ_i est un vecteur propre de K calculé précédemment dans le cas 1D, λ_i la valeur propre associée et $\otimes$ le produit de Kronecker.
ϕ_{ij} s'écrit :

$$\phi_{ij} = \begin{pmatrix} \sin(\pi i h_x)\sin(\pi j h_y) \\ \sin(2\pi i h_x)\sin(\pi j h_y) \\ \dots \\ \sin(N_x\pi i h_x)\sin(\pi j h_y) \\ \sin(\pi i h_x)\sin(2\pi j h_y) \\ \sin(2\pi i h_x)\sin(2\pi j h_y) \\ \dots \\ \sin(N_x\pi i h_x)\sin(2\pi j h_y) \\ \dots \\ \sin(\pi i h_x)\sin(N_y\pi j h_y) \\ \sin(2\pi i h_x)\sin(N_y\pi j h_y) \\ \dots \\ \sin(N_x\pi i h_x)\sin(N_y\pi j h_y) \end{pmatrix}$$

Montrons que les ϕ_{ij} ainsi définis sont les vecteurs propres de K_{2D} associés aux valeurs propres λ_{ij}. La ligne k du produit $K_{2D}\phi_{ij}$ correspond à la ligne k de K_{2D} que multiplie ϕ_{ij}. Ici k doit être vu comme une double coordonnée $k = (k_1, k_2)$ avec k_1 indice de colonne et k_2 indice de ligne sur l'espace 2D.

$$\begin{aligned}
K_{2D}\phi_{ij} &= 4\sin(k_1\pi i h_x)\sin(k_2\pi j h_y) \\
&- \sin[(k_1-1)\pi i h_x]\sin(k_2\pi j h_y) \\
&- \sin[(k_1+1)\pi i h_x]\sin(k_2\pi j h_y) \\
&- \sin(k_1\pi i h_x)\sin[(k_2-1)\pi j h_y] \\
&- \sin(k_1\pi i h_x)\sin[(k_2+1)\pi j h_y] \\
K_{2D}\phi_{ij} &= 2(1-\cos(\pi i h_x)\sin(k_1\pi i h_x)\sin(k_2\pi i h_y) \\
&+ 2(1-\cos(\pi j h_y)\sin(k_1\pi i h_x)\sin(k_2\pi i h_y) \\
K_{2D}\phi_{ij} &= [2(1-\cos(\pi i h_x)+2(1-\cos(\pi j h_y)]\sin(k_1\pi i h_x)\sin(k_2\pi i h_y) \\
&= (\lambda_i+\lambda_j)\phi_{ij} \\
&= \lambda_{ij}\phi_{ij}
\end{aligned}$$

On connaît maintenant les valeurs propres λ_{ij} et des vecteurs propres ϕ_{ij} de K_{2D}. On normalise les ϕ_{ij} pour obtenir les ϕ'_{ij}. On peut décomposer F dans la base des ϕ'_{ij} :

$$F = \sum_j \sum_i \left\langle F, \phi'_{ij} \right\rangle \phi'_{ij} = \sum_j \sum_i a_{ij} \phi'_{ij}$$

On cherche U tel que $\frac{1}{dx^2} K_{2D} U = F$. On pose :

$$U = dx^2 \sum_j \sum_i \frac{a_{ij}}{\lambda_{ij}} \phi'_{ij} \tag{3.2.16}$$

On vérifie :

$$\begin{aligned} \frac{1}{dx^2} K_{2D} U &= \sum_j \sum_i \frac{a_{ij}}{\lambda_{ij}} K_{2D} \phi'_{ij} \\ &= \sum_j \sum_i \frac{a_{ij}}{\lambda_{ij}} \lambda_{ij} \phi'_{ij} \\ &= \sum_j \sum_i a_{ij} \phi'_{ij} \\ &= F \end{aligned}$$

Donc connaissant les ϕ'_{ij} et λ_{ij}, on peut trouver U tel que $\frac{1}{dx^2} K U = F$:

$$U = dx^2 \sum_j \sum_i \frac{\left\langle F, \phi'_{ij} \right\rangle}{\lambda_{ij}} \phi'_{ij} \tag{3.2.17}$$

Pour obtenir le mouvement, on effectue :

$$\mathbf{w} = \nabla^{\perp} U. \tag{3.2.18}$$

3.3 Projection de Galerkin pour l'obtention du modèle réduit

On présente, dans cette section la projection de Galerkin du modèle de mouvement à divergence nulle, décrit par le système 3.2.7, sur des bases Ψ_ξ et Ψ_I :

- $\Psi_\xi = \{\psi_{\xi,i}\}_{i=1,\dots K}$ est la base associée à la vorticité ξ,
- $\Psi_I = \{\psi_{I,j}\}_{j=1,\dots L}$ est la base associée à la pseudo-image I_s.

Soient $a_i(t)$ le cœfficient de projection de $\xi(\mathbf{x},t)$ sur $\psi_{\xi,i}$, $\xi(\mathbf{x},t)$ est alors approximé par $\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})$. De même, $I_s(\mathbf{x},t)$ est approximé par $\sum_{j=1}^{L} b_j(t)\psi_{I,j}(\mathbf{x})$ avec b_j le cœfficient de projection de $I_s(\mathbf{x},t)$ sur $\psi_{I,j}$.
$\xi(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ sont remplacés par leurs approximations dans le système 3.2.7 afin d'obtenir :

$$\begin{cases} \sum_{i=1}^{K} \frac{da_i}{dt}(t)\psi_{\xi,i}(\mathbf{x}) + \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right) \cdot \nabla \left(\sum_{j=1}^{K} a_j(t)\psi_{\xi,j}(\mathbf{x})\right) = 0 \\ \sum_{i=1}^{L} \frac{db_i}{dt}(t)\psi_{I,i}(\mathbf{x}) + \mathbf{w}\left(\sum_{i=1}^{K} \psi_{\xi,i}(\mathbf{x})\right) \cdot \nabla \left(\sum_{j=1}^{L} b_j(t)\psi_{I,j}(\mathbf{x})\right) = 0 \\ \sum_{i=1}^{K} a_i(0)\psi_{\xi,i}(\mathbf{x}) = \sum_{i=1}^{K} a_{i0}\psi_{\xi,i}(\mathbf{x}) \\ \sum_{j=1}^{L} b_j(0)\psi_{I,j}(\mathbf{x}) = \sum_{j=1}^{L} b_{j0}\psi_{I,j}(\mathbf{x}) \end{cases} \tag{3.3.1}$$

On fait ensuite le produit scalaire du système d'équations 3.3.1 avec $\psi_{\xi,k}$ et $\psi_{I,l}$ afin d'obtenir le système d'équations différentielles ordinaires suivant :

$$\begin{cases} \frac{da_k}{dt}(t)\left\langle \psi_{\xi,k}, \psi_{\xi,k}\right\rangle + \left\langle \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right) \cdot \nabla \left(\sum_{j=1}^{K} a_j(t)\psi_{\xi,j}\right), \psi_{\xi,k}\right\rangle = 0, k = 1, \ldots, K \\ \frac{db_l}{dt}(t)\left\langle \psi_{I,l}, \psi_{I,l}\right\rangle + \left\langle \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right) \cdot \nabla \left(\sum_{j=1}^{L} b_j(t)\psi_{I,j}\right), \psi_{I,l}\right\rangle = 0, l = 1, \ldots, L \\ a_k(0) = a_{k0}, \\ b_l(0) = b_{l0}, \end{cases} \tag{3.3.2}$$

Soit Φ la famille de vecteurs ϕ_i obtenus à partir des vecteurs $\psi_{\xi,i}$ selon l'équation 3.2.8 :

$$\Phi = \{\phi_i = \mathbf{w}(\psi_{\xi,i})\}_{i=1,\ldots K} \tag{3.3.3}$$

Les ϕ_i sont les champs de mouvement correspondant aux champs de vorticité $\psi_{\xi,i}$.

Par linéarité du produit scalaire et en divisant les équations par $< \psi_{\xi,k}, \psi_{\xi,k} >$ et $< \psi_{I,l}, \psi_{I,l} >$, le système 3.3.2 se simplifie en :

$$\begin{cases} \frac{da_k}{dt}(t) + \sum_{i=1}^{K}\sum_{j=1}^{K} a_i(t)a_j(t)\frac{\langle \phi_i \cdot \nabla\psi_{\xi,j}, \psi_{\xi,k}\rangle}{\langle \psi_{\xi,k}, \psi_{\xi,k}\rangle} = 0, k = 1, \ldots, K \\ \frac{db_l}{dt}(t) + \sum_{i=1}^{K}\sum_{j=1}^{L} a_i(t)b_j(t)\frac{\langle \phi_i \cdot \nabla\psi_{I,j}, \psi_{I,l}\rangle}{\langle \psi_{I,l}, \psi_{I,l}\rangle} = 0, l = 1, \ldots, L \\ a_k(0) = a_{k0}, \\ b_l(0) = b_{l0}, \end{cases} \tag{3.3.4}$$

ou encore :

$$\begin{cases} \frac{da_k}{dt}(t) + a^T(t)B(k)a(t) = 0, \quad k = 1 \ldots K. \\ \frac{db_l}{dt}(t) + a^T(t)G(l)b(t) = 0, \quad l = 1 \ldots L. \\ a(0) = a^0 \\ b(0) = b^0 \end{cases} \tag{3.3.5}$$

avec :

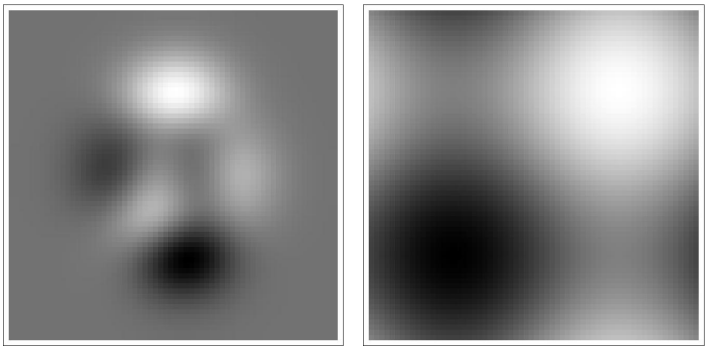
- $a(t) = (a_1(t), \ldots, a_K(t))^T$,
- $B(k)$ la matrice de taille $K \times K$ dont les éléments sont $B(k)_{i,j} = \dfrac{\langle \phi_i \cdot \nabla \psi_{\xi,j}, \psi_{\xi,k} \rangle}{\langle \psi_{\xi,k}, \psi_{\xi,k} \rangle}$,
- $b(t) = (b_1(t), \ldots, b_L(t))^T$,
- $G(l)$ la matrice de taille $K \times L$ dont les éléments sont $G(l)_{i,j} = \dfrac{\langle \phi_i \cdot \nabla \psi_{I,j}, \psi_{I,l} \rangle}{\langle \psi_{I,l}, \psi_{I,l} \rangle}$,
- a^0 vecteur de taille K,
- b^0 vecteur de taille L.

3.4 Simulation

3.4.1 Simulation du modèle complet

Le modèle complet, décrit par le système 3.2.7, est simulé à partir des conditions initiales (ξ_0, I_0), représentées sur la figure 3.3, avec les valeurs de paramètres suivantes :

- $dx = dy = 1100\,\text{m}$.
- $dt = 900\,\text{s}$.
- $Nx = Ny = 64$ soit 70 400 mètres,
- $Nt = 82$ soit 20 h30 m.



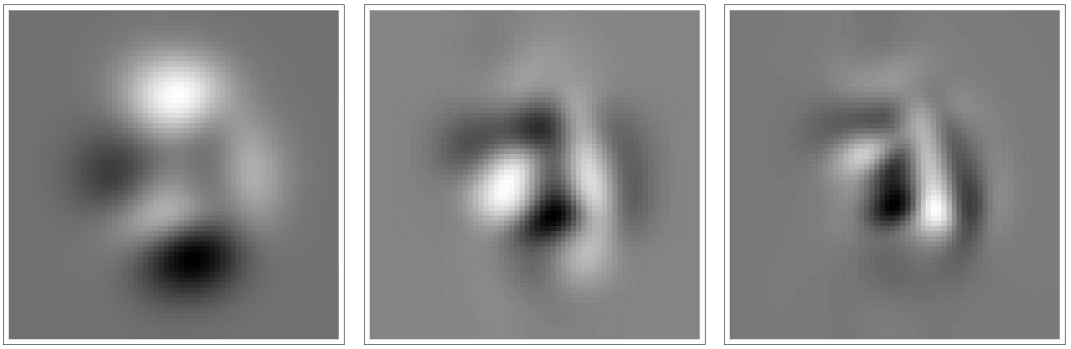
(a) ξ_0. (b) I_0.

FIGURE 3.3 – Conditions initiales de l'expérience. Gauche : vorticité. Droite : image.

On obtient deux séquences $\xi(x,t)$ et $I_s(x,t)$ de vorticité et d'images visibles sur la partie haute des figures 3.7 et 3.8.

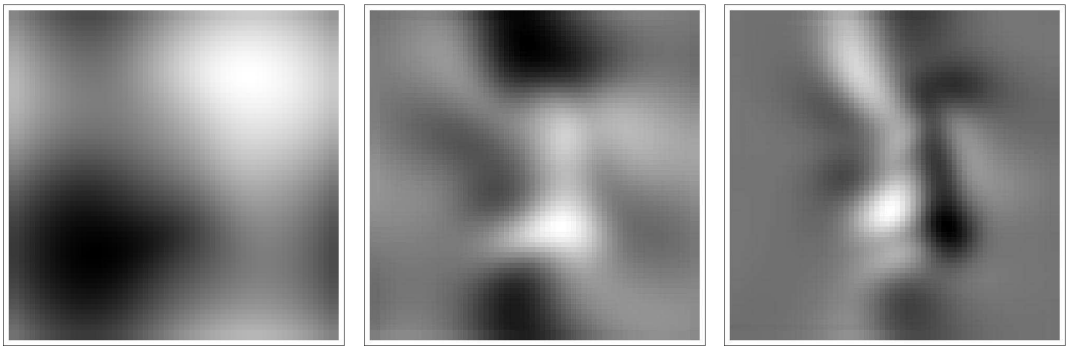
3.4.2 Simulation du modèle réduit

On choisit les dates $t_z = 15\,\text{min}, 5\,\text{h}15\,\text{min}, 5\,\text{h}15\,\text{min}, 10\,\text{h}15\,\text{min}, 20\,\text{h}15\,\text{min}$ pour former les séquences de vorticité $\xi = \{\xi^z\} = \{\xi(x,t_z)\}_{z=1\ldots Z}$ et d'images $I = \{I^z\} = \{I(x,t_z)\}_{z=1\ldots Z}$. Une POD est appliquée à ces deux séquences et fournit les bases de vorticité Ψ_ξ^f et image Ψ_I^f (l'exposant f vient du mot full, pour indiquer qu'on considère les bases complètes). On utilise ensuite les 3 premiers éléments pour former les bases Ψ_ξ et Ψ_I afin de conserver 99% de la variance. La norme de chaque élément de ces bases est, comme dans la section 2.4.1, choisie telle que $\langle \xi_0, \psi_{\xi,k} \rangle = 1, \quad k = 1, \ldots, K$ et $\langle I_0, \psi_{I,k} \rangle = 1, \quad l = 1, \ldots, L$. Ainsi, on a : $a^0 = \begin{pmatrix} 1 & \ldots & 1 \end{pmatrix}^T$ et $b^0 = \begin{pmatrix} 1 & \ldots & 1 \end{pmatrix}^T$. Les bases sont présentées sur les figures 3.4 et 3.5 $(K = 3, \quad L = 3)$.



(a) $\psi_{\xi,1}$ (b) $\psi_{\xi,2}$ (c) $\psi_{\xi,3}$

FIGURE 3.4 – Base Ψ_{ξ}.



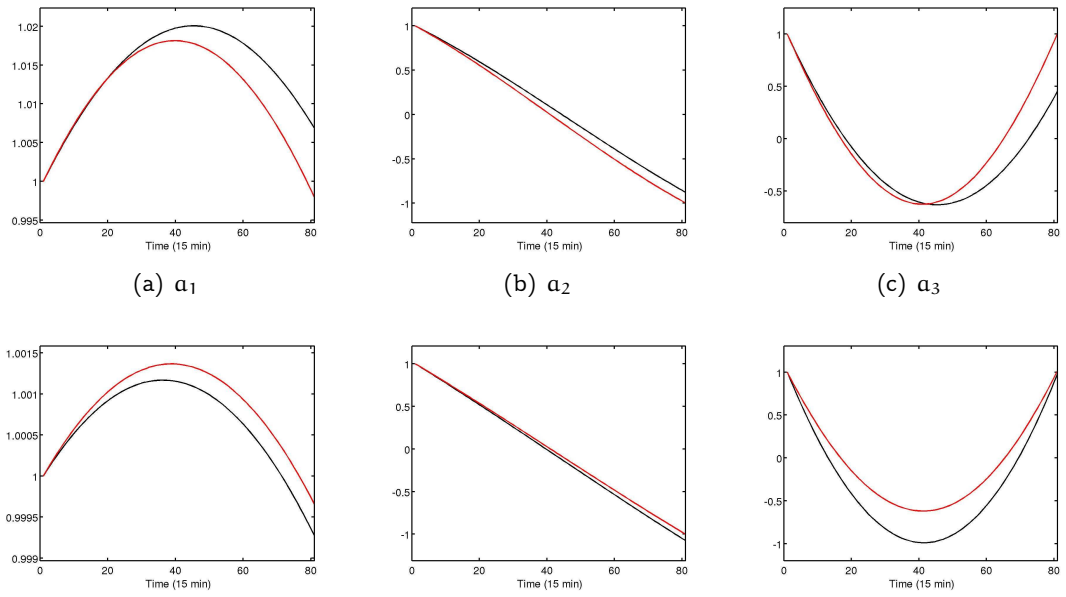
(a) $\psi_{I,1}$ (b) $\psi_{I,2}$ (c) $\psi_{I,3}$

FIGURE 3.5 – Base Ψ_{I}.

La famille de vecteurs Φ définie par l'équation 3.3.3 ainsi que les matrices $B(k)$ et $G(l)$ du système 3.3.5 sont ensuite calculées. Les dérivées spatiales intervenant dans l'expression des cœfficients $B(k)_{i,j}$ et $G(l)_{i,j}$ sont évaluées en utilisant un noyau "Cany-Deriche" de paramètre $\alpha = 1.9$.

Le modèle réduit, obtenu par projection de Galerkin sur les bases Ψ_ξ et Ψ_I du modèle complet AIMIDF est noté MR-AIMIDF_POD-POD, afin d'expliquer que ξ et I_s sont projetés sur des bases POD.

Le modèle réduit est simulé sur 82 pas de temps avec un $dt = 900\,s$, soit sur une fenêtre temporelle de 20 h30 m. La figure 3.6 présente les courbes de simulation a_i^s et b_i^s du modèle réduit, comparées aux projections a_i^p et b_i^p de la simulation du modèle complet. On observe que les courbes sont très proches. Les résultats correspondant pour la vorticité et pour les images sont présentés sur la ligne du bas des figures 3.7 et 3.8, et des statistiques d'erreur sont fournies sur le tableau 3.1. Visuellement, les séquences de vorticité et d'images des modèles complet et réduit sont semblables. Les statistiques d'erreurs sont très bonnes : les corrélations sont très proches de 1 et la NRMSE est de 0.16% pour les images et 0.25% pour la vorticité.



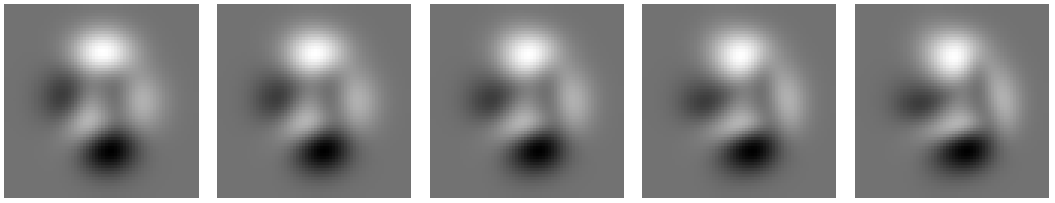
(a) a_1 (b) a_2 (c) a_3

(d) b_1 (e) b_2 (f) b_3

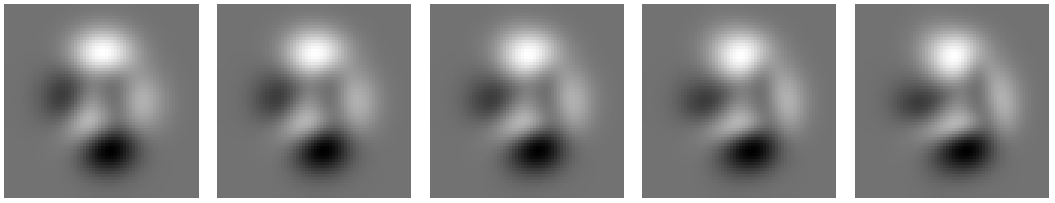
FIGURE 3.6 – Expérience de simulation, cœfficients projetés et simulés. Haut : $a_i^p(t)$ en rouge, $a_i^s(t)$ en noir. Bas : $b_i^p(t)$ en rouge, $b_i^s(t)$ en noir.

3.4.3 Conclusions

Les statistiques d'erreur, et en particulier la NRMSE, montrent que le modèle réduit, obtenu par projection de Galerkin sur les bases Ψ_ξ et Ψ_I, constitue une très bonne approximation du modèle complet 3.2.7. On réalise donc, dans la section 3.5, des expériences d'assimilation dans le modèle réduit MR-AIMIDF-POD-POD.

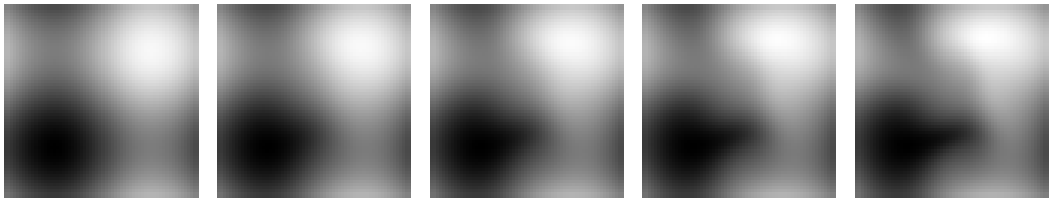


(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min

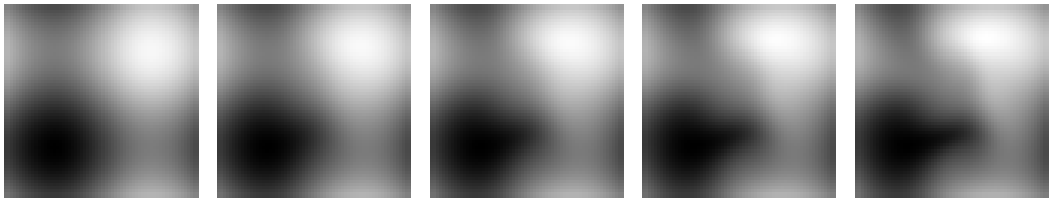


(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

FIGURE 3.7 – Simulations de vorticité ξ. Haut : modèle complet. Bas : reconstruction à partir du modèle réduit.



(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min



(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

FIGURE 3.8 – Simulations des pseudo-images I_s. Haut : modèle complet. Bas : reconstruction à partir du modèle réduit.

Vorticité ξ	valeur min	moyenne	valeur max
Modèle complet	-0.0334	2e-6	0.0378
Modèle réduit	-0.0334	2e-6	0.0379
Image I_s	valeur min	moyenne	valeur max
Modèle complet	0	0.500	1.029
Modèle réduit	0	0.500	1.031

	Vorticité	Image
RMSE	1.75e-4	1.66e-3
NRMSE (%)	0.25	0.16
Corrélation	0.99	0.99

TABLE 3.1 – Statistiques d'erreur entre la séquence simulée par le modèle complet et celle reconstruite à partir du modèle réduit.

3.5 Assimilation dans le modèle réduit

On présente, dans cette section, deux expériences d'assimilation : une expérience jumelle et une expérience cousine. La différence entre ces expériences est la manière de générer les observations :

- L'expérience jumelle utilise comme observations les cœfficients $b_j^{obs,jum}$ simulés par le modèle réduit.
- L'expérience cousine utilise comme observations les cœfficients de projection $b_j^{obs,cou}$ de la séquence $I_s(x,t)$ obtenue par le modèle complet.

Les séquences synthétiques utilisées dans ces expériences sont celles décrites dans la sous-section 3.4.1 pour l'expérience cousine et dans la sous-section 3.4.2 pour l'expérience jumelle. Les observations $b_j^{obs,jum}$ et $b_j^{obs,cou}$ sont assimilées dans le modèle réduit (système 3.3.5) sur la fenêtre temporelle $[0, 20\,\text{h}30\,\text{min}]$.

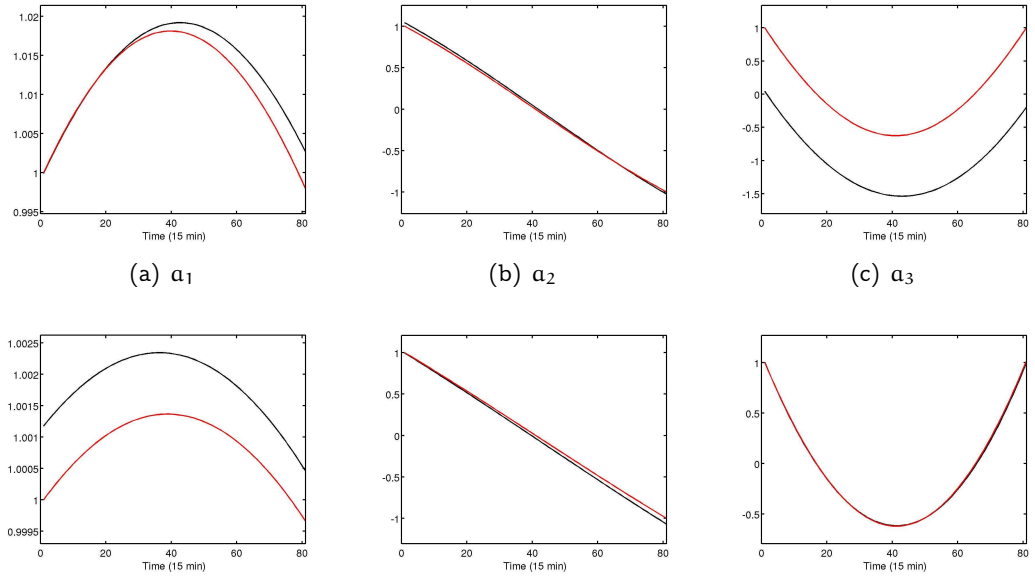
La méthode d'assimilation est celle qui a été décrite au chapitre 1. La valeur de background est 0 pour les cœfficients a_i de la vorticité et on utilise la première observation comme valeur de background sur b. La variable de contrôle est la condition initiale $\mathbf{X}_R(0) = \begin{pmatrix} a(0) \\ b(0) \end{pmatrix}$ du modèle réduit.

Les paramètres de l'expérience d'assimilation sont les suivants :

- $dt = 900\,\text{s}$.
- $Nt = 82$ soit $20\,\text{h}30\,\text{min}$,
- $B = 1$,
- $R = 0.01$,
- dates d'observation : $15\,\text{min}$, $5\,\text{h}15\,\text{min}$, $5\,\text{h}15\,\text{min}$, $10\,\text{h}15\,\text{min}$, $20\,\text{h}15\,\text{min}$

L'ordre de grandeur des cœfficients étant de 1, on choisit un écart-type de 0.1 pour l'erreur d'observation qui correspond à une variance $R = 0.01$. L'ébauche étant sur la condition initiale étant moins bien connue, on choisit un écart-type de 1 soit une variance $B = 1$.

Résultats. Pour chaque expérience, jumelle ou cousine, on obtient une séquence $\begin{pmatrix} a^a(t) \\ b^a(t) \end{pmatrix}$ (l'exposant a signifie que ce sont des résultats d'assimilation), à partir de laquelle on reconstruit l'image $I_s^a(\mathbf{x}, t)$, le champ de vorticité $\xi^a(\mathbf{x}, t)$ et le champ de mouvement $\mathbf{w}^a(\mathbf{x}, t)$, pour $t \in [0, 20\,\text{h}30\,\text{min}]$. Les courbes obtenues sont présentées sur les figures 3.9 (pour l'expérience jumelle) et 3.10 (pour l'expérience cousine). L'évolution temporelle des cœfficients b_j^a (pour les deux expériences) est très proche de l'évolution temporelle des cœfficients de projection de la simulation du modèle complet. Les courbes a_1^a et a_2^a obtenues pour les deux expériences sont elles aussi proches de la vérité terrain. Seules les courbes a_3^a ne sont pas correctement estimées (biais constant). Afin de quantifier l'écart entre les résultats d'assimilation et la vérité terrain, les champs obtenus à partir des courbes sont présentés sur les figures 3.12 à 3.13 et des statistiques d'erreurs sont fournies dans les tableaux 3.2 et 3.5. Les images I_s^a obtenues sont visuellement très proches des simulations du modèle complet. La corrélation entre les séquences estimées et la vérité terrain est de 0.99 et la NRMSE inférieure à 0.40%. On observe sur les figures 3.12 et 3.13 que les séquences de vorticité et de mouvement sont également semblables. Les NRMSE sont inférieures à 0.50% à $t = 0$ et inférieures à 0.51% sur l'ensemble des séquences. Les corrélations sont, dans tous les cas, égales à 0.99.



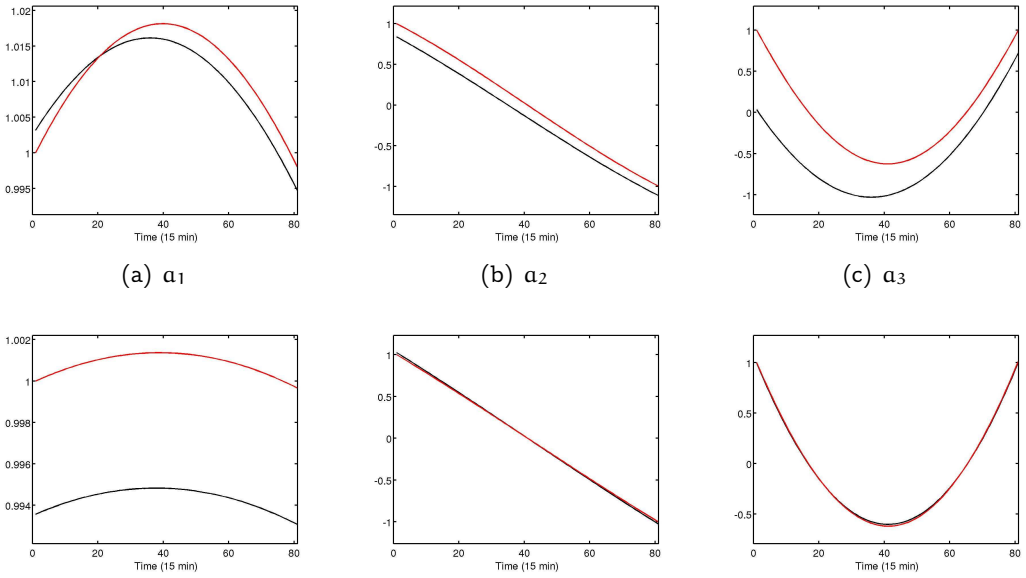
(a) a_1 (b) a_2 (c) a_3

(d) b_1 (e) b_2 (f) b_3

FIGURE 3.9 – Expérience jumelle : Vérité terrain en rouge et résultats d'assimilation en noir.

3.5.1 Conclusions

Les résultats des expériences jumelle et cousine sont très satisfaisants. Les courbes d'assimilation sont très proches des courbes de projection de la simulation du modèle complet. Les reconstructions des séquences à partir des résultats d'assimilation sont visuellement identiques à la simulation du modèle complet. De plus, les scores statistique sont bons. Les erreurs sont plus faibles dans le cas des expériences jumelles, ce qui est logique car,



(a) a_1 (b) a_2 (c) a_3

(d) b_1 (e) b_2 (f) b_3

FIGURE 3.10 – Expérience cousine : Vérité terrain en rouge et résultats d'assimilation en noir.

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Analyse	-0.0332	2e-6	0.0374
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	0.000	0.500	1.000
Analyse	0.000	0.501	1.001

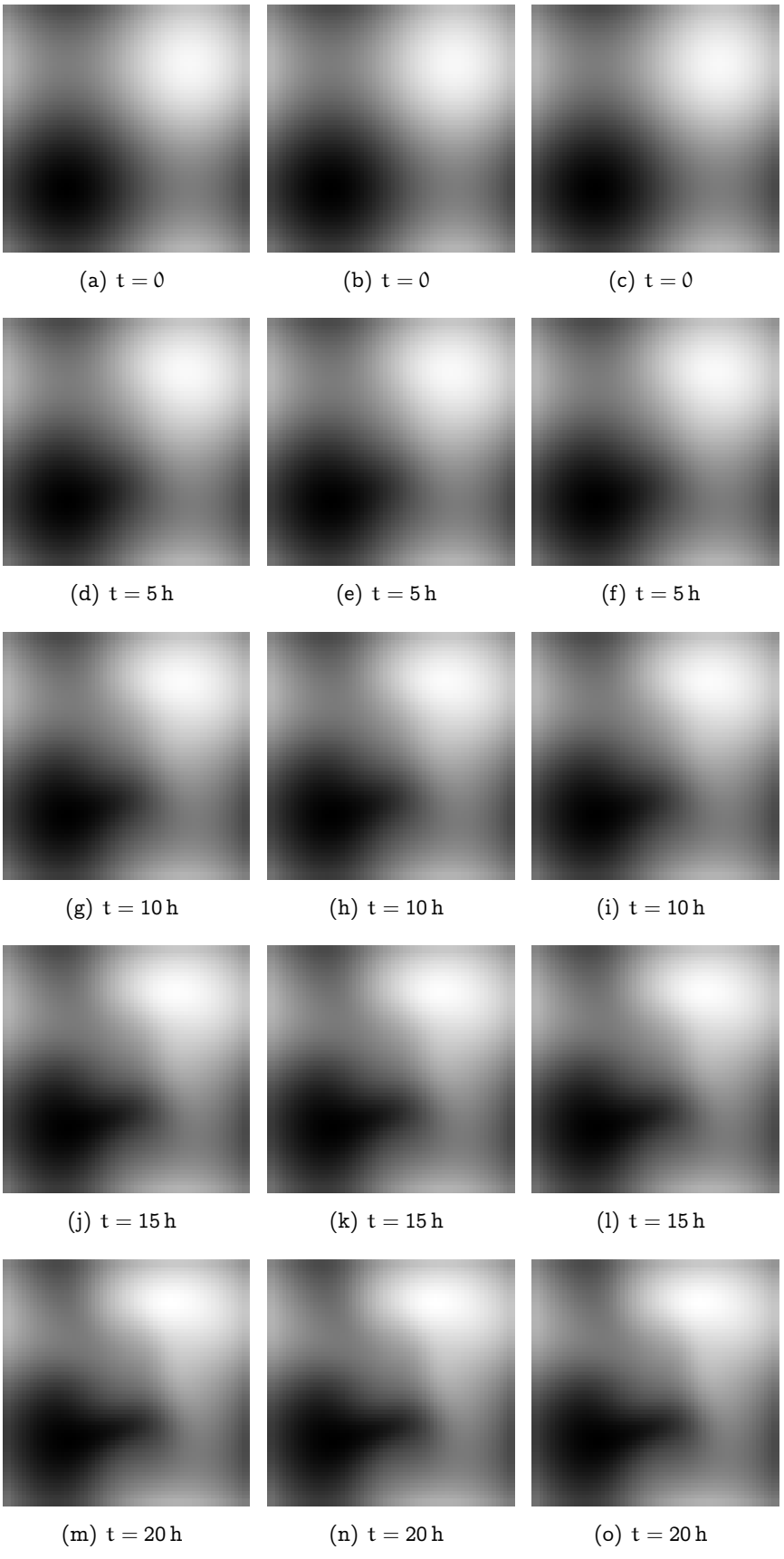
	Vorticité	Image
RMSE	3.58e-4	8.11e-4
NRMSE (%)	0.50	0.08
Corrélation	0.99	0.99

TABLE 3.2 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation de l'expérience jumelle à $t = 0$.

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Résultat d'assimilation	-0.0332	2e-6	0.0376
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	0.000	0.500	1.029
Résultat d'assimilation	0.000	0.500	1.032

	Vorticité	Image
RMSE	3.62e-4	1.37e-3
NRMSE (%)	0.51	0.13
Corrélation	0.99	0.99

TABLE 3.3 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation de l'expérience jumelle sur toutes les dates.



(a) $t = 0$ (b) $t = 0$ (c) $t = 0$

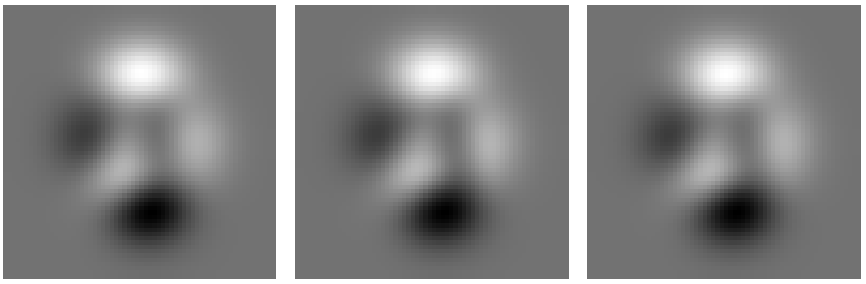
(d) $t = 5\,h$ (e) $t = 5\,h$ (f) $t = 5\,h$

(g) $t = 10\,h$ (h) $t = 10\,h$ (i) $t = 10\,h$

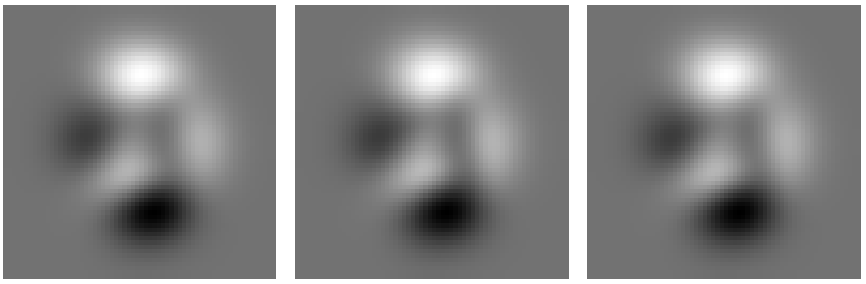
(j) $t = 15\,h$ (k) $t = 15\,h$ (l) $t = 15\,h$

(m) $t = 20\,h$ (n) $t = 20\,h$ (o) $t = 20\,h$

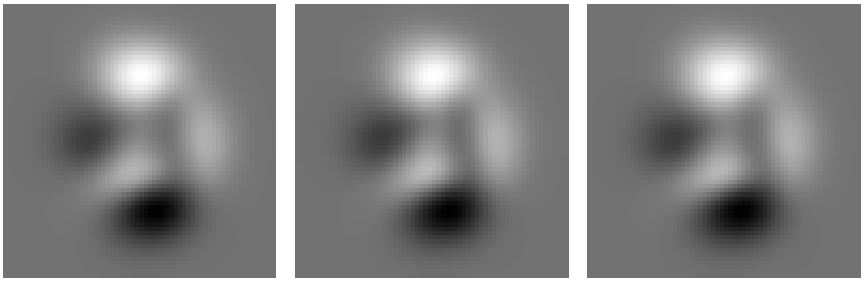
FIGURE 3.11 – $I_s(x,t)$ (gauche), $I_s^{a,jum}(x,t)$ (milieu), $I_s^{a,cou}(x,t)$ (droite).



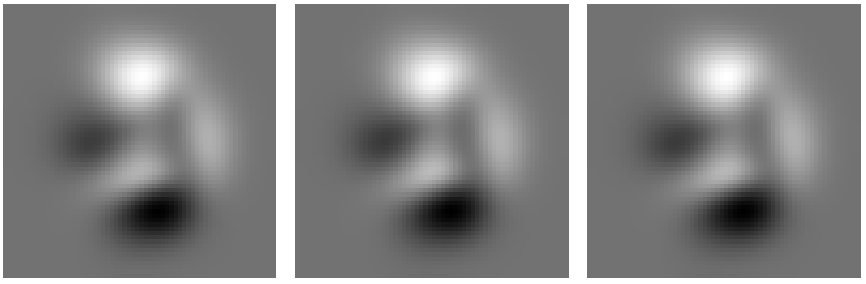
(a) $t = 0$ (b) $t = 0$ (c) $t = 0$



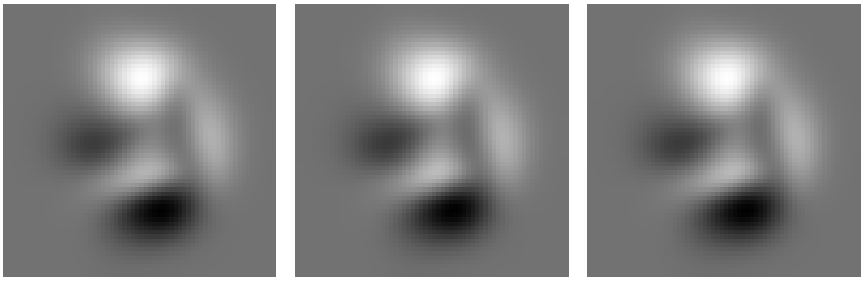
(d) $t = 5\,h$ (e) $t = 5\,h$ (f) $t = 5\,h$



(g) $t = 10\,h$ (h) $t = 10\,h$ (i) $t = 10\,h$

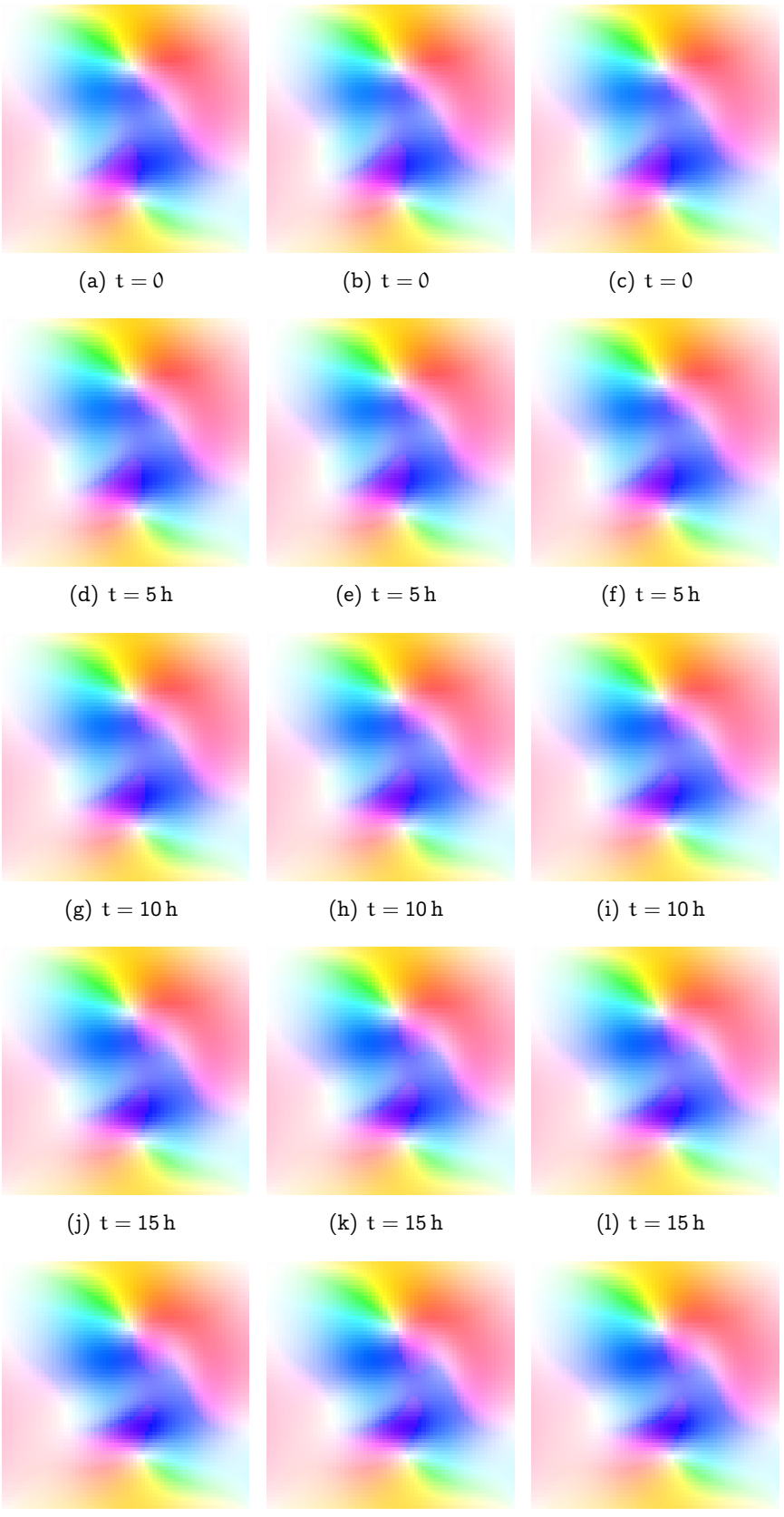


(j) $t = 15\,h$ (k) $t = 15\,h$ (l) $t = 15\,h$



(m) $t = 20\,h$ (n) $t = 20\,h$ (o) $t = 20\,h$

FIGURE 3.12 – $\xi(x,t)$ (gauche), $\xi^{a,jum}(x,t)$ (milieu), $\xi^{a,cou}(x,t)$ (droite).



(a) $t = 0$ (b) $t = 0$ (c) $t = 0$

(d) $t = 5\,h$ (e) $t = 5\,h$ (f) $t = 5\,h$

(g) $t = 10\,h$ (h) $t = 10\,h$ (i) $t = 10\,h$

(j) $t = 15\,h$ (k) $t = 15\,h$ (l) $t = 15\,h$

(m) $t = 20\,h$ (n) $t = 20\,h$ (o) $t = 20\,h$

FIGURE 3.13 – $\mathbf{w}(\mathbf{x}, t)$ (gauche), $\mathbf{w}^{a,jum}(\mathbf{x}, t)$ (milieu), $\mathbf{w}^{a,cou}(\mathbf{x}, t)$ (droite).

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Analyse	-0.0332	2e-6	0.0375
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	0.000	0.500	1.000
Analyse	0.000	0.500	0.994

	Vorticité	Image
RMSE	5.50e-4	3.69e-3
NRMSE (%)	0.63	0.37
Corrélation	0.99	0.99

TABLE 3.4 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation de l'expérience cousine à $t = 0$.

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Résultat d'assimilation	-0.0332	2e-6	0.0376
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	0.000	0.500	1.029
Résultat d'assimilation	0.000	0.497	1.023

	Vorticité	Image
RMSE	3.32e-4	3.68e-3
NRMSE (%)	0.45	0.36
Corrélation	0.99	0.99

TABLE 3.5 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation de l'expérience cousine sur toutes les dates.

dans ce cas, les observations ont étés générées avec le modèle réduit.

La taille du vecteur d'état est fortement réduite : 8192 composantes pour le modèle complet, 6 pour le modèle réduit. Le temps de calcul est par conséquent significativement diminué : 12 s pour une itération de l'algorithme de minimisation du modèle complet contre 1 ms pour une itération du modèle réduit.

Ce temps de calcul réduit permet, comme nous le montrerons dans la section 3.6, d'estimer le mouvement sur de longues fenêtres temporelles, grâce à une méthode dite de fenêtre glissante.

3.6 Expérience à fenêtre glissante

3.6.1 Introduction

Les difficultés majeures de l'estimation du mouvement par assimilation de données dans un modèle complet est, comme nous l'avons vu au chapitre 1, l'important temps de calcul ainsi que la mémoire vive nécessaire à la mise en œuvre. En effet, le temps de calcul est proportionnel à la taille des images multipliée par la durée de l'intervalle temporel. Comme nous l'avons montré dans la section 3.5, le modèle réduit MR-AIMIDF-POD-POD permet de significativement réduire le temps de calcul ainsi que la taille du vecteur d'état. Cependant, la mise en œuvre de ce modèle réduit nécessite une estimation de la vorticité initiale afin de calculer la base Ψ_ξ par décomposition orthogonale propre. On se propose donc, dans cette section, de coupler les modèles complet et réduit par une expérience à fenêtre glissante afin de profiter des qualités de chaque modèle, et de pouvoir estimer le mouvement sur de longs intervalles temporels.

3.6.2 Description de l'expérience

L'expérience "fenêtre glissante" vise à estimer le mouvement sur de longues séquences d'images. La séquence est tout d'abord découpée en sous-séquences se recouvrant de moitié. Soit Wi_m la $m^{ième}$ fenêtre ainsi obtenue. Les images de la fenêtre Wi_1 sont assimilées dans le modèle complet. Cette première étape fournit l'estimation de la vorticité sur Wi_1. La vorticité, ainsi estimée, au premier pas de temps de Wi_2 sert alors de background et permet de définir la base de vorticité sur Wi_2 et par conséquent de définir le modèle réduit MR-AIMIDF-POD-POD $\mathbb{M}_R^2$ sur cet intervalle. Les cœfficients de projection des images de Wi_2 sur la base POD sont ensuite assimilées dans le modèle réduit pour estimer la vorticité sur Wi_2. Ce résultat est de nouveau utilisé comme background de vorticité sur Wi_3 et permet de définir le modèle réduit $\mathbb{M}_R^3$ sur Wi_3. Le procédé est itéré pour estimer la vorticité sur la séquence d'images complète. L'avantage de cette technique est de réduire le temps de calcul. L'assimilation dans le modèle complet, efficace mais coûteuse en temps de calcul et en mémoire, est uniquement mise en œuvre sur la fenêtre Wi_1 dans une expérience fenêtre glissante. Sur les fenêtres suivantes, l'assimilation dans le modèle MR-AIMIDF-POD-POD réduit considérablement la complexité des calculs. La figure 3.14 résume le principe de la méthode.

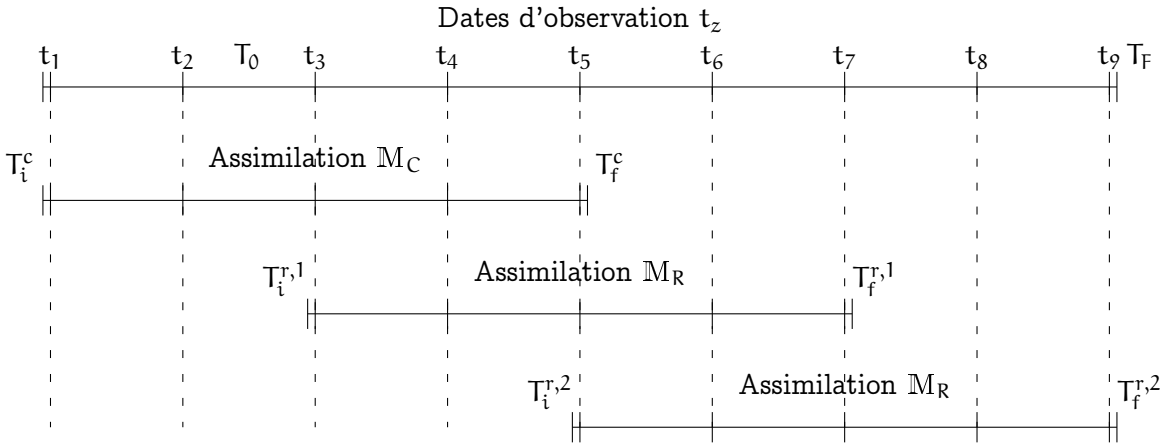


FIGURE 3.14 – Illustration de la méthode fenêtre glissante.

3.6.3 Expérience jumelle par fenêtre glissante

Séquence synthétique. Le modèle complet, décrit par le système 3.2.7, est simulé à partir des conditions initiales (ξ_0, I_0) représentées sur la figure 3.3. Une séquence $(\xi(\mathbf{x},t), I_s(\mathbf{x},t))$ est obtenue sur l'intervalle temporel $[T_0 = 0, T_F = 90\,\text{h}30\,\text{min}]$. Des dates d'observation $(t = 15\,\text{min}, 5\,\text{h}15\,\text{min}, 10\,\text{h}15\,\text{min}, \ldots, 90\,\text{h}15\,\text{min})$ sont alors choisies pour former la séquence I, qui comprend 19 images.

Assimilation dans le modèle complet AIMIDF. La première fenêtre $Wi_1 = [0, 20\,\text{h}30\,\text{min}]$ est composée des 5 premières observations. Celles-ci sont assimilées dans le modèle complet AIMIDF.

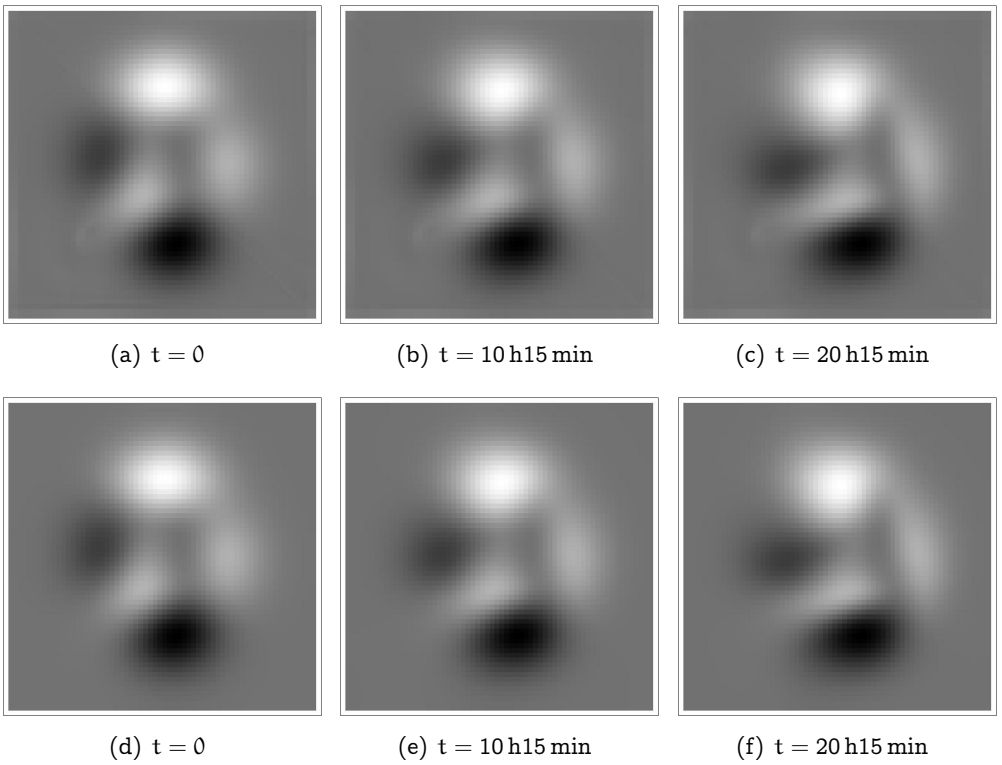
Paramètres de l'assimilation dans le modèle complet.

- $dx = dy = 1100\,\text{m}$.
- $dt = 900\,\text{s}$.
- $Nx = Ny = 64$ soit $70\,400\,\text{m}$,
- $Nt = 82$ soit $20\,\text{h}30\,\text{min}$,
- $B = 0.25$,
- $R = 0.25$,
- dates d'observation : 15 min, 5 h15 min, 10 h15 min, 15 h15 min, 20 h15 min

L'ordre de grandeur des observations et de l'ébauche étant de 1, on choisit des écarts-types de 0.5, qui correspondent aux variances $R = B = 0.25$.

Cette étape d'assimilation fournit un champ de vorticité $\xi^a(\mathbf{x},t)$ pour $t \in [0, 20\,\text{h}30\,\text{min}]$, qui permet de restituer $\mathbf{w}^a(\mathbf{x},t)$ (exposant a pour assimilation). Les résultats sont présentés sur les figures 3.15 et 3.16.

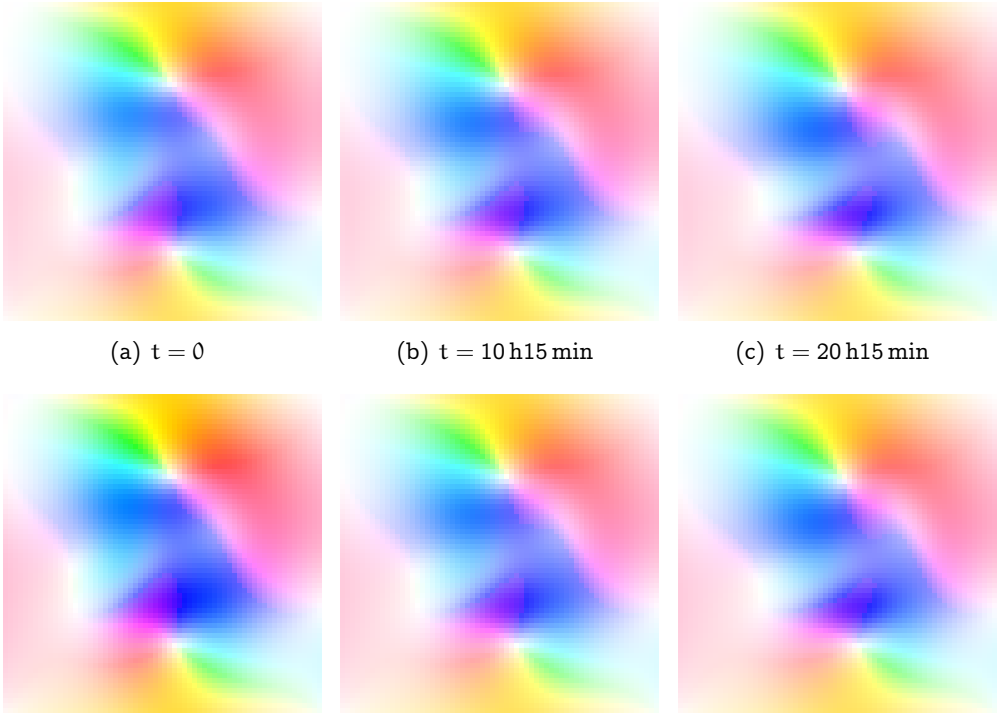
Assimilation dans le modèle réduit MR-AIMIDF-POD-POD, pour la $m^{\text{ième}}$ fenêtre. Dans la fenêtre Wi_m, on assimile les cœfficients de projection des images dans



(a) $t = 0$ (b) $t = 10\,\text{h}15\,\text{min}$ (c) $t = 20\,\text{h}15\,\text{min}$

(d) $t = 0$ (e) $t = 10\,\text{h}15\,\text{min}$ (f) $t = 20\,\text{h}15\,\text{min}$

FIGURE 3.15 – Champ $\xi^a(\mathbf{x}, t)$ restitué après assimilation dans le modèle complet (haut) – Vérité terrain $\xi(\mathbf{x}, t)$ (bas).



(a) $t = 0$ (b) $t = 10\,\text{h}15\,\text{min}$ (c) $t = 20\,\text{h}15\,\text{min}$

(d) $t = 0$ (e) $t = 10\,\text{h}15\,\text{min}$ (f) $t = 20\,\text{h}15\,\text{min}$

FIGURE 3.16 – Champ $\mathbf{w}^a(\mathbf{x}, t)$ restitué après assimilation dans le modèle complet (haut) – Vérité terrain $\mathbf{w}(\mathbf{x}, t)$ (bas).

le modèle réduit $\mathbb{M}_R^m$, comme décrit dans la sous-section 3.6.2. Les Wi_m sont définies de la façon suivante :

- $Wi_2 = [10\,\text{h}, 30\,\text{h}30\,\text{min}]$
- $Wi_3 = [20\,\text{h}, 40\,\text{h}30\,\text{min}]$
- $Wi_4 = [30\,\text{h}, 50\,\text{h}30\,\text{min}]$
- $Wi_5 = [40\,\text{h}, 60\,\text{h}30\,\text{min}]$
- $Wi_6 = [50\,\text{h}, 70\,\text{h}30\,\text{min}]$
- $Wi_7 = [60\,\text{h}, 80\,\text{h}30\,\text{min}]$

Le tableau 3.6 présente les dates d'observation pour chaque fenêtre temporelle Wi_m.

fenêtre temporelle	Wi_2	Wi_3	Wi_4	Wi_5	Wi_6	Wi_7
observation 1	10 h15 min	20 h15 min	30 h15 min	40 h15 min	50 h15 min	60 h15 min
observation 2	15 h15 min	25 h15 min	35 h15 min	45 h15 min	55 h15 min	65 h15 min
observation 3	20 h15 min	30 h15 min	40 h15 min	50 h15 min	60 h15 min	70 h15 min
observation 4	25 h15 min	35 h15 min	45 h15 min	55 h15 min	65 h15 min	75 h15 min
observation 5	30 h15 min	40 h15 min	50 h15 min	60 h15 min	70 h15 min	80 h15 min

TABLE 3.6 – Expérience fenêtre glissante – Dates d'observations.

Calcul des bases Ψ_I^m et Ψ_ξ^m pour la $m^{\text{ième}}$ fenêtre.

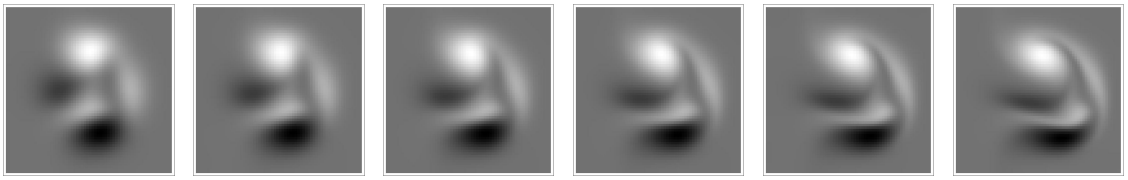
- La base réduite image Ψ_I^m est calculée en utilisant les observations image de la fenêtre Wi_m.
- Pour calculer la base Ψ_ξ^m ($m > 1$), on génère une séquence de vorticité en simulant le modèle complet sur la fenêtre Wi_m. On utilise comme condition initiale le champ de vorticité obtenu comme résultat de l'assimilation sur la fenêtre Wi_{m-1} au premier pas de temps de Wi_m.

Paramètres de l'assimilation dans le modèle réduit POD-POD. Sur chaque fenêtre Wi_m, les bases Ψ_I et Ψ_ξ sont composées de 3 éléments ($K = L = 3$). Les paramètres suivants sont communs à toutes les fenêtres Wi_m :

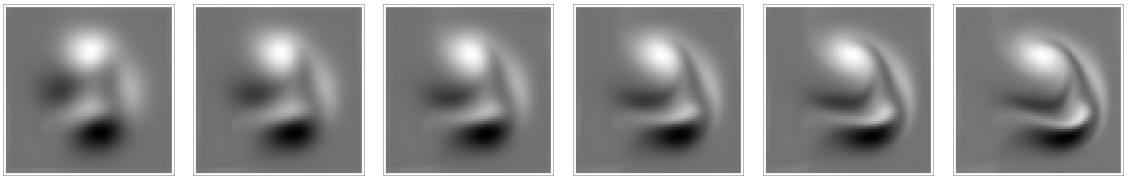
- $dt = 900s$,
- $Nt = 82$ soit 20 h30 min,
- $R = B = 10^{-2}$ (écart-type de 10^{-1}),
- Nombre d'observations : 5.

Les valeurs des cœfficients des observations et de l'ébauche sont de l'ordre de 1. On choisit donc des écarts-types d'erreur d'observation et d'ébauche de 0.1. Ces valeurs correspondent à des variances $R = B = 10^{-2}$.

Résultats. On présente, sur les figures 3.17 et 3.18, les résultats d'assimilation pour la vorticité et le mouvement. Quelques statistiques d'erreur sont fournies sur le tableau 3.7. Les séquences estimées sont très proches de la vérité terrain sur les 6 fenêtres traitées par le modèle réduit MR-AIMIDF-POD-POD. Les séquences I_s^a estimées par assimilation sont fortement corrélées aux observations (corrélation > 0.99). La corrélation entre ξ et ξ^a est de 0.99 sur la fenêtre Wi_2 et décroît ensuite jusque 0.90 sur la fenêtre Wi_7. Par ailleurs, la NRMSE est de 1.15% sur Wi_2 et ne dépasse pas 6.83% sur l'ensemble des fenêtres Wi_m.

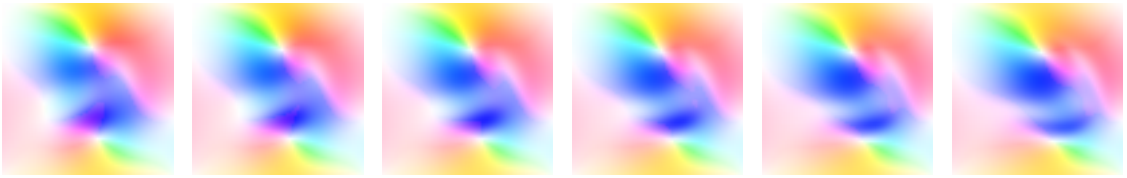


(a) 15 h15 min (b) 25 h15 min (c) 35 h15 min (d) 45 h15 min (e) 55 h15 min (f) 65 h15 min

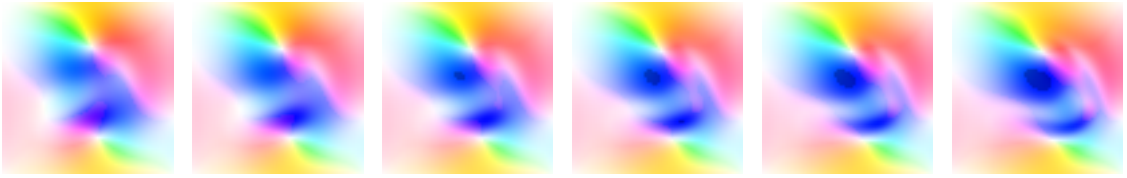


(g) 15 h15 min (h) 25 h15 min (i) 35 h15 min (j) 45 h15 min (k) 55 h15 min (l) 65 h15 min

FIGURE 3.17 – Vérité terrain ξ (haut), comparée au résultat d'assimilation ξ^a (bas).



(a) 15 h15 min (b) 25 h15 min (c) 35 h15 min (d) 45 h15 min (e) 55 h15 min (f) 65 h15 min



(g) 15 h15 min (h) 25 h15 min (i) 35 h15 min (j) 45 h15 min (k) 55 h15 min (l) 65 h15 min

FIGURE 3.18 – Vérité terrain $\mathbf{w}$ (haut), comparée au résultat d'assimilation $\mathbf{w}^a$ (bas).

fenêtre temporelle	1	2	3	4
RMSE Vorticité	$8.18e-4$	$9.44e-4$	$1.191e-3$	$1.60e-3$
NRMSE Vorticité (%)	1.15	1.34	1.68	2.25
Corrélation Vorticité	0.99	0.99	0.99	0.99
RMSE Pseudo-image	$1.53e-3$	$3.11e-3$	$4.35e-3$	$5.33e-3$
NRMSE Pseudo-image (%)	0.15	0.30	0.42	0.51
Corrélation Pseudo-image	0.99	0.99	0.99	0.99

fenêtre temporelle	5	6	7
RMSE Vorticité	$2.15e-3$	$2.90e-3$	$4.94e-3$
NRMSE Vorticité (%)	3.00	4.03	6.83
Corrélation Vorticité	0.98	0.96	0.90
RMSE Pseudo-image	$6.18e-3$	$5.66e-3$	$7.52e-3$
NRMSE Pseudo-image (%)	0.59	0.55	0.74
Corrélation Pseudo-image	0.99	0.99	0.99

TABLE 3.7 – Statistiques d'erreur entre la vérité terrain $(\xi(\mathbf{x},t), I_s(\mathbf{x},t))$ et les séquences reconstruites $(\xi^a(\mathbf{x},t), I_s^a(\mathbf{x},t))$.

3.6.4 Remarques et conclusions

Remarques.

- Le temps de calcul nécessaire à l'assimilation dans le modèle complet sur la première fenêtre $[0, 82]$ est de plusieurs heures (environ 4 heures pour une image de taille 64×64 pixels). Le temps de calcul pour les 6 fenêtres d'assimilation du modèle réduit est inférieur à 1 minute.
- Le vecteur d'état du modèle complet comporte $64 \times 64 \times 2 = 8192$ composantes alors que le vecteur d'état du modèle réduit ne comporte que 6 composantes.
- Les statistiques d'erreur calculées entre la vérité terrain $(\xi(\mathbf{x}, t), I_s(\mathbf{x}, t))$ et les séquences estimées $(\xi^a(\mathbf{x}, t), I_s^a(\mathbf{x}, t))$ sur les 6 premières fenêtres d'assimilation sont très faibles. La corrélation entre les séquences estimées et la vérité terrain pour la pseudo-image est égale à 0.99 sur chacune des fenêtres. Pour la vorticité, la corrélation est supérieure à 0.96 sur les 6 premières fenêtres. Elle décroît de 0.99 sur la première fenêtre jusqu'à 0.90 pour la septième et dernière.
- La NRMSE sur I_s n'excède pas 0.55% sur les 6 premières fenêtres. La NRMSE sur ξ n'excède pas 4.03% sur les 6 premières fenêtres. Elle est croissante, ce qui semble normal, de 1.15% sur la première fenêtre à 4.03% sur la sixième.
- La vorticité, et donc les champs de mouvement, sont très bien estimés sur les 7 fenêtres d'assimilation correspondant à la fenêtre temporelle $[0\,\text{min}, 80\,\text{h}15\,\text{min}]$.

Conclusions.

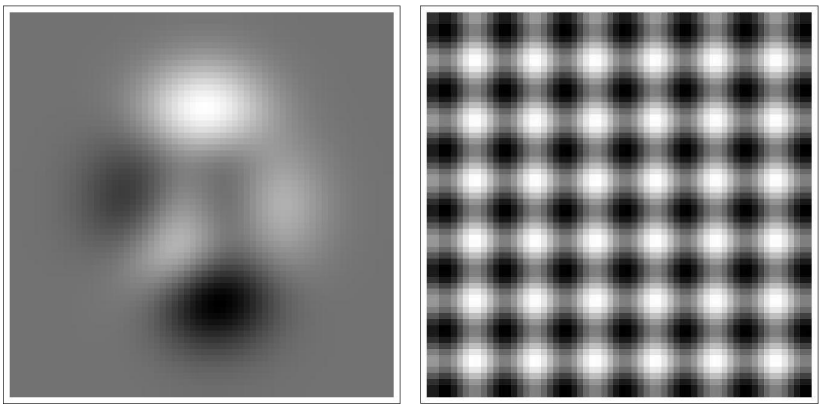
- L'objectif de réduction du temps de calcul et de mémoire utilisée est atteint : le temps de calcul est très faible ($< 1\,\text{min}$) et le vecteur d'état du modèle réduit n'a que 6 composantes. Il est donc possible d'estimer le mouvement sur de longues séquences d'images en utilisant cette technique de réduction.
- Lorsque l'on dispose d'une bonne estimation de la condition initiale, il est possible, à l'aide d'une séquence d'images, d'estimer le mouvement de manière rapide et satisfaisante en utilisant un modèle réduit POD.

3.7 Comparaison avec l'état de l'art

Une autre expérience jumelle d'assimilation dans le modèle réduit a été réalisée pour comparer les résultats de la méthode proposée aux résultats de quelques méthodes de l'état de l'art. Une simulation du modèle complet a été réalisée à partir des conditions initiales présentées sur la figure 3.19. Les paramètres de la simulation sont les suivants :

- $dx = dy = 1100\,\text{m}$.
- $dt = 900\,\text{s} = 15\,\text{min}$.
- $Nx = Ny = 64$ soit $70\,400\,\text{m}$,
- $Nt = 82$ soit $20\,\text{h}30\,\text{m}$.

Des observations ont été choisies aux dates $t = 15\,\text{min}, 5\,\text{h}15\,\text{min}, \ldots, 20\,\text{h}15\,\text{min}$. Les résultats ainsi que la vérité terrain sont présentés sur la figure 3.20. Ces résultats sont comparés à 7 méthodes de l'état de l'art [7, 24, 25, 26, 34, 41, 42]. Pour ces méthodes, des paramètres optimaux ont été choisis. Quelques statistiques sont fournies sur le tableau 3.8. Les méthodes d'assimilation dans les modèles complet et réduit à divergence nulle obtiennent les meilleurs résultats. Les erreurs du modèle réduit sont légèrement



(a) ξ_0. (b) I_0.

FIGURE 3.19 – Conditions initiales de l'expérience. Gauche : vorticité. Droite : Image.

Méthode	Erreur Angulaire (deg) Moyenne	écart-type	Erreur relative en norme Moyenne (%)
Horn[24]	15.26	9.65	45.75
Papadakis [34]	13.89	5.03	45.59
Corpetti[7]	12.54	9.49	38.70
Isambert [26]	10.61	6.92	34.84
Suter[42]	10.41	5.34	37.65
Huot [25]	9.91	7.38	31.45
Sun [41]	8.76	4.26	29.07
Modèle complet	0.18	0.10	0.65
Modèle réduit	0.19	0.11	6.50

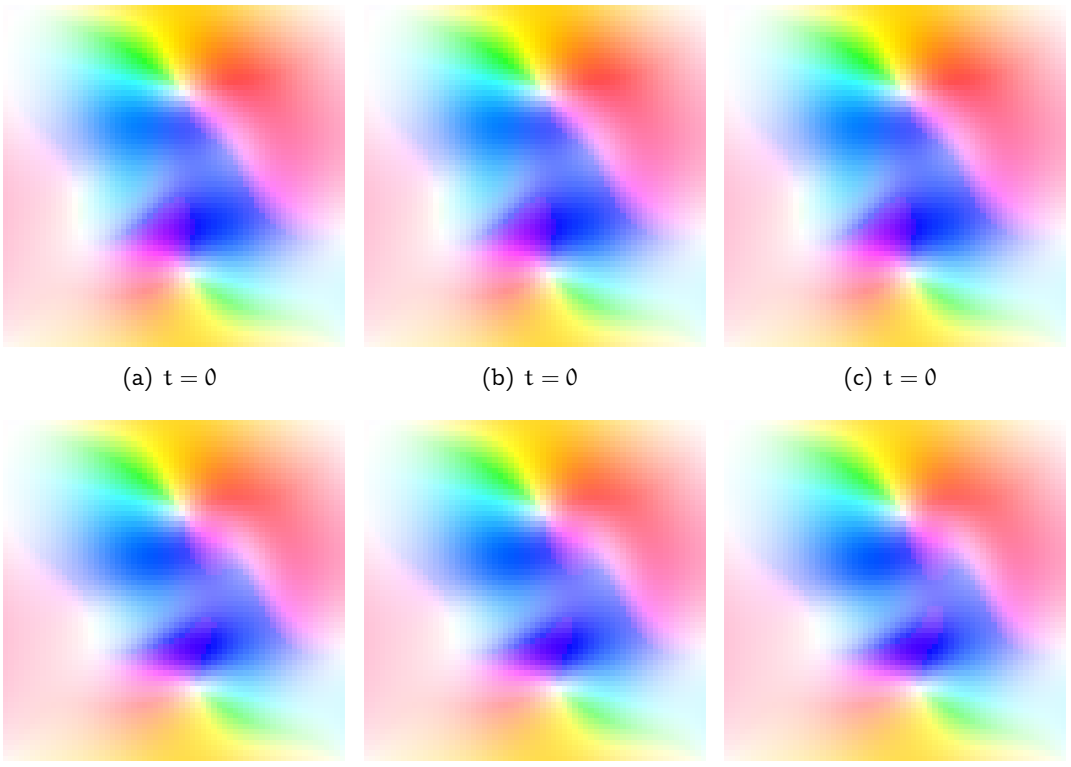
TABLE 3.8 – Statistiques d'erreur entre la vérité terrain et les estimations du mouvement.

plus importantes que celles du modèle complet, mais restent raisonnables et largement inférieures à celles obtenues par des méthodes l'état de l'art.

3.8 Conclusions

Nous avons présenté dans ce chapitre un modèle d'évolution du mouvement et des images sous l'hypothèse d'une divergence nulle. Ce modèle est adapté à l'estimation du mouvement sur des séquences d'images satellite en océanographie. Le modèle réduit, projection du modèle complet sur des bases POD, permet de réduire de manière importante le temps de calcul de l'estimation par assimilation de données. Nous avons présenté une application de ce modèle réduit dans une expérience à fenêtre glissante. Cette méthode est particulièrement adaptée au traitement de longue séquences d'images. Enfin les estimations des modèles réduit et complet ont été comparées, à travers une expérience jumelle, aux résultats des méthodes de l'état de l'art.

Les bases POD, utilisées dans ce chapitre, dépendent de la séquence d'image à traiter. Dans la partie II nous allons présenter l'utilisation de bases fixes pour l'estimation du



(a) $t = 0$ (b) $t = 0$ (c) $t = 0$

(d) $t = 81$ (e) $t = 81$ (f) $t = 81$

FIGURE 3.20 – Vérité terrain (gauche), estimation par le modèle complet (centre), estimation par le modèle réduit (droite).

mouvement. Celles-ci permettront d'éviter l'utilisation du modèle complet, même sur la première fenêtre temporelle. Ces bases devront satisfaire des propriétés optimales pour l'estimation du mouvement et permettre de traiter de longues séquences d'images sur des domaines spatiaux complexes, tel que celui de la mer Noire par exemple.

Deuxième partie

Bases fixes pour l'estimation du mouvement

Chapitre 4

Projection du modèle à divergence nulle sur une base de fonctions sinus

4.1 Introduction

Les algorithmes d'assimilation décrits dans les chapitres 1 à 3, ainsi que dans les articles [14, 21], ont en commun le problème de l'estimation de la base réduite pour le mouvement. Les solutions proposées passent par l'estimation d'un champ $\mathbf{w}_0(\mathbf{x})$, intégré en temps par le modèle complet, afin de générer une séquence à partir de laquelle la base du mouvement est obtenue. La mise en œuvre des méthodes proposées nécessite donc un pré-traitement permettant l'obtention de $\mathbf{w}_0(\mathbf{x})$. Afin de contourner ce problème, on présente, dans ce chapitre, l'utilisation d'une base réduite fixe.

Des représentations paramétriques des champs de mouvement pour son estimation ont été utilisées dans [9, 26, 33, 40, 43, 46]. Le principal intérêt de ces méthodes est que la solution estimée bénéficie des propriétés spatiales choisies pour construire la base (régularité spatiale par exemple). La plupart de ces méthodes [33, 40, 46] ne prennent pas en compte la cohérence temporelle des champs de mouvement. Dans [9, 43], les fonctions servant à paramétrer le mouvement sont choisies en fonction des images.

Dans ce chapitre, le mouvement et les images sont projetés sur une base fixe ne dépendant pas du temps, et l'estimation du mouvement est réalisée par un algorithme d'assimilation de données dans un modèle réduit. Le modèle complet, utilisé à titre d'illustration, est le modèle à divergence nulle, présenté au chapitre 3. La base choisie est la base de fonctions sinus, utilisée pour la méthode algébrique de résolution de l'équation de Poisson dans le même chapitre.

La projection du modèle complet sur cette base réduite fixe permet, comme nous le verrons dans la sous-section 4.7.2, dédiée aux résultats, d'imposer la régularité spatiale de la solution. De plus, utiliser une représentation globale du mouvement, au moyen des cœfficients de la base, permet d'effectuer une intégration temporelle simplifiée (il s'agit d'équations différentielles ordinaires) d'un modèle réduit et d'éviter les problèmes d'instabilité numérique de l'intégration temporelle d'une équation aux dérivées partielles.

4.2 Rappels du formalisme

Soit $\mathbf{X}(\mathbf{x},t) = \begin{pmatrix} \xi(\mathbf{x},t) \\ I_s(\mathbf{x},t) \end{pmatrix}$ le vecteur d'état du modèle complet, composé de la vorticité $\xi(\mathbf{x},t)$ et de la pseudo-image $I_s(\mathbf{x},t)$, définies sur le domaine spatio-temporel borné $A = \Omega \times [0,T]$. On rappelle les équations d'évolution, dans le cas d'un mouvement à divergence nulle :

$$\begin{cases} \dfrac{\partial \xi}{\partial t}(\mathbf{x},t) + \mathbf{w}(\xi)\cdot\nabla\xi(\mathbf{x},t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x},t) + \mathbf{w}(\xi)\cdot\nabla I_s(\mathbf{x},t) = 0 \end{cases} \tag{4.2.1}$$

$\mathbf{w}(\xi)$ est le champ de vitesse associé à la vorticité ξ. Il a pour expression : $\mathbf{w}(\xi) = -\nabla^{\perp}\Delta^{-1}(\xi)$. En notant $\mathbb{M}$ le modèle complet, le système 4.2.1 se simplifie en :

$$\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}(\mathbf{X}) = 0 \tag{4.2.2}$$

Rappelons les grands éléments de l'obtention du modèle réduit MR-AIMIDF par projection de Galerkin du modèle AIMIDF. La vorticité et les pseudo-images sont approximées par leurs projections sur les bases réduites $\Psi_\xi = \{\psi_{\xi,i}\}_{i=1\ldots K}$, $\Psi_I = \{\psi_{I,j}\}_{j=1\ldots L}$:

$$\xi(\mathbf{x},t) \approx \sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x}) \tag{4.2.3}$$

$$I_s(\mathbf{x},t) \approx \sum_{j=1}^{L} b_j(t)\psi_{I,j}(\mathbf{x}) \tag{4.2.4}$$

La projection de Galerkin du système 4.2.1 s'écrit :

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T(t)B(k)a(t) = 0, \quad k = 1\ldots K. \\ \dfrac{db_l}{dt}(t) + a^T(t)G(l)b(t) = 0, \quad l = 1\ldots L. \end{cases} \tag{4.2.5}$$

avec :

- $a(t) = \begin{pmatrix} a_1(t) & \ldots & a_K(t) \end{pmatrix}^T$,
- $B(k)$ la matrice de taille $K\times K$ dont l'élément (i,j) est $B(k)_{i,j} = \dfrac{\langle \mathbf{w}(\psi_{\xi,i})\cdot\nabla\psi_{\xi,j}, \psi_{\xi,k}\rangle}{\langle \psi_{\xi,k}, \psi_{\xi,k}\rangle}$,
- $b(t) = \begin{pmatrix} b_1(t) & \ldots & b_L(t) \end{pmatrix}^T$,
- $G(l)$ la matrice de taille $K\times L$ dont l'élément (i,j) est $G(l)_{i,j} = \dfrac{\langle \mathbf{w}(\psi_{\xi,i})\cdot\nabla\psi_{I,j}, \psi_{I,l}\rangle}{\langle \psi_{I,l}, \psi_{I,l}\rangle}$,

On note $\mathbb{M}_R$ le modèle réduit et $\mathbf{X}_R(\mathbf{x},t) = \begin{pmatrix} a(t) \\ b(t) \end{pmatrix}$ le vecteur d'état réduit. Le système 4.2.5 se réécrit :

$$\frac{d\mathbf{X}_R}{dt} + \mathbb{M}_R(\mathbf{X}_R) = 0 \tag{4.2.6}$$

4.3 Espace réduit – Base de fonctions sinus

Rappelons tout d'abord quelques notations :

- les pas de discrétisation spatiale sont respectivement notés dx et dy. On suppose $dx = dy$, ce qui est le cas des images satellite utilisées pour les applications.
- le nombre de pixels dans les directions x et y est noté respectivement Nx et Ny.

Une image discrète est donc associée à un vecteur à $NxNy$ composantes. Les composantes u et v du mouvement $\mathbf{w}$ et la vorticité ξ sont également représentées par des vecteurs de même taille.

Pour approximer la vorticité, on considère l'espace réduit engendré en choisissant pour base Ψ_ξ la base sinus décrite au chapitre 3. L'expression du $i^{\text{ème}}$ élément de la base discrète est la suivante :

$$\psi_{\xi,i} = \psi_{\xi,(i_1,i_2)} = \begin{pmatrix} \sin(\pi i_1 h_x)\sin(\pi i_2 h_y) \\ \sin(2\pi i_1 h_x)\sin(\pi i_2 h_y) \\ \dots \\ \sin(N_x \pi i_1 h_x)\sin(\pi i_2 h_y) \\ \sin(\pi i_1 h_x)\sin(2\pi i_2 h_y) \\ \sin(2\pi i_1 h_x)\sin(2\pi i_2 h_y) \\ \dots \\ \sin(N_x \pi i_1 h_x)\sin(2\pi i_2 h_y) \\ \dots \\ \sin(\pi i_1 h_x)\sin(N_y \pi i_2 h_y) \\ \sin(2\pi i_1 h_x)\sin(N_y \pi i_2 h_y) \\ \dots \\ \sin(N_x \pi i_1 h_x)\sin(N_y \pi i_2 h_y) \end{pmatrix}$$

avec :

- $i = (i_1, i_2)$ un indice double,
- $h_x = \dfrac{1}{N_x + 1}$,
- $h_y = \dfrac{1}{N_y + 1}$.

$\psi_{\xi,i}$ est un vecteur propre de l'opérateur Δ, associé à la valeur propre :

$$\lambda_{(i_1,i_2)} = 2[1 - \cos(\pi i_1 h_x) + 1 - \cos(\pi i_2 h_y)] \times \frac{1}{d_x^2} \tag{4.3.1}$$

La base Ψ_ξ est choisie pour représenter la vorticité parce qu'elle présente les propriétés suivantes :

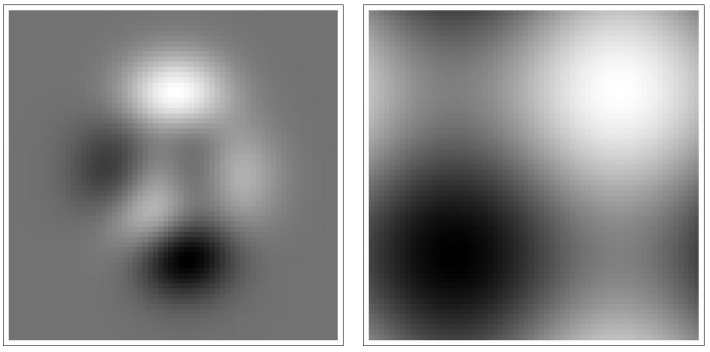
- contrairement à une base POD, estimée à partir d'un champ de vitesse initial $\mathbf{w}_0$, cette base est fixe et ne nécessite pas d'effectuer un quelconque prétraitement des images,
- la représentation globale des champs de vorticité sur cette base, au moyen des coefficients de projection permet d'utiliser l'intégration temporelle simplifiée d'un modèle réduit et d'éviter les instabilités numériques locales,
- la solution estimée est spatialement régulière, car c'est la somme pondérée de fonctions elles-même régulières. Cette régularité spatiale de la solution constitue un

avantage majeur dans le cas de l'estimation du mouvement à partir d'images satellite très bruitées,
- contrairement à une base périodique de Fourier, Ψ_ξ vérifie les conditions aux limites de Dirichlet : $\forall \mathbf{x} \in \Gamma, \quad \psi_{\xi,i}(\mathbf{x}) = 0$ avec Γ la frontière du domaine spatial Ω. Le modèle réduit vérifie ainsi les mêmes conditions aux bords que le modèle complet.

4.4 Simulation

4.4.1 Simulation modèle complet

Le modèle complet (4.2.1) est simulé à partir des conditions initiales (ξ_0, I_0) présentées sur la figure 4.1.



(a) Condition initiale de vorticité ξ_0.

(b) Condition initiale image I_0.

FIGURE 4.1 – Conditions initiales de l'expérience. Gauche : vorticité. Droite : image.

Le modèle complet est simulé avec les valeurs suivantes des paramètres :
- $dx = dy = 1100\,m$,
- $dt = 900\,s = 15\,min$,
- $Nx = Ny = 64$ soit $70\,400\,m$,
- $Nt = 82$ soit $20\,h30\,min$.

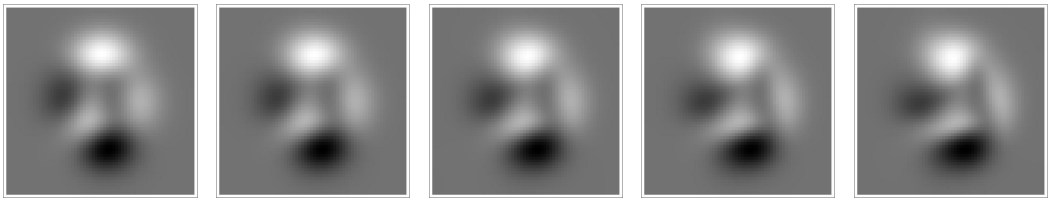
On obtient ainsi une séquence de vorticité $\xi(\mathbf{x}, t)$ et une séquence de pseudo-images $I_s(\mathbf{x}, t)$.

4.4.2 Simulation modèle réduit

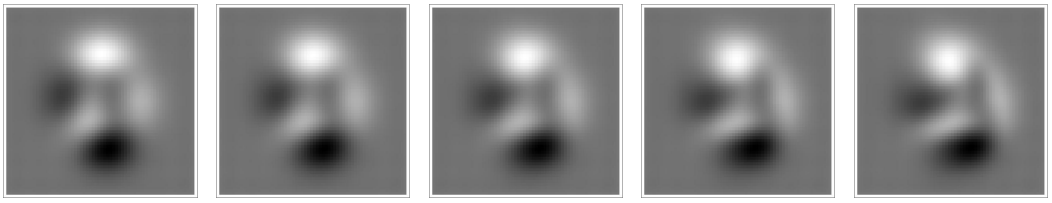
On choisit les dates $t_z = 15\,min, 5\,h15\,min, \ldots, 20\,h15\,min$ pour obtenir la séquence de 5 observations image $I = \{I^z\} = \{I(\mathbf{x}, t_z)\}_{z=1\ldots Z}$. Une POD est appliquée à la séquence I et fournit la base image Ψ_I^f (f pour full), dont on conserve les 3 premiers éléments pour former la base réduite Ψ_I qui est utilisée pour représenter les images. La base Ψ_ξ est restreinte à une sous-base de 256 éléments de la base sinus, correspondant aux basses fréquences. On conserve les éléments de la base qui correspondent aux indices $\{(i_1, i_2)\}_{i_1=1\ldots16, i_211=1\ldots16}$. Pour ne pas alourdir les notations, on conserve la notation Ψ_ξ pour cette sous-base. Le modèle réduit obtenu par projection de Galerkin du modèle AIMIDF sur les bases Ψ_ξ et Ψ_I est noté MR-AIMIDF_sinus-POD. Ceci afin d'expliquer que ξ est projetée sur la base de fonctions sinus et I_s sur une base POD.

Le modèle réduit est simulé sur 82 pas de temps, avec un $dt = 15\,min$, soit sur une fenêtre temporelle de 20 h30 m.

Les résultats de simulation sont présentés sur les figures 4.2 et 4.3, et quelques statistiques d'erreur sont fournies sur le tableau 4.1. Visuellement, les séquences de vorticité et d'images des modèles complet et réduit sont semblables. Les statistiques d'erreur sont très bonnes : les corrélations sont très proches de 1 et la NRMSE est de 0.23% pour les images et 1.01% pour la vorticité.

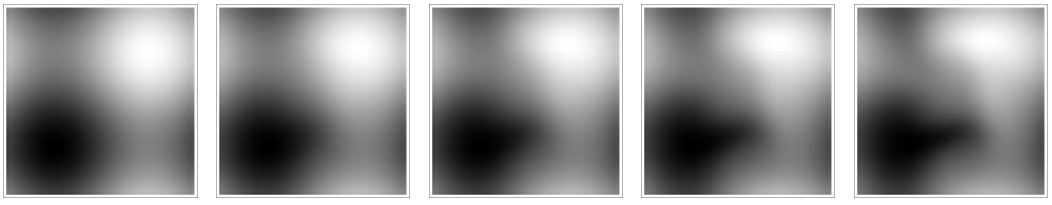


(a) $t = 15\,min$ (b) $t = 5\,h15\,min$ (c) $t = 10\,h15\,min$ (d) $t = 15\,h15\,min$ (e) $t = 20\,h15\,min$

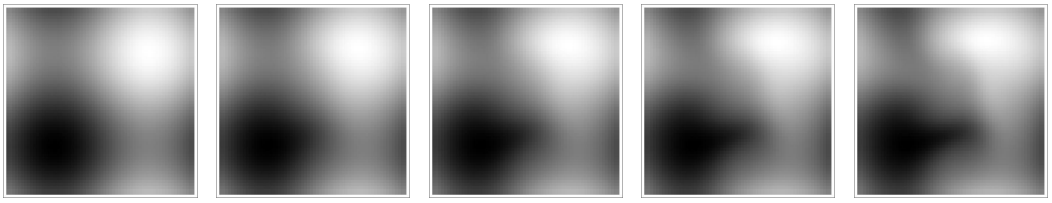


(f) $t = 15\,min$ (g) $t = 5\,h15\,min$ (h) $t = 10\,h15\,min$ (i) $t = 15\,h15\,min$ (j) $t = 20\,h15\,min$

FIGURE 4.2 – Simulation de vorticité ξ, du modèle complet (haut), comparé au modèle réduit (bas).



(a) $t = 15\,min$ (b) $t = 5\,h15\,min$ (c) $t = 10\,h15\,min$ (d) $t = 15\,h15\,min$ (e) $t = 20\,h15\,min$



(f) $t = 15\,min$ (g) $t = 5\,h15\,min$ (h) $t = 10\,h15\,min$ (i) $t = 15\,h15\,min$ (j) $t = 20\,h15\,min$

FIGURE 4.3 – Simulation du traceur I_s du modèle complet (haut), comparé au modèle réduit (bas).

Vorticité ξ	valeur min	moyenne	valeur max
Modèle complet	-0.0334	2e-6	0.0378
Modèle réduit	-0.0330	4.3e-5	0.0375
Image I_s	valeur min	moyenne	valeur max
Modèle complet	0	0.500	1.030
Modèle réduit	4e-6	0.500	1.021

	Vorticité	Image
RMSE	7.21	2.357e-3
NRMSE (%)	1.01	0.23
Corrélation	0.99	0.99

TABLE 4.1 – Statistiques d'erreur entre les séquences simulées par les modèles complet et réduit.

4.4.3 Conclusions

Les statistiques d'erreur, et en particulier la NRMSE, montrent que le modèle réduit 4.2.5, obtenu par projection de Galerkin sur les bases Ψ_ξ et Ψ_I, constitue une très bonne approximation du modèle complet 4.2.1.

4.5 Expérience jumelle d'assimilation

4.5.1 Assimilation dans le modèle réduit MR-AIMIDF_sinus-POD

La séquence synthétique utilisée dans cette expérience est celle décrite dans la sous-section 4.4. Les coefficients de projection $\{b^z\}_{z=1...Z}$ des images I^z sont assimilées dans le modèle réduit 4.2.5 sur la fenêtre temporelle $[0, 20\,\text{h}30\,\text{min}]$.

La méthode d'assimilation est décrite dans le chapitre 1. La valeur de background est 0 pour les coefficients a_i de la vorticité, et on utilise les coefficients de projection de la première image I^1 comme valeur de background sur b. La variable de contrôle est la condition initiale $\mathbf{X}_R(0) = \begin{pmatrix} a(0) \\ b(0) \end{pmatrix}$ du modèle réduit.

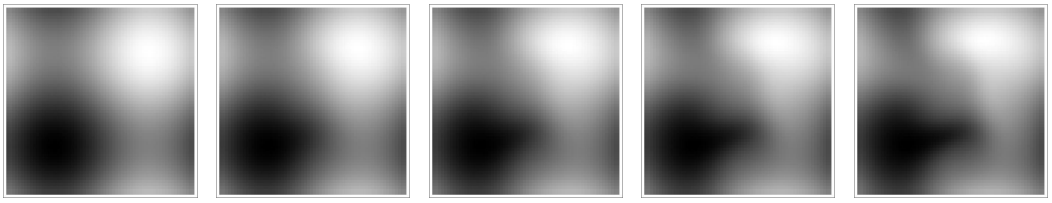
Les valeurs des paramètres de l'expérience d'assimilation sont les suivantes :

- $dt = 900\,\text{s} = 15\,\text{min}$,
- $Nt = 82$ soit $20\,\text{h}30\,\text{min}$,
- $B = 0.01$,
- $R = 0.01$,
- dates d'observation : 15 min, 5 h15 min, 10 h15 min, 15 h15 min, 20 h15 min

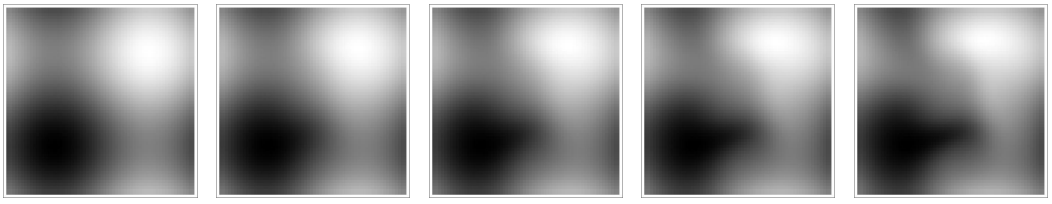
Les valeurs des cœfficients des observations et de l'ébauche sont de l'ordre de 1. On choisit donc des écarts-types d'erreur d'observation et d'ébauche de 0.1.

Résultats. Après assimilation, on obtient une séquence $\begin{pmatrix} a^a(t) \\ b^a(t) \end{pmatrix}$, à partir de laquelle on reconstruit l'image $I_s^a(\mathbf{x}, t)$, le champ de vorticité $\xi^a(\mathbf{x}, t)$, et le champ de mouvement $\mathbf{w}^a(\mathbf{x}, t)$, pour $t \in [0, 20\,\text{h}30\,\text{min}]$. Les résultats sont présentés sur les figures 4.4, 4.5 et 4.6 et quelques statistiques d'erreur sont fournies sur les tableaux 4.2 et 4.3.

La NRMSE pour la vorticité à $t = 0$ est de 7.97% et sur l'ensemble de la séquence la valeur est de 8.17%. Les images I_s^a obtenues sont très proches des observations I^z. La corrélation entre les deux séquences est de 0.99 et la NRMSE de 0.23%. La vorticité est sensiblement différente de la vérité terrain. On observe, sur les figures 4.5 et 4.6, que les grandes structures de la vorticité ξ^a et du mouvement sont présentes mais ne sont pas correctement positionnées.

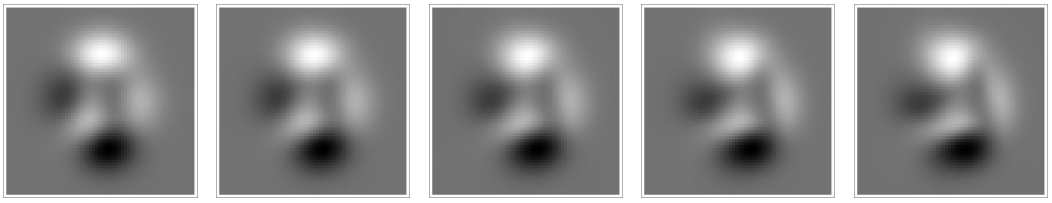


(a) $t = 15\,\text{min}$ (b) $t = 5\,\text{h}15\,\text{min}$ (c) $t = 10\,\text{h}15\,\text{min}$ (d) $t = 15\,\text{h}15\,\text{min}$ (e) $t = 20\,\text{h}15\,\text{min}$

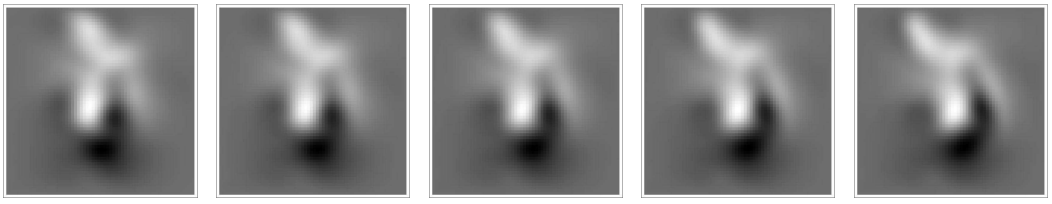


(f) $t = 15\,\text{min}$ (g) $t = 5\,\text{h}15\,\text{min}$ (h) $t = 10\,\text{h}15\,\text{min}$ (i) $t = 15\,\text{h}15\,\text{min}$ (j) $t = 20\,\text{h}15\,\text{min}$

FIGURE 4.4 – Vérité terrain I_s du modèle complet (haut), comparée aux résultats d'assimilation du modèle réduit I_s^a (bas).



(a) $t = 15\,\text{min}$ (b) $t = 5\,\text{h}15\,\text{min}$ (c) $t = 10\,\text{h}15\,\text{min}$ (d) $t = 15\,\text{h}15\,\text{min}$ (e) $t = 20\,\text{h}15\,\text{min}$

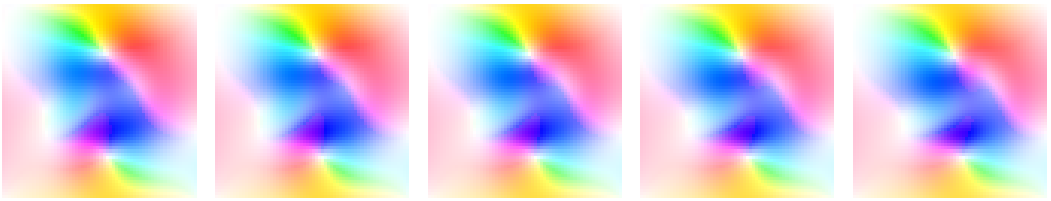


(f) $t = 15\,\text{min}$ (g) $t = 5\,\text{h}15\,\text{min}$ (h) $t = 10\,\text{h}15\,\text{min}$ (i) $t = 15\,\text{h}15\,\text{min}$ (j) $t = 20\,\text{h}15\,\text{min}$

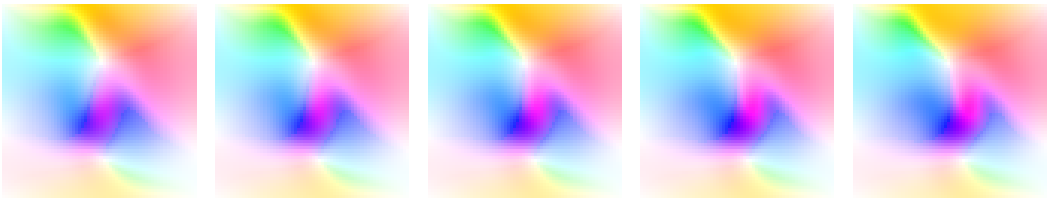
FIGURE 4.5 – Vérité terrain ξ du modèle complet (haut), comparée aux résultats d'assimilation du modèle réduit ξ^a (bas).

4.5.2 Conclusions

L'assimilation dans le modèle MR-AIMIDF_sinus-POD ne permet pas d'estimer de manière satisfaisante la séquence de vorticité. La perte d'information due à la projection



(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min



(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

FIGURE 4.6 – Vérité terrain $\mathbf{w}$ du modèle complet (haut), comparée aux résultats d'assimilation du modèle réduit w^a (bas).

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Analyse	-0.0210	1.36e-3	0.0279
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	0	0.500	1.00
Analyse	4e-6	0.500	1.00

	Vorticité	Image
RMSE	5.67	4.51e-4
NRMSE (%)	7.97	4.51e-2
Corrélation	0.79	0.99

TABLE 4.2 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation à t = 0.

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Résultat d'assimilation	-0.0330	4.3e-5	0.0375
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	4e-6	0.500	1.021
Résultat d'assimilation	4e-6	0.500	1.028

	Vorticité	Image
RMSE	5.8e-3	2.31e-3
NRMSE (%)	8.17	0.226
Corrélation	0.77	0.99

TABLE 4.3 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation sur toutes les dates.

des images sur une base POD à trois éléments ($L = 3$) est une explication possible de ce résultat. Une alternative au modèle MR-AIMIDF_sinus-POD est de ne pas projeter les observations et de conserver ainsi l'intégralité de l'information contenue dans la séquence I observée. Il s'agit alors de définir un nouveau modèle réduit dont le vecteur d'état est $\begin{pmatrix} a(t) \\ I_s(x,t) \end{pmatrix}$. Ce modèle est noté MR-AIMIDF_sinus afin d'expliquer que ξ est projetée sur la base sinus et que I_s n'est pas projetée. Cette solution est présentée dans la section 4.6.

4.6 Projection de l'équation du mouvement sur base de fonctions sinus

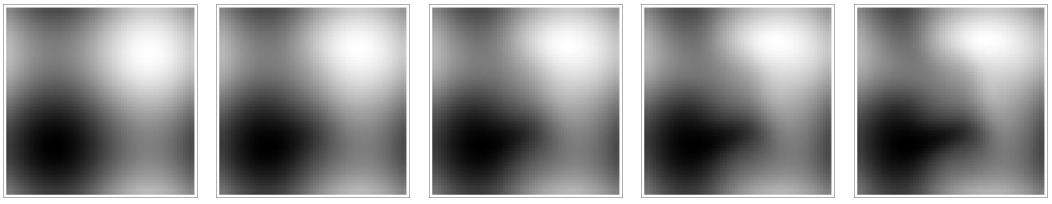
4.6.1 Introduction

On propose dans cette section de définir un nouveau modèle réduit, en ne projetant que l'équation de vorticité du modèle complet défini dans le système 4.2.1. Le vecteur d'état est $\mathbf{X}_R(x,t) = \begin{pmatrix} a(t) \\ I_s(x,t) \end{pmatrix}$, et les équations d'évolution temporelle, sous la contrainte d'un mouvement à divergence nulle, s'écrivent :

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T(t)B(k)a(t) = 0, \quad k = 1 \dots K. \\ \dfrac{\partial I_s}{\partial t}(x,t) + \left[\displaystyle\sum_{i=1}^{K} a_i(t) w(\psi_{\xi,i}) \right] \cdot \nabla I_s(x,t) = 0 \end{cases} \tag{4.6.1}$$

4.6.2 Assimilation dans le modèle réduit MR-AIMIDF_sinus

On réalise une expérience jumelle d'assimilation dans les mêmes conditions que celles de la section 4.5. Les observations pour ce modèle sont les images I^z de la figure 4.7. Ce sont les mêmes que celles utilisées dans l'expérience de la section 4.5. Elles sont assimilées dans le modèle réduit MR-AIMIDF_sinus sur la fenêtre $[0, 20\,\text{h}30\,\text{min}]$. La méthode d'assimilation est décrite au chapitre 2. La valeur de background est 0 pour les coefficients a_i de la vorticité et on utilise la première image I^1 pour le background de I_s.



(a) $t = 15\,\text{min}$ (b) $t = 5\,\text{h}15\,\text{min}$ (c) $t = 10\,\text{h}15\,\text{min}$ (d) $t = 15\,\text{h}15\,\text{min}$ (e) $t = 20\,\text{h}15\,\text{min}$

FIGURE 4.7 – Observations image I^z.

Dans ce modèle, l'erreur d'ébauche $\mathcal{E}_b(x)$ est définie sur le domaine spatial Ω et l'erreur d'observation $\mathcal{E}_O(x,t)$ sur le domaine $A = \Omega \times [0,T]$. On fait les hypothèses suivantes :

- $\mathcal{E}_b(x)$ et $\mathcal{E}_O(x,t)$ sont gausiennes de moyennes nulles,
- $\mathcal{E}_b(x)$ est décorrélée en espace,
- $\mathcal{E}_O(x,t)$ est décorrélée en temps et en espace.

$\mathcal{E}_b(\mathbf{x})$ et $\mathcal{E}_O(\mathbf{x},t)$ sont ainsi caractérisée par leurs matrices de covariance respectives $B(\mathbf{x})$ et $R(\mathbf{x},t)$.

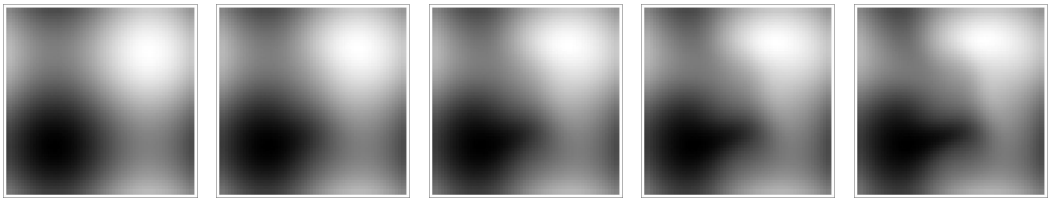
Les valeurs des paramètres de l'expérience d'assimilation sont les suivantes :

- $dx = dy = 1100\,m$.
- $dt = 900\,s$.
- $Nx = Ny = 64$ soit $70\,400\,m$,
- $Nt = 82$ soit $20\,h30\,min$,
- $B = 0.01$,
- $R = 0.01$,
- dates d'observation : 15 min, 5 h15 min, 10 h15 min, 15 h15 min, 20 h15 min.

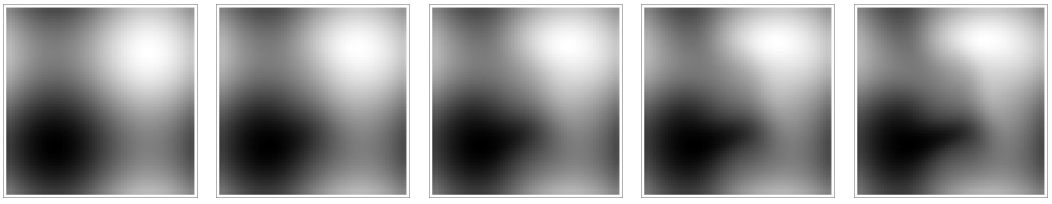
Les valeurs prises par les images I^z sont comprises entre 0 et 1, on choisit donc un écart-type pour les erreurs d'observation et de background égale à 0.1, à savoir 10% de leur ordre de grandeur. Les matrices de covariance $R(\mathbf{x},t)$ et $B(\mathbf{x})$ sont diagonales avec la valeur 10^{-2} sur la diagonale.

Résultats. On obtient une séquence $\begin{pmatrix} a^a(t) \\ I_s^a(\mathbf{x},t) \end{pmatrix}$, à partir de laquelle on reconstruit le champ de vorticité $\xi^a(\mathbf{x},t)$, et le mouvement $\mathbf{w}^a(\mathbf{x},t)$, pour $t \in [0, 20\,h30\,min]$. Les résultats sont présentés sur les figures 4.8, 4.9 et 4.10 et des statistiques d'erreurs sont fournies sur les tableaux 4.4 et 4.5.

Les images I_s^a obtenues sont très proches des observations I^z : la corrélation entre les deux séquences est de 0.99 et la NRMSE de 0.23%. Contrairement à l'expérience avec le modèle MR-AIMIDF_sinus-POD, la vorticité estimée est maintenant très proche de la vérité terrain. On observe sur les figures 4.9 et 4.10, que les séquences de vorticité et de mouvement sont visuellement semblables. La NRMSE est de 2.19% à $t = 0$ et de 2.22% sur l'ensemble de la séquence (à comparer avec celles de l'expérience précédente 7.97% à $t = 0$ et 8.17% sur l'ensemble de la séquence). La corrélation est de 0.98 à $t = 0$ et 0.99 sur l'ensemble de la séquence (0.79 et 0.77 dans l'expérience précédente).

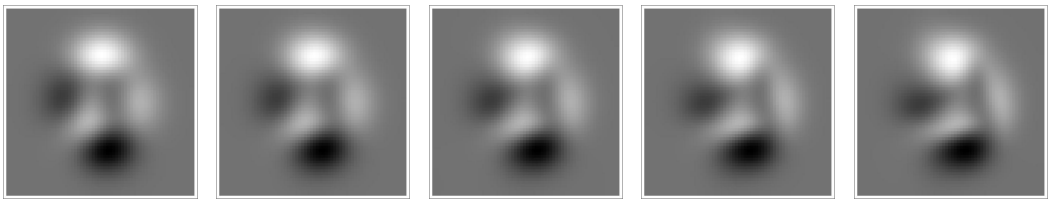


(a) $t = 15\,min$ (b) $t = 5\,h15\,min$ (c) $t = 10\,h15\,min$ (d) $t = 15\,h15\,min$ (e) $t = 20\,h15\,min$

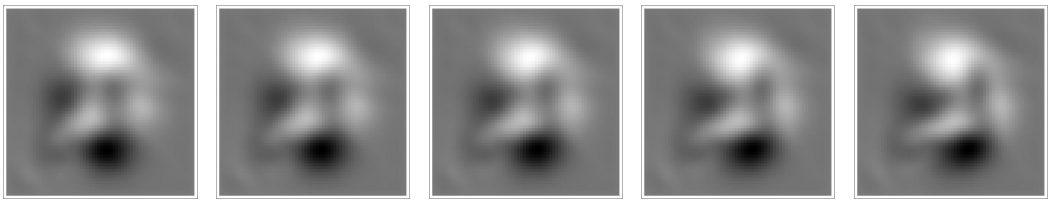


(f) $t = 15\,min$ (g) $t = 5\,h15\,min$ (h) $t = 10\,h15\,min$ (i) $t = 15\,h15\,min$ (j) $t = 20\,h15\,min$

FIGURE 4.8 – Vérité terrain image I_s du modèle complet (haut), comparée aux résultats d'assimilation du modèle réduit I_s^a (bas).

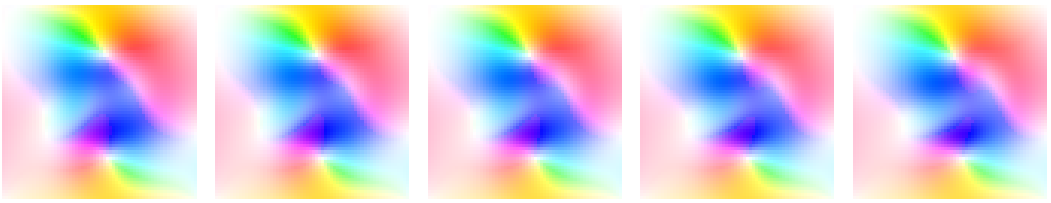


(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min

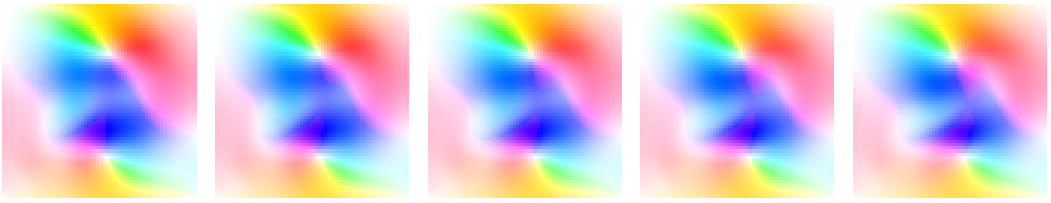


(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

FIGURE 4.9 – Vérité terrain ξ du modèle complet (haut), comparée aux résultats d'assimilation du modèle réduit ξ^a (bas).



(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min



(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

FIGURE 4.10 – Vérité terrain $\mathbf{w}$ du modèle complet (haut), comparée aux résultats d'assimilation du modèle réduit w^a (bas).

4.6.3 Conclusions

Ces résultats confirment les conclusions de la section 4.5 : l'assimilation directe des observations image est plus efficace que celle des trois coefficients POD b_j. La projection des observations sur une base POD entraîne une perte d'information trop importante pour permettre d'estimer correctement la vorticité, par assimilation de données.

Le modèle réduit MR-AIMIDF-sinus est plus efficace que le modèle réduit MR-AIMIDF_sinus-POD pour estimer le mouvement. Contrairement aux modèles présentés dans la première partie, le sous-espace de projection pour la vorticité est fixe et ne nécessite pas l'estimation d'un champ de mouvement initial $\mathbf{w}_0$. Les avantages de la représentation globale des champs de vorticité (discutés dans la section 4.3) sont conservés.

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0334	2e-6	0.0378
Résultat d'assimilation	-0.0320	6.8e-5	0.0344
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	0.00	0.500	1.00
Résultat d'assimilation	0.00	0.500	1.00

	Vorticité	Image
RMSE	1.56e-3	0.77e-3
NRMSE (%)	2.19	0.77
Corrélation	0.98	0.99

TABLE 4.4 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation à la date $t = 0$.

Vorticité ξ	valeur min	moyenne	valeur max
Vérité terrain	-0.0330	4.30e-5	0.0375
Résultat d'assimilation	-0.0322	6.90e-5	0.0346
Image I_s	valeur min	moyenne	valeur max
Vérité terrain	4.0e-6	0.500	1.02
Résultat d'assimilation	0.00	0.500	1.03

	Vorticité	Image
RMSE	1.49e-3	2.40e-3
NRMSE (%)	2.22	0.23
Corrélation	0.99	0.99

TABLE 4.5 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation sur toutes les dates.

La réduction proposée ne permet en revanche pas de réduire le temps de calcul, qui est comparable à celui du modèle complet AIMIDF.

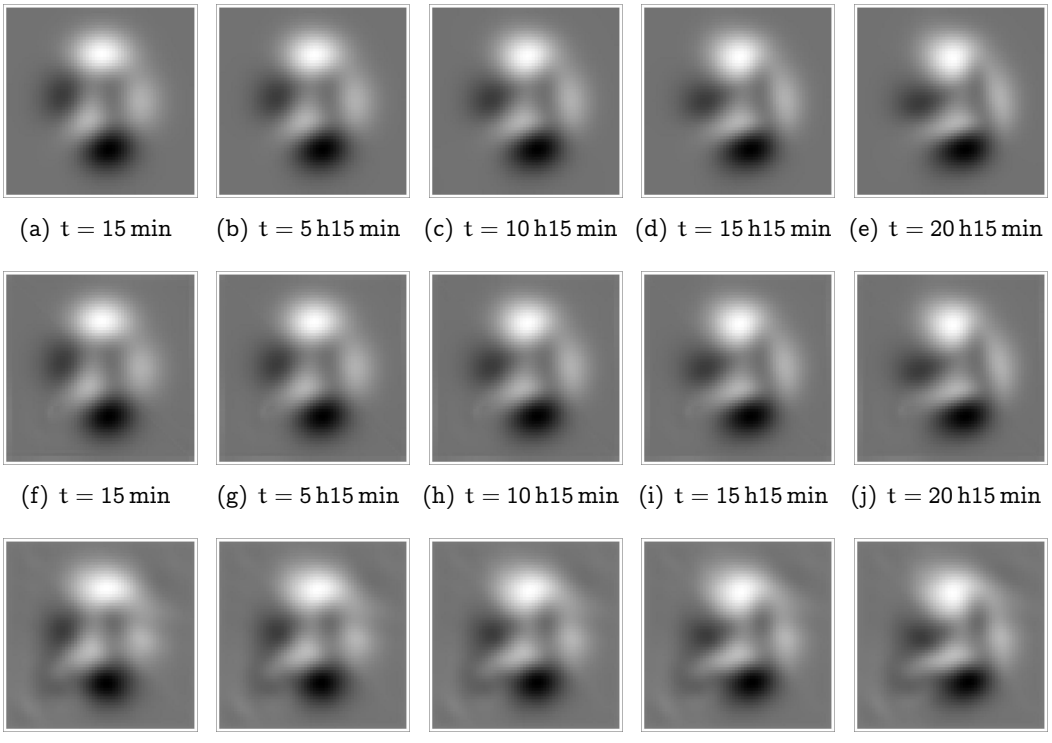
Dans la section suivante, on compare les performances de ce modèle pour l'estimation de la vorticité à celles du modèle complet AIMIDF.

4.7 Comparaison des résultats d'assimilation dans les modèles complet et réduit

4.7.1 Expérience d'assimilation sur des données synthétiques

Résultats. Dans cette section, on compare une expérience jumelle d'assimilation dans le modèle complet AIMIDF, présentée au chapitre 3, et une expérience cousine d'assimilation dans le modèle réduit MR-AIMIDF-sinus (dans la sous-section 4.6.2). Ces expériences ont été réalisées sur les observations synthétiques décrites dans la section 4.5. On fournit, sur les figures 4.11 et 4.12 et les tableaux 4.6 et 4.7, les résultats de chacune des méthodes. Les statistiques d'erreur du modèle complet sont un petit peu meilleures que celles du modèle réduit :

- la NRMSE est trois fois plus faible pour la vorticité,
- les corrélations sont du même ordre de grandeur.



(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min

(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

(k) t = 15 min (l) t = 5 h15 min (m) t = 10 h15 min (n) t = 15 h15 min (o) t = 20 h15 min

FIGURE 4.11 – Vérité terrain de la vorticité ξ (haut) – Résultats d'assimilation du modèle complet (milieu) – Résultats d'assimilation du modèle réduit (bas).

(a) t = 15 min (b) t = 5 h15 min (c) t = 10 h15 min (d) t = 15 h15 min (e) t = 20 h15 min

(f) t = 15 min (g) t = 5 h15 min (h) t = 10 h15 min (i) t = 15 h15 min (j) t = 20 h15 min

(k) t = 15 min (l) t = 5 h15 min (m) t = 10 h15 min (n) t = 15 h15 min (o) t = 20 h15 min

FIGURE 4.12 – Vérité terrain de l'image I_s (haut) – Résultats d'assimilation du modèle complet (milieu) – Résultats d'assimilation du modèle réduit (bas).

	Vorticité	Image
RMSE	5.23e-4	1.48e-4
NRMSE (%)	0.734	0.014
Corrélation	0.99	1

	Vorticité	Image
RMSE	1.49e-3	2.40e-3
NRMSE (%)	2.22	0.23
Corrélation	0.99	0.99

TABLE 4.6 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation des modèles complet (haut) et réduit (bas).

Méthode	Erreur angulaire (degrés) Moyenne	écart-type	Erreur relative en norme. Moyenne (%)
Horn[24]	15.26	9.65	45.75
Papadakis [34]	13.89	5.03	45.59
Corpetti[7]	12.54	9.49	38.70
Isambert [26]	10.61	6.92	34.84
Suter[42]	10.41	5.34	37.65
Huot [25]	9.91	7.38	31.45
Sun [41]	8.76	4.26	29.07
Modèle complet	0.18	0.10	0.65
Modèle réduit	1.53	1.10	0.06

TABLE 4.7 – Statistiques d'erreur entre les résultats de mouvement des modèles complet, réduit et quelques méthodes de l'état de l'art.

Conclusions. La représentation des champs de vorticité dans le modèle complet, plus précise que dans le modèle réduit, permet d'obtenir une estimation légèrement meilleure de la vorticité dans cette expérience. Cette différence est cependant négligeable en comparaison des différences avec les autres méthodes de l'état de l'art. Les résultats des modèles complet et réduit sont très proches.

Afin d'évaluer l'impact de la régularité de la base de fonctions sinus sur les images satellite bruitées, on réalise dans la sous-section suivante une comparaison des modèles MR-AIMIDF-sinus et AIMIDF sur des données satellite.

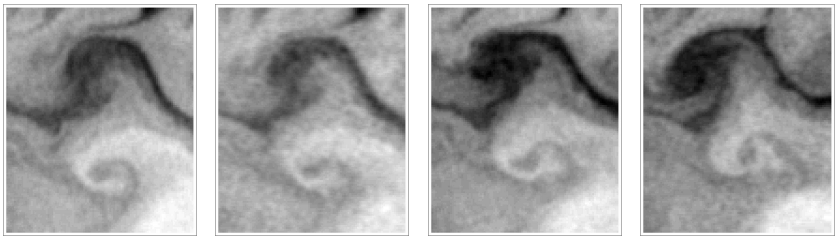
4.7.2 Assimilation d'images satellite

Introduction. Dans cette section, on réalise des expériences d'assimilation d'images satellite dans le modèle MR-AIMIDF-sinus. Les séquences utilisées sont des images de température de surface (SST, Sea Surface Temperature), acquises par le satellite NOAA/AVHRR sur la mer Noire. Elles ont une résolution spatiale d'environ 1 km. Dans la couche supérieure de la mer Noire, l'ordre de grandeur du mouvement horizontal est 30 cm/s pour les moyennes échelles, contre 10^{-4} cm/s pour le mouvement vertical [29]. L'hypothèse d'une divergence 2D nulle est donc approximativement vérifiée. Les modèles présentés dans les sections précédentes peuvent donc être appliqués à ce type de données.

Les 3 séquences sont présentées sur les figures 4.13, 4.14, 4.15 et décrites sur le tableau 4.8.

(a) t = 15 min (b) t = 8 h15 min (c) t = 12 h45 min (d) t = 21 h30 min (e) t = 24 h15 min

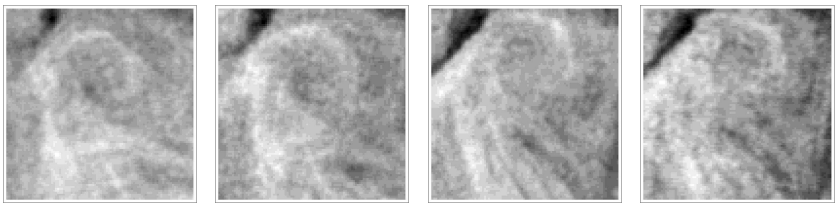
FIGURE 4.13 – Séquence SST numéro 1.



(a) t = 15 min (b) t = 5 h45 min (c) t = 15 h30 min (d) t = 29 h30 min

FIGURE 4.14 – Séquence SST numéro 2.

Estimation du mouvement par assimilation d'images SST. Les valeurs des paramètres communs aux trois expériences d'assimilation sont les suivantes :



(a) t = 15 min1 (b) t = 4 h15 min (c) t = 21 h45 min (d) t = 28 h

FIGURE 4.15 – Séquence SST numéro 3.

Indice de séquence	1	2	3
Date	27/07/2007	19/10/2007	28/08/2008
Domaine spatial	360 km × 325 km	500 km × 600 km	480 km × 490 km
Nx × Ny pixels	72 × 65	100 × 120	96 × 98
Fenêtre temporelle	24 h30 min	29 h45 min	28 h15 min
Nombre d'images	5	4	4
Dates (dt = 900 s = 15 min)			
Observation 1	15 min	15 min	15 min
Observation 2	8 h15 min	5 h45 min	4 h15 min
Observation 3	12 h45 min	15 h30 min	21 h45 min
Observation 4	21 h30 min	29 h30 min	28 h
Observation 5	24 h15 min	–	–

TABLE 4.8 – Caractéristiques des séquences satellite.

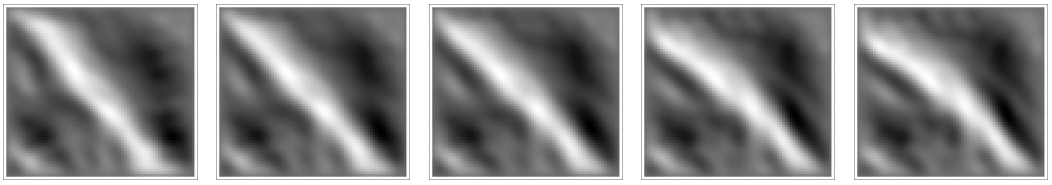
- dx = dy = 1100 m.
- dt = 900 s = 15 min.
- B = 1,
- R = 1,
- K = 256.

Les températures varient entre 22 et 25 degrés, on choisit donc des écarts-types d'erreur d'observation et de background de 1 degré.

Résultats. Les résultats sont présentés sur les figures 4.16 à 4.21, et les statistiques d'erreurs sur les images sont fournies sur le tableau 4.9. La NRMSE entre les séquences d'images et les résultats d'assimilation est comprise entre 6.27% (pour la séquence 1) et 15% (pour la séquence 3). Les corrélations sont supérieures à 0.86 pour les 2 premières séquences et à 0.5 pour la dernière. L'écart aux observations est faible pour les séquences 1 et 2 et toutefois un peu plus important pour la séquence 3.

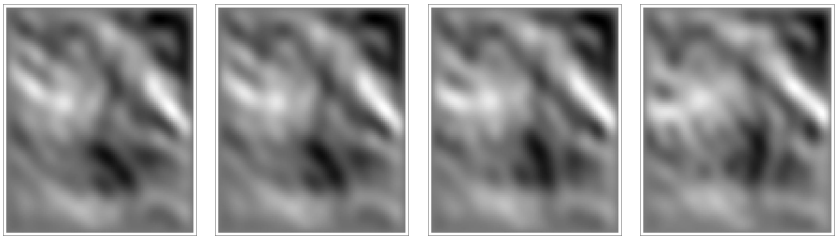
Afin d'évaluer la qualité des résultats obtenus, on compare les vorticités obtenues à celles estimées par le modèle complet dans la sous-section suivante.

Comparaison avec le modèle complet. On compare, dans cette sous-section, les résultats d'assimilation des modèles complet et réduit. Les paramètres des expériences d'assimilation dans le modèle complet (variances, Nx, Ny, Nt ...) ainsi que les séquences d'images à assimiler sont identiques pour les deux modèles.



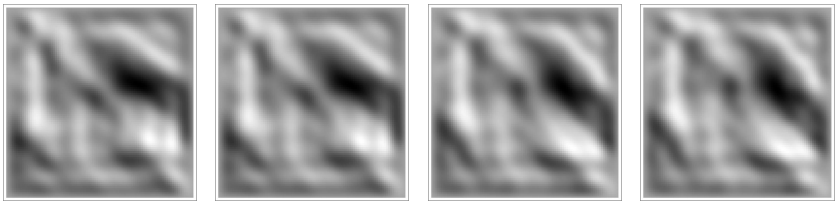
(a) t = 15 min (b) t = 8 h15 min (c) t = 12 h45 min (d) t = 21 h30 min (e) t = 24 h15 min

FIGURE 4.16 – Résultats de vorticité – Séquence numéro 1.



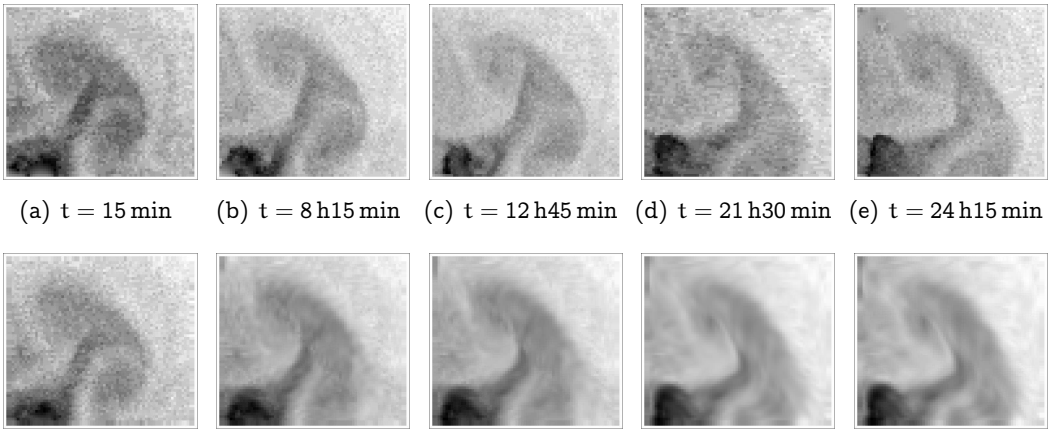
(a) t = 15 min (b) t = 5 h45 min (c) t = 15 h30 min (d) t = 29 h30 min

FIGURE 4.17 – Résultats de vorticité – Séquence numéro 2.



(a) t = 15 min (b) t = 4 h15 min (c) t = 21 h45 min (d) t = 28 h

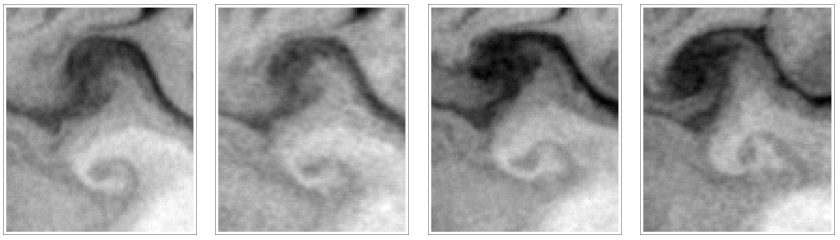
FIGURE 4.18 – Résultats de vorticité – Séquence numéro 3.



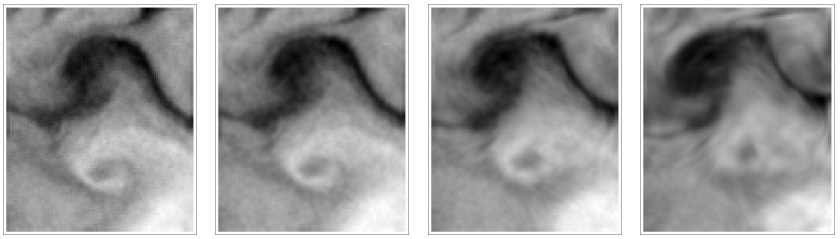
(a) t = 15 min (b) t = 8 h15 min (c) t = 12 h45 min (d) t = 21 h30 min (e) t = 24 h15 min

(f) t = 15 min (g) t = 8 h15 min (h) t = 12 h45 min (i) t = 21 h30 min (j) t = 24 h15 min

FIGURE 4.19 – Séquence numéro 1 – Observations (haut), comparées aux résultats d'assimilation (bas).

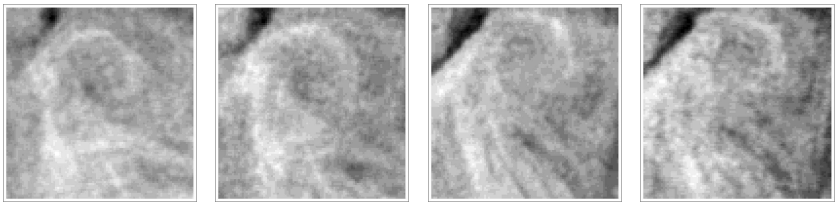


(a) t = 15 min (b) t = 5 h45 min (c) t = 15 h30 min (d) t = 29 h30 min

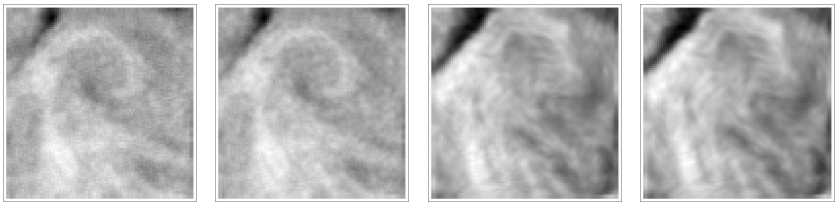


(e) t = 15 min (f) t = 5 h45 min (g) t = 15 h30 min (h) t = 29 h30 min

FIGURE 4.20 – Séquence numéro 2 – Observations (haut), comparées aux résultats d'assimilation (bas).



(a) t = 15 min (b) t = 4 h15 min (c) t = 21 h45 min (d) t = 28 h



(e) t = 15 min (f) t = 4 h15 min (g) t = 21 h45 min (h) t = 28 h

FIGURE 4.21 – Séquence numéro 3 – Observations (haut), comparées aux résultats d'assimilation (bas).

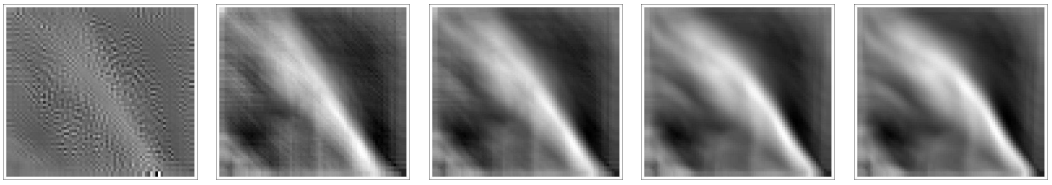
Indice de séquence	1	2	3
RMSE	4.33e-1	3.92e-1	3.91e-1
NRMSE (%)	6.27	8.34	15.04
Corrélation	0.86	0.88	0.50

TABLE 4.9 – Statistiques d'erreur entre les observations image et les résultats d'assimilation.

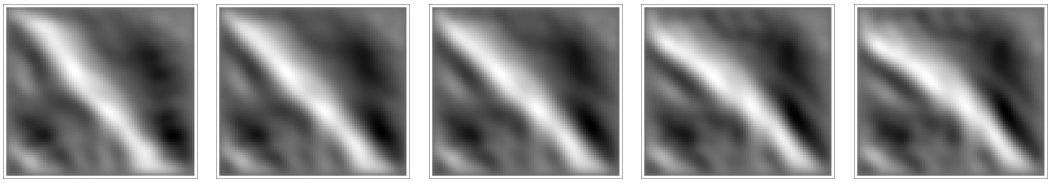
On présente sur les figures 4.22 à 4.24 les estimations de vorticité, obtenues avec les deux modèles, pour les séquences 1 à 3. Des statistiques entre les séquences estimées par les modèles complet et réduit sont fournies sur le tableau 4.10.

Résultats.

- la NRMSE entre les séquences de vorticité, estimées par les modèles complet et réduit, est inférieure à 5.8% et les corrélations supérieures à 72%,
- les séquences de vorticité obtenues par les deux modèles sont visuellement semblables pour les séquences 1 et 2. Des différences sont visibles sur la séquence 3, malgré des NRMSE et des corrélations comparables aux deux premières séquences,
- les résultats du modèle réduit sont nettement plus réguliers spatialement, en particulier sur les premières images ($t = 0$) des figures 4.22 à 4.24,
- les acquisitions satellite sont bruitées et ce bruit se retrouve dans la solution estimée par le modèle complet en raison de l'équation d'observation 1.2.3. Le modèle réduit, en revanche, contraint la vorticité dans un sous-espace régulier, ce qui a pour effet d'obtenir une solution sans bruit.



(a) $t = 15$ min (b) $t = 8$ h15 min (c) $t = 12$ h45 min (d) $t = 21$ h30 min (e) $t = 24$ h15 min



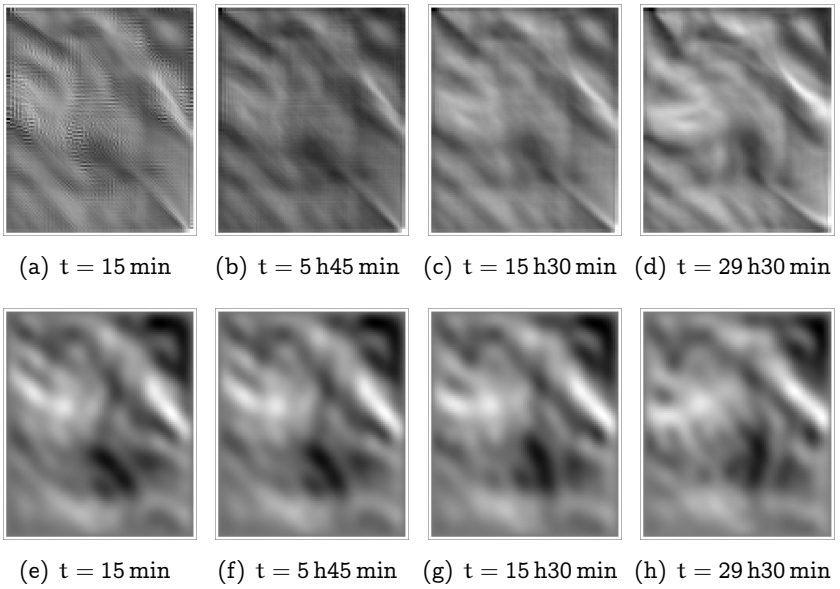
(f) $t = 15$ min (g) $t = 8$ h15 min (h) $t = 12$ h45 min (i) $t = 21$ h30 min (j) $t = 24$ h15 min

FIGURE 4.22 – Résultats de vorticité du modèle complet (haut) et du modèle réduit (bas). Séquence numéro 1.

Indice de séquence	1	2	3
RMSE	9.71e-3	1.17e-2	6.55e-3
NRMSE (%)	5.73	2.51	4.78
Corrélation	0.73	0.86	0.73

TABLE 4.10 – Statistiques d'erreur entre les résultats de vorticité des modèles complet et réduit.

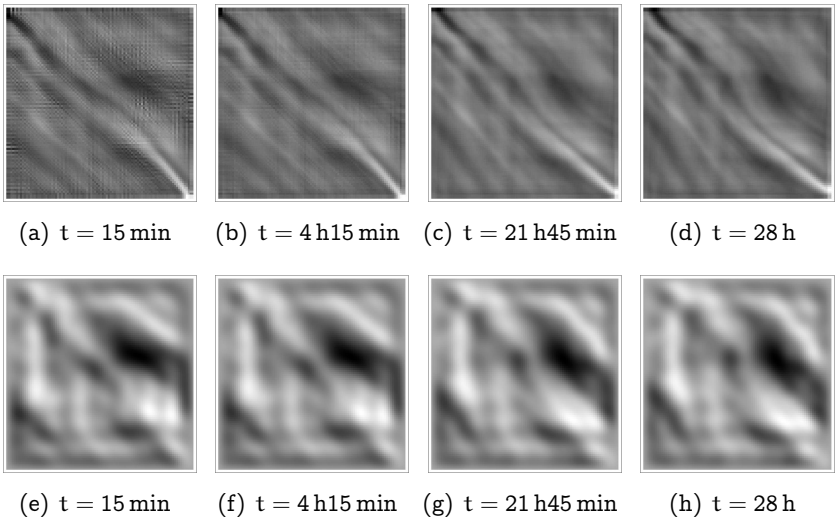
Conclusions. Les acquisitions satellite sont bruitées, cela se traduit par des discontinuités locales dans les images. Ces discontinuités n'ont pas de cohérence temporelle : elles apparaissent et disparaissent. L'algorithme d'assimilation dans le modèle complet calcule un champ de vorticité $\xi(\mathbf{x}, 0)$ cohérent avec les images et donc bruité spatialement.



(a) t = 15 min (b) t = 5 h45 min (c) t = 15 h30 min (d) t = 29 h30 min

(e) t = 15 min (f) t = 5 h45 min (g) t = 15 h30 min (h) t = 29 h30 min

FIGURE 4.23 – Résultats de vorticité du modèle complet (haut) et du modèle réduit (bas). Séquence numéro 2.



(a) t = 15 min (b) t = 4 h15 min (c) t = 21 h45 min (d) t = 28 h

(e) t = 15 min (f) t = 4 h15 min (g) t = 21 h45 min (h) t = 28 h

FIGURE 4.24 – Résultats de vorticité du modèle complet (haut) et du modèle réduit (bas). Séquence numéro 3.

Toutefois, l'intégration de l'équation d'évolution de ξ et I_s contribue au lissage de la solution. Ceci explique que les irrégularités présentes à $t = 0$ disparaissent au cours du temps. Le modèle réduit est la projection du complet sur une base de fonctions de sinus. Cette base étant spatialement régulière, le champ de vorticité estimé est contraint à la même régularité. Dans le cadre applicatif de l'estimation du mouvement à partir d'images SST, cette régularité est un avantage, les problèmes locaux liés au bruit d'acquisition des images sont lissés et n'impactent pas l'estimation du mouvement.

La principale limitation du modèle MR-AIMIDF-sinus est son temps de calcul important (comparable à celui du modèle complet) et proportionnel à la durée de l'intervalle temporel. Afin de mettre à profit les bons résultats du modèle MR-AIMIDF-sinus et de réduire le temps de calcul, on propose dans la section suivante de coupler ce modèle réduit avec le modèle MR-AIMIDF-POD-POD dans une expérience "fenêtre glissante".

4.8 Expérience fenêtre glissante

4.8.1 Introduction

Les résultats de l'expérience d'assimilation dans le modèle réduit MR-AIMIDF-sinus étant de bonne qualité, on se propose, dans cette section, de coupler ce modèle au modèle réduit MR-AIMIDF-POD-POD décrit dans le chapitre 3, au moyen d'une expérience "fenêtre glissante". Cela permet d'estimer le mouvement sur de longues séquences d'images en réduisant significativement le temps de calcul.

4.8.2 Description de l'expérience

Le principe de l'expérience "fenêtre glissante", décrit au chapitre 3, est d'estimer le mouvement sur de longues séquences d'images. La séquence est tout d'abord découpée en sous-séquences se recouvrant de moitié. Soit Wi_m la mième fenêtre ainsi obtenue. Les images de Wi_1 sont assimilées dans le modèle MR-AIMIDF-sinus. L'assimilation dans le modèle MR-AIMIDF-sinus est efficace mais coûteuse en temps de calcul et en mémoire. Les images des fenêtres Wi_m pour $m \neq 1$ sont assimilées dans les modèles réduits MR-AIMIDF-POD-POD, nommés $\mathbb{M}_R^m$. Comme cela a été expliqué au chapitre 3, l'avantage de cette technique est de réduire le temps de calcul, afin de permettre de réaliser l'assimilation sur de longues fenêtres temporelles.

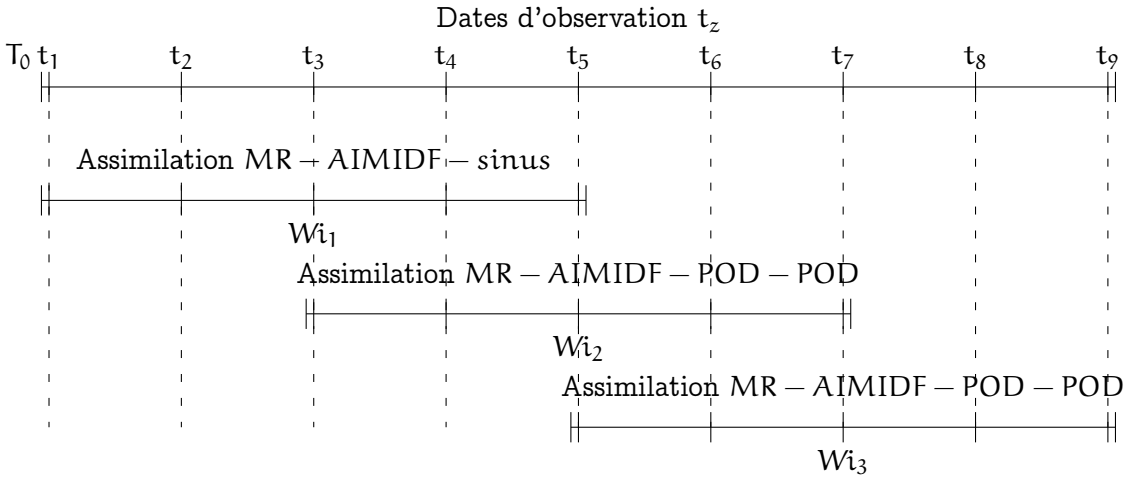


FIGURE 4.25 – Illustration de la méthode fenêtre glissante

4.8.3 Expérience jumelle par fenêtre glissante

Séquence synthétique. Le modèle complet 4.2.1 est simulé, à partir des conditions initiales (ξ_0, I_0) représentées sur la figure 4.1, pour obtenir une séquence $(\xi(\mathbf{x}, t), I_s(\mathbf{x}, t))$ sur la fenêtre temporelle $[T_0 = 0, T_F = 90\,\text{h}30\,\text{min}]$. Les 19 observations images I^z sont choisies aux dates $t = 15\,\text{min}, 5\,\text{h}15\,\text{min}, 10\,\text{h}15\,\text{min}, \ldots, 90\,\text{h}15\,\text{min}$.

Assimilation dans le modèle réduit MR-AIMIDF-sinus. Les 5 premières observations sont assimilées dans le modèle réduit MR-AIMIDF-sinus sur la fenêtre temporelle $Wi_1 = [0, 20\,\text{h}30\,\text{min}]$. Cette expérience ainsi que les résultats ont étés décrits dans la sous-section 4.6.2. Cette étape d'assimilation fournit un champ de vorticité $\xi^a(\mathbf{x}, t)$ et un champ de vitesse $\mathbf{w}^a(\mathbf{x}, t)$ pour $t \in [0, 20\,\text{h}30\,\text{min}]$.

Assimilation dans le modèle réduit MR-AIMIDF-POD-POD pour la $m^{\text{ième}}$ fenêtre. Comme décrit dans la sous-section 4.8.2, on assimile les coefficients de projection des images dans les modèles réduits $\mathbb{M}_R^m$ sur les fenêtres temporelles $Wi_2, Wi_3, \ldots$ définies de la façon suivante :

- $Wi_2 = [10\,\text{h}, 30\,\text{h}30\,\text{min}]$
- $Wi_3 = [20\,\text{h}, 40\,\text{h}30\,\text{min}]$
- $Wi_4 = [30\,\text{h}, 50\,\text{h}30\,\text{min}]$
- $Wi_5 = [40\,\text{h}, 60\,\text{h}30\,\text{min}]$
- $Wi_6 = [50\,\text{h}, 70\,\text{h}30\,\text{min}]$
- $Wi_7 = [60\,\text{h}, 80\,\text{h}30\,\text{min}]$

Les dates d'observation sont fournies pour chaque fenêtre dans le tableau 4.11.

fenêtre temporelle	Wi_2	Wi_3	Wi_4	Wi_5	Wi_6	Wi_7
observation 1	10 h15 min	20 h15 min	30 h15 min	40 h15 min	50 h15 min	60 h15 min
observation 2	15 h15 min	25 h15 min	35 h15 min	45 h15 min	55 h15 min	65 h15 min
observation 3	20 h15 min	30 h15 min	40 h15 min	50 h15 min	60 h15 min	70 h15 min
observation 4	25 h15 min	35 h15 min	45 h15 min	55 h15 min	65 h15 min	75 h15 min
observation 5	30 h15 min	40 h15 min	50 h15 min	60 h15 min	70 h15 min	80 h15 min

TABLE 4.11 – Expérience fenêtre glissante – Dates d'observation.

Calcul des bases Ψ_I^m et Ψ_ξ^m.

- La base réduite image Ψ_I^m est calculée en utilisant les observations image de la fenêtre Wi_m.
- Pour calculer la base Ψ_ξ^m ($m > 1$), on génère une séquence de vorticité en simulant le modèle complet sur la fenêtre Wi_m. On utilise comme condition initiale le champ de vorticité $\xi_0(\mathbf{x})$ obtenu comme résultat de l'assimilation sur la fenêtre Wi_{m-1}, comme discuté au chapitre 3.

Paramètres de l'assimilation dans le modèle réduit POD-POD. Sur chaque fenêtre Wi_m, les bases Ψ_I et Ψ_ξ sont composées de 3 éléments ($K = L = 3$). Les valeurs de paramètres suivants sont communes à toutes les fenêtres Wi_m :

- $dt = 900s$,
- $Nt = 82$ soit $20h30min$,
- $R = B = 10^{-2}$ (écart-type de 10^{-1}),
- Nombre d'observations : 5.

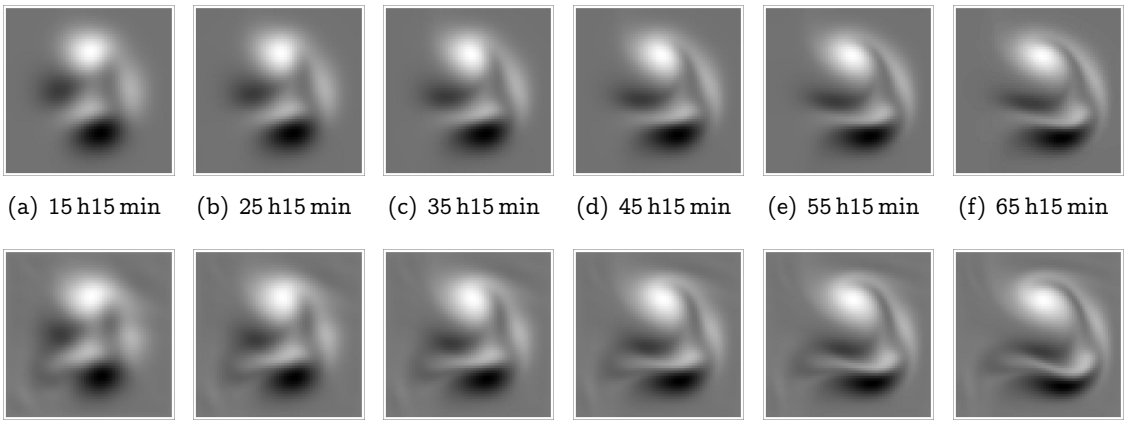
Les valeurs prises par les cœfficients observés sont de l'ordre de 1. On choisit des écarts-types d'erreur d'observation et de background de 0.1.

Résultats. Les résultats d'assimilation pour la vorticité et le traceur sont présentés sur les figures 4.26 et 4.27 et quelques statistiques d'erreur sont fournies sur le tableau 4.12. Les séquences estimées sont très proches de la vérité terrain sur les 6 fenêtres concernées par le modèle réduit POD-POD. Les séquences I_s^a estimées par assimilation sont semblables aux observations (corrélation > 0.99). La corrélation entre ξ et ξ^a est de 0.98 sur la fenêtre Wi_2 et décroît jusque 0.86 sur la fenêtre Wi_7. Par ailleurs, la NRMSE est de 2.26% sur Wi_2 et ne dépasse pas 6.97% sur l'ensemble des fenêtres Wi_m.

4.8.4 Conclusions

Cet exemple d'utilisation des modèles MR-AIMIDF-sinus et MR-AIMIDF-POD-POD montre comment combiner les qualités respectives de ces deux modèles pour l'estimation du mouvement :

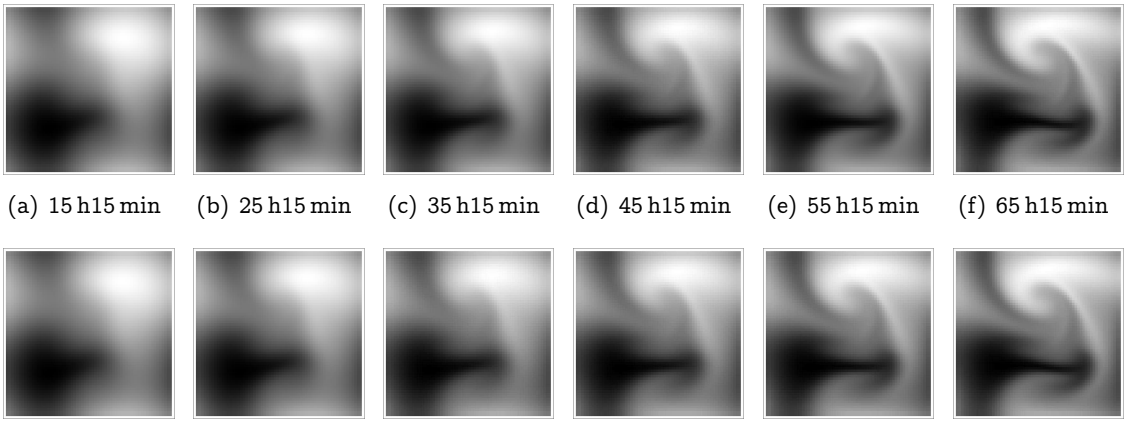
- comme expliqué dans la section 4.3, la représentation globale des champs évite les problèmes locaux d'instabilité de l'intégration temporelle,
- la régularité spatiale des fonctions sinus impose, à l'estimation de la vorticité, la même régularité : le bruit présent dans les images ne se retrouve pas dans la solution,
- le couplage des deux modèles permet de réduire significativement le temps de calcul nécessaire pour traiter de longues séquences d'images. L'estimation du mouvement



(a) 15 h15 min (b) 25 h15 min (c) 35 h15 min (d) 45 h15 min (e) 55 h15 min (f) 65 h15 min

(g) 15 h15 min (h) 25 h15 min (i) 35 h15 min (j) 45 h15 min (k) 55 h15 min (l) 65 h15 min

FIGURE 4.26 – Vérité terrain de la vorticité ξ, (haut), comparée au résultat d'assimilation ξ^a (bas).



(a) 15 h15 min (b) 25 h15 min (c) 35 h15 min (d) 45 h15 min (e) 55 h15 min (f) 65 h15 min

(g) 15 h15 min (h) 25 h15 min (i) 35 h15 min (j) 45 h15 min (k) 55 h15 min (l) 65 h15 min

FIGURE 4.27 – Vérité terrain de images I_s (haut), comparée au résultat d'assimilation I_s^a (bas).

fenêtre temporelle	2	3	4	5	6
RMSE Vorticité	1.60e-3	1.92e-3	2.36e-3	2.90e-3	3.63e-3
NRMSE Vorticité (%)	2.26	2.72	3.32	4.08	5.06
Corrélation Vorticité	0.98	0.98	0.96	0.95	0.92
RMSE Traceur	4.66e-3	3.72e-3	3.39e-3	5.75e-3	8.31e-3
NRMSE Traceur (%)	0.45	0.36	0.33	0.55	0.80
Corrélation Traceur	0.99	0.99	0.99	0.99	0.99
fenêtre temporelle	7				
RMSE Vorticité	5.02e-3				
NRMSE Vorticité (%)	6.97				
Corrélation Vorticité	0.86				
RMSE Traceur	1.28e-2				
NRMSE Traceur (%)	1.24				
Corrélation Traceur	0.99				

TABLE 4.12 – Statistiques d'erreur entre la vérité terrain et le résultat d'assimilation.

par assimilation dans le modèle MR-AIMIDF_sinus n'est mis en œuvre que sur la première fenêtre. Sur les fenêtres suivantes, le modèle MR-AIMIDF-POD-POD, et son vecteur d'état à 6 composantes, permet de significativement réduire la complexité de l'algorithme d'assimilation.

Chapitre 5

Estimation du mouvement sur bases d'ondelettes

5.1 Introduction

Nous avons présenté, dans le chapitre 4, l'estimation du mouvement dans un modèle réduit, obtenu par projection du modèle complet sur des bases fixes. Le principal intérêt de ce type de modèle est d'imposer les propriétés désirées à la solution, en choisissant une base mouvement dotée des mêmes propriétés. Dans le cas de la base de fonctions sinus, présentée au chapitre 4, la régularité spatiale et les valeurs nulles aux bords du domaine sont imposées au champ de vorticité estimé. Dans plusieurs cadres applicatifs, il est toutefois possible de vouloir disposer de propriétés diversifiées sur les champs de mouvement et de vouloir travailler sur des domaines complexes, ni carré, ni rectangulaire.

Plusieurs pistes de recherche ont été explorées afin d'assurer que le champ de mouvement, estimé à partir d'une paire d'images, soit doté des propriétés requises.
Dans [26], [42] et [43], la solution est recherchée sous la forme d'une combinaison linéaire de fonctions splines. Ces fonctions sont régulières spatialement et il est possible de tirer profit de leurs propriétés multi-échelles pour analyser le mouvement. Des bases d'ondelettes ont été utilisées dans [16] et [46]. Cette représentation des champs de mouvement permet également une étude naturelle des propriétés multi-échelles. Dans [9], des bases formées d'ondelettes à divergence nulle et d'ondelettes irrotationelles sont mises à profit pour estimer le mouvement dans des sous-espaces de faible dimension. Ces bases d'ondelettes ont étés introduites et utilisées dans [5] et [8] pour la simulation d'écoulements fluides.

Mais, toutes ces méthodes estiment un mouvement sur une paire d'images, sans imposer une cohérence temporelle sur la séquence complète. On propose donc, dans ce chapitre, de revenir à l'approche modèle réduit et de projeter un modèle dynamique dans un sous-espace, dont on choisira et détaillera par la suite les propriétés, afin de profiter à la fois de la cohérence temporelle du modèle dynamique et des propriétés des bases choisies.

Les bases élaborées dans cette thèse sont définies et décrites pour un domaine carré dans la section 5.2. La section 5.3 est alors consacrée à des expériences de projection et de simulation de ce modèle réduit. On présente, dans la section 5.4, les bases obtenues pour le domaine de la mer Noire ainsi qu'une expérience jumelle d'assimilation. Enfin, la section 5.5 conclut ce chapitre.

5.2 Bases d'ondelettes

On présente dans cette section deux types de bases d'ondelettes : dans la sous-section 5.2.1, une base d'ondelettes mouvement à divergence nulle et, dans la sous-section 5.2.2, une base scalaire pour les images.

5.2.1 Base d'ondelettes pour le mouvement à divergence nulle

On décide de s'intéresser à l'estimation de mouvement, sous la propriété de divergence nulle. Nous verrons, dans la suite du chapitre, que cette propriété est approximativement vérifiée par le mouvement de surface de la mer Noire : la composante verticale du mouvement est négligeable par rapport à la composante horizontale. On impose donc une divergence nulle comme contrainte à la base Φ^{div} du sous-espace recherché pour projeter les équations du mouvement, $\Phi^{div} = \{\phi_i^{div}(\mathbf{x})\}_{i=1\dots K}$ de taille K. Les éléments ϕ_i^{div} de la base sont obtenus en résolvant séquentiellement, pour $i = 1 \dots K$, les systèmes d'équations notés S_i^{div} :

$$S_i^{div} = \begin{cases} \phi_i^{div} = \min\limits_{\mathbf{f}\in L_2(\Omega)^2} \langle \nabla \mathbf{f}, \nabla \mathbf{f} \rangle, & (5.2.1) \\ \operatorname{div}\left(\phi_i^{div}(\mathbf{x})\right) = 0, \quad \forall \mathbf{x} \in \Omega, & (5.2.2) \\ \left\langle \phi_i^{div}, \phi_k^{div} \right\rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.2.3) \\ \phi_i^{div}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.2.4) \end{cases}$$

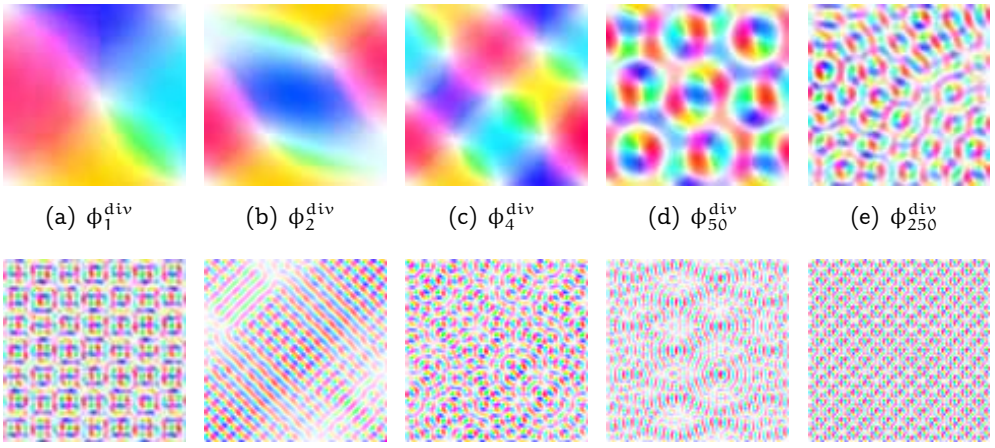
avec :
- $L_2(\Omega)^2$ l'espace des fonctions de carré intégrable de Ω dans $\mathbb{R}^2$ muni du produit scalaire $\langle \mathbf{f}, \mathbf{g} \rangle = \int_\Omega \mathbf{f}(\mathbf{x}) \cdot \mathbf{g}(\mathbf{x}) d\mathbf{x}$,
- $\partial\Omega$ le bord du domaine Ω,
- $\mathbf{n}(\mathbf{x})$ le vecteur normal au bord du domaine, défini pour $\mathbf{x} \in \partial\Omega$.

L'équation 5.2.1 est le critère à minimiser, à savoir l'intégrale de la norme du gradient. Les équations (5.2.2 à 5.2.4) représentent les contraintes imposées aux éléments de la base Φ^{div} :
- une divergence nulle sur tout le domaine Ω par l'équation 5.2.2,
- l'orthogonalité de la base par l'équation 5.2.3,
- et enfin une projection nulle de la vitesse sur la normale au bord du domaine $\partial\Omega$ pour l'équation 5.2.4,

Quelques-uns des éléments de la base Φ^{div}, obtenus pour un domaine Ω carré, sont représentés sur la figure 5.1. Les champs de vitesse sont représentés en pseudo-couleurs à l'aide de l'outil de représentation de la « Middlebury database [1] ». On observe que les premiers éléments ($i < 10$) de Φ^{div} représentent les grandes échelles du mouvement et que, plus i augmente, plus les échelles représentées sont fines.

1. http://vision.middlebury.edu/flow/

Ceci s'explique par le caractère séquentiel du calcul de la base. Lors du calcul du premier élément ($i = 1$) on minimise l'intégrale de la norme du gradient, équation 5.2.1, avec pour seules contraintes une divergence nulle, équation 5.2.2, et le respect des conditions aux bords, équation 5.2.4. L'équation 5.2.3 n'apporte aucune contrainte supplémentaire. La fonction la plus régulière sous ces contraintes fait donc naturellement apparaître les grandes échelles. Plus i augmente et plus l'équation 5.2.3 apporte de contraintes à l'ondelette calculée. Celle-ci est orthogonale aux précédentes ondelettes et fait donc apparaître des échelles de plus en plus fines.
Les bases ainsi construites sont ordonnées par échelle, le choix de K permet de plus ou moins filtrer les hautes fréquences du champ de mouvement.



(a) ϕ_1^{div} (b) ϕ_2^{div} (c) ϕ_4^{div} (d) ϕ_{50}^{div} (e) ϕ_{250}^{div}

(f) ϕ_{500}^{div} (g) ϕ_{750}^{div} (h) ϕ_{1000}^{div} (i) ϕ_{1500}^{div} (j) ϕ_{2000}^{div}

FIGURE 5.1 – Quelques éléments de la base d'ondelettes Φ^{div}, utilisée pour représenter le mouvement à divergence nulle.

5.2.2 Base image ondelettes

Dans cette section, on présente l'obtention d'une base image Ψ ayant des propriétés similaires aux bases mouvement :

- faible dimension,
- respect de conditions aux bords choisies,
- filtrage possible des hautes fréquences, effectué en choisissant le nombre d'éléments de la base.

On conserve les notations de la sous-section 5.2.1 pour le domaine spatial Ω ainsi que sa frontière $\partial\Omega$. On définit une base image (scalaire) $\Psi = \{\psi_i(\mathbf{x})\}_{i=1\dots L}$ de taille L. Les ψ_i sont obtenus en résolvant séquentiellement (pour $i = 1 \dots L$) les systèmes $S_i^{\mathbf{I}}$:

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i = \min\limits_{f \in L_2(\Omega)} \langle \nabla f, \nabla f \rangle , & (5.2.5) \\ \langle \psi_i, \psi_k \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.2.6) \\ \nabla \psi_i(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.2.7) \end{cases}$$

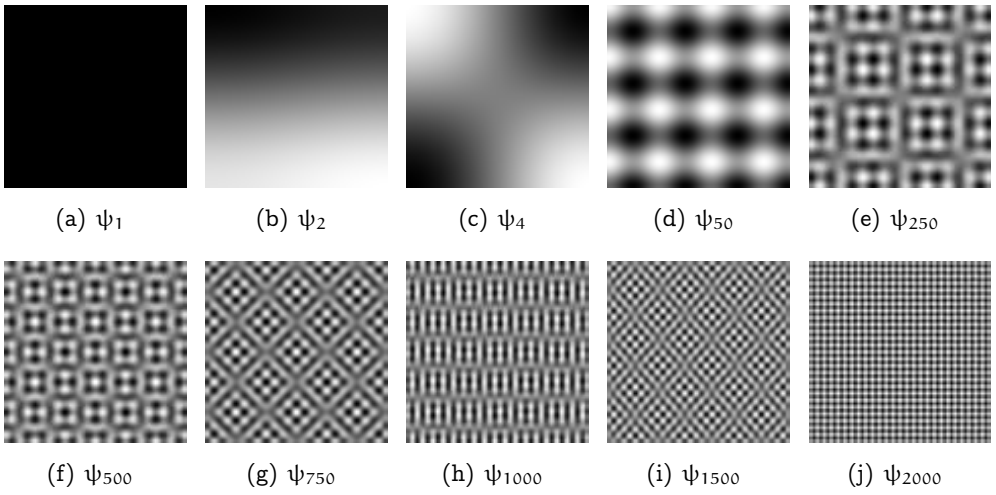
avec :

- $L_2(\Omega)$ l'espace des fonctions de carré intégrable de Ω dans $\mathbb{R}$ muni du produit scalaire $\langle f, g\rangle = \int_\Omega f(\mathbf{x}) \times g(\mathbf{x})d\mathbf{x}$,
- $\partial\Omega$ le bord du domaine Ω,
- $\mathbf{n}(\mathbf{x})$ le vecteur normal au bord du domaine, défini pour $\mathbf{x} \in \partial\Omega$.

L'équation 5.2.5 est le critère à minimiser, à savoir l'intégrale de la norme du gradient. Les équations 5.2.6 et 5.2.7 représentent les contraintes imposées aux éléments de la base Ψ :

- l'orthogonalité de la base pour l'équation 5.2.6,
- et enfin, un gradient nul sur la normale aux bords du domaine $\partial\Omega$ pour l'équation 5.2.7.

Quelques-uns des éléments de la base Ψ, obtenus pour un domaine carré Ω, sont représentés sur la figure 5.2. Tout comme pour la base d'ondelettes calculée pour représenter le mouvement, la base obtenue est également ordonnée par échelles décroissantes : les faibles valeur de i représentent les grandes échelles du mouvement et les grandes valeurs de i les plus petites échelles.



(a) ψ_1 (b) ψ_2 (c) ψ_4 (d) ψ_{50} (e) ψ_{250}

(f) ψ_{500} (g) ψ_{750} (h) ψ_{1000} (i) ψ_{1500} (j) ψ_{2000}

FIGURE 5.2 – Quelques éléments de la base ondelettes Ψ, utilisée pour représenter les images.

5.3 Expériences synthétiques

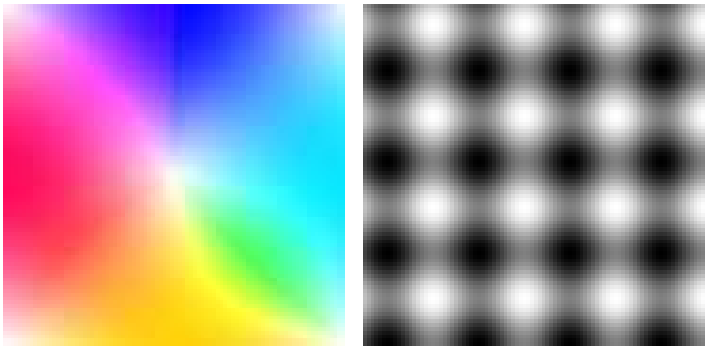
On présente, dans la sous-section 5.3.1, les résultats de la projection de champs de mouvement sur Φ^{div} et d'images sur Ψ afin de quantifier la représentativité des bases Φ^{div} et Ψ. Une expérience de simulation du modèle réduit, obtenu par projection du modèle complet AIMIDF sur les bases d'ondelettes Φ^{div} et Ψ, est présentée dans la sous-section 5.3.2.

5.3.1 Advection du mouvement et des images à divergence nulle : projection sur Φ^{div} et Ψ

Simulation du modèle complet. Une simulation du modèle complet AIMIDF est effectuée à partir des initialisations $\mathbf{w}_0(\mathbf{x})$ et $I_0(\mathbf{x})$, représentées sur la figure 5.3. Les séquences $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ obtenues serviront dans l'expérience de cette sous-section et dans la suivante. Les paramètres de la simulation sont les suivants :

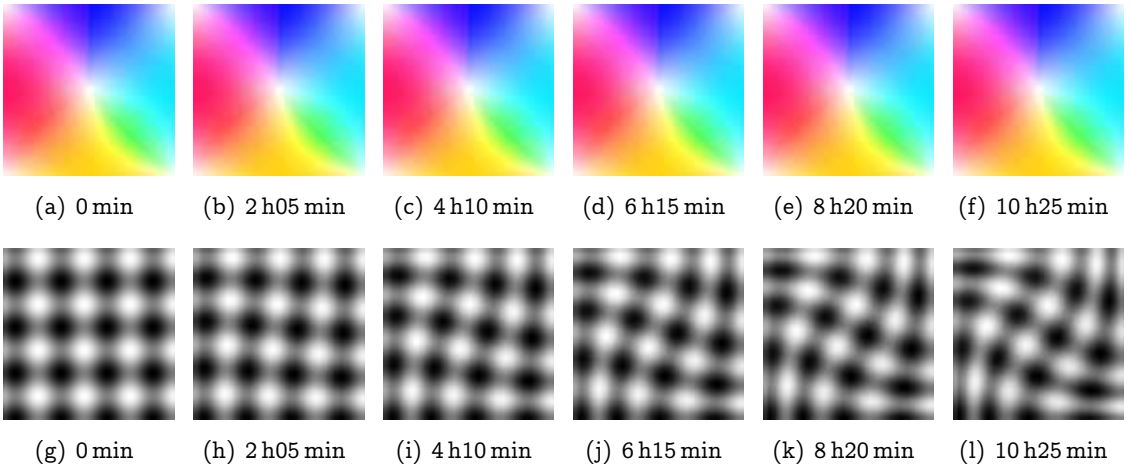
- $dx = dy = 1100\,m$.
- $dt = 25\,min$.
- $Nx = Ny = 90$ soit $99\,km$,
- $Nt = 28$ soit environ $12\,h$,

Les résultats $(\mathbf{w}(\mathbf{x},t), I_s(\mathbf{x},t))$ de la simulation sont présentés sur la figure 5.4 aux dates $t = 0\,min, 2\,h05\,min, \ldots, 10\,h25\,min$.



(a) $\mathbf{w}_0(\mathbf{x})$ (b) $I_0(\mathbf{x})$

FIGURE 5.3 – Conditions initiales de la simulation du modèle réduit AIMIDF.



(a) 0 min (b) 2 h05 min (c) 4 h10 min (d) 6 h15 min (e) 8 h20 min (f) 10 h25 min

(g) 0 min (h) 2 h05 min (i) 4 h10 min (j) 6 h15 min (k) 8 h20 min (l) 10 h25 min

FIGURE 5.4 – Simulation du modèle complet $\mathbf{w}(\mathbf{x},t)$ (haut) et $I_s(\mathbf{x},t)$ (bas).

Expérience de projection sur les bases Φ^{div} et Ψ. On réalise des expériences de projection de champs de mouvement et de pseudo-images sur les bases Φ^{div} et Ψ définies dans la section 5.2.

Les séquences $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ simulées par le modèle complet AIMIDF sont respectivement projetées sur les bases Φ^{div} et Ψ pour différentes valeurs de K et L. Quelques statistiques d'erreur sont fournies sur les tableaux 5.1 et 5.2. Les statistiques d'erreur de projection de $\mathbf{w}(\mathbf{x},t)$ sur Φ sont très faibles : la RMSE et la NRMSE décroissent respectivement de 9.29×10^{-3} et 1.03% pour $K=32$ à 6.82×10^{-3} et 0.76% pour $K=2000$. La projection sur Φ^{div} de la séquence de mouvement est donc une très bonne approximation de la simulation du modèle complet, y compris pour de faibles valeur de K. Les statistiques d'erreur de projection de $I_s(\mathbf{x},t)$ sur Ψ sont elles aussi très faibles : la RMSE et la NRMSE décroissent respectivement de 1.16 et 21.96% pour $L=32$ à 0.146 et 2.769% pour $L=4096$. Pour $L>768$ la NRMSE est inférieure à 0.79%. La projection sur Ψ de la séquence de mouvement est donc une bonne approximation de la simulation du modèle complet pour $L>768$. Les séquences $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ sont très régulières spatialement ce qui explique naturellement que les erreurs de projection sont faibles pour des $K=32$ et $L=768$.

Les bases Φ^{div} et Ψ permettent de bien représenter les champs de mouvement et les pseudo-images. Afin de valider ces bases pour l'estimation du mouvement par assimilation de données, il est aussi nécessaire que le modèle réduit, obtenu par projection de Galerkin du modèle complet AIMIDF sur Φ^{div} et Ψ, soit une bonne approximation du modèle complet. Une comparaison quantitative des simulations des modèles complet et réduit est présentée dans la sous-section 5.3.2.

	Projection de $\mathbf{w}(\mathbf{x},t)$							
K	32	64	128	256	512	768	1024	1280
RMSE (e-3)	9.29	8.56	7.93	7.58	7.17	7.02	6.93	6.89
NRMSE (%)	1.03	0.95	0.88	0.84	0.79	0.78	0.77	0.76
K	1536	1792	2000					
RMSE	6.86	6.83	6.82					
NRMSE (%)	0.76	0.76	0.76					

TABLE 5.1 – Statistiques d'erreur de projection de $\mathbf{w}(\mathbf{x},t)$ sur Φ en fonction de la taille K de la base.

	Projection de $I_s(\mathbf{x})$							
L	32	64	128	256	512	768	1024	4096
RMSE (e-1)	11.6	0.49	1.99e-1	1.53e-1	1.47e-1	1.465e-1	1.462e-1	1.460
NRMSE (%)	21.96	9.23	3.79	2.91	2.79	2.779	2.774	2.769

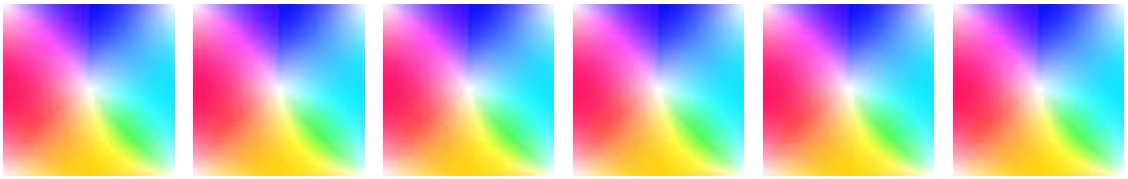
TABLE 5.2 – Statistiques d'erreur de projection de $I_s(\mathbf{x})$ sur Ψ en fonction de la taille L de la base.

5.3.2 Simulation du modèle réduit

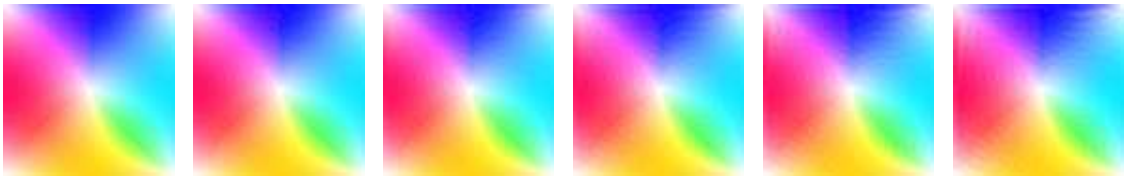
La projection du modèle AIMIDF sur Φ^{div} et Ψ est notée MR-AIMIDF-ON, afin d'indiquer qu'il s'agit d'un modèle réduit (MR) d'advection des images et du mouvement, à divergence nulle (AIMIDF), sur des bases d'ondelettes (ON). Le modèle réduit MR-AIMIDF-ON est simulé sur 28 pas de temps avec un $dt = 1500\,s$ $(25\,min)$, sur une

fenêtre temporelle d'environ 12 h. Les conditions initiales de la simulation sont les projections, sur les bases Φ^{div} et Ψ, des conditions initiales utilisées pour le modèle complet, et représentées sur la figure 5.3.

Les résultats de la simulation du modèle réduit sont présentés sur les figures 5.5 et 5.6, après reconstruction du mouvement et des images. Quelques statistiques d'erreur sont fournies sur le tableau 5.3. Visuellement, les séquences de mouvement et les pseudo-images des modèles complet et réduit sont semblables. Les statistiques d'erreur pour le mouvement sont bonnes : la RMSE croit de de 6.96×10^{-3} à 10.48×10^{-3} et la NRMSE de 0.77% à 1.16%. Pour les pseudo-images, les statistiques d'erreur sont légèrement moins bonnes : la RMSE croit de de 0.12 à 0.31 et la NRMSE de 2.30% à 5.90%. Le modèle réduit MR-AIMIDF-ON est malgré tout une excellente approximation du modèle complet AIMIDF.

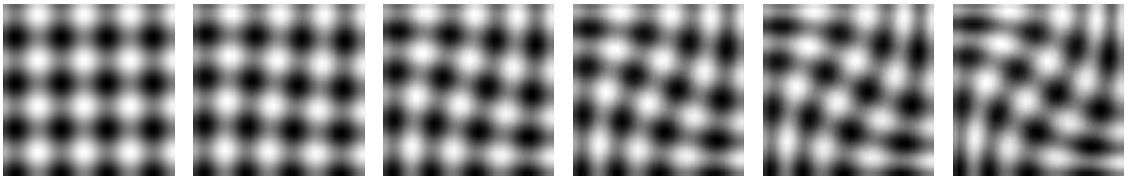


(a) 0 min (b) 2 h05 min (c) 4 h10 min (d) 6 h15 min (e) 8 h20 min (f) 10 h25 min

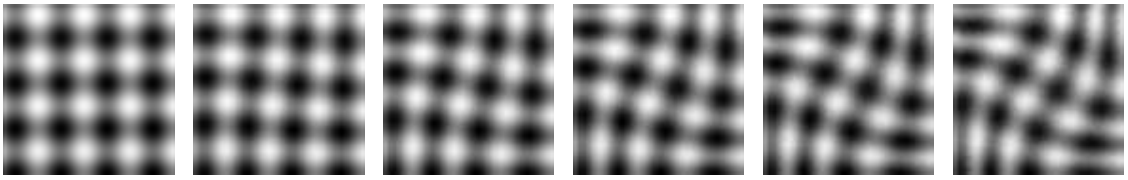


(g) 0 min (h) 2 h05 min (i) 4 h10 min (j) 6 h15 min (k) 8 h20 min (l) 10 h25 min

FIGURE 5.5 – Simulation du mouvement $\mathbf{w}$ par le modèle complet (haut), comparée à celle du modèle réduit (bas).



(a) 0 min (b) 2 h05 min (c) 4 h10 min (d) 6 h15 min (e) 8 h20 min (f) 10 h25 min



(g) 0 min (h) 2 h05 min (i) 4 h10 min (j) 6 h15 min (k) 8 h20 min (l) 10 h25 min

FIGURE 5.6 – Simulation du traceur I_s par le modèle complet (haut), comparée à celle du modèle réduit (bas).

	$\mathbf{w}(\mathbf{x},t)$					
t	0	5	10	15	20	25
RMSE (e-3)	6.96	7.24	8.00	8.82	9.63	10.48
NRMSE (%)	0.77	0.80	0.88	0.98	1.07	1.16
	$I_s(\mathbf{x},t)$					
t	0	5	10	15	20	25
RMSE	0.12	0.11	0.13	0.18	0.24	0.31
NRMSE (%)	2.30	2.16	2.62	3.50	4.62	5.90

TABLE 5.3 – Statistiques d'erreur entre les séquences de mouvement $\mathbf{w}(\mathbf{x},t)$ et de pseudo-image $I_s(\mathbf{x},t)$, simulées par les modèles complet et réduit.

5.4 Estimation du mouvement de surface de la mer Noire

Dans cette section, on présente une méthode d'assimilation d'images dans un modèle réduit. Le but est d'estimer le mouvement sur le bassin de la mer Noire en représentant les champs de mouvement et les images par leurs projections sur les bases fixes, décrites dans la section 5.2. Un avantage supplémentaire de ces bases, que nous n'avons pas encore abordé, est qu'elles peuvent être définies sur des domaines Ω à géométrie complexe. On génère donc les bases pour le domaine Ω de la mer Noire, visualisé sur la figure 5.7 :

- une base ondelettes mouvement à divergence nulle Φ^{div},
- et une base ondelettes scalaire Ψ pour les images.

On rappelle que ces bases vérifient les propriétés suivantes :

- faible dimension : choix de K pour la taille de la base mouvement et de L pour la base image,
- faible erreur de projection des champs de mouvement et des images,
- respect des conditions aux bords, choisies par l'utilisateur,
- filtrage des hautes fréquences, déterminé par l'utilisateur,
- régularité spatiale.

On présente les bases obtenues dans la sous-section 5.4.1. La sous-section 5.4.2 est consacrée à la présentation des données qui seront utilisées dans cette section. Des expériences numériques sont présentées dans les sous-sections 5.4.3 et 5.4.4.

FIGURE 5.7 – Domaine spatial Ω : Bassin de la mer Noire.

5.4.1 Bases d'ondelettes pour la mer Noire

Une description complète de l'obtention des bases ondelettes est faite dans la section 5.2. On se contente dans cette sous-section de rappeler les systèmes d'équations à résoudre et de présenter les résultats obtenus sur le domaine de la mer Noire.

Base mouvement à divergence nulle. On impose une divergence nulle comme contrainte à la base Φ^{div}, $\Phi^{div} = \{\phi_i^{div}(\mathbf{x})\}_{i=1\dots K}$ de taille K. Les éléments ϕ_i^{div} sont obtenus en résolvant séquentiellement les systèmes S_i^{div} pour $i = 1 \dots K$:

$$S_i^{div} = \begin{cases} \phi_i^{div} = \min\limits_{\mathbf{f}\in L_2(\Omega)^2} \langle \nabla \mathbf{f}, \nabla \mathbf{f} \rangle, & (5.4.1) \\ \operatorname{div}\left(\phi_i^{div}(\mathbf{x})\right) = 0, \quad \forall \mathbf{x} \in \Omega, & (5.4.2) \\ \left\langle \phi_i^{div}, \phi_k^{div} \right\rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.4.3) \\ \phi_i^{div}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.4.4) \end{cases}$$

Quelques uns des éléments de la base Φ^{div} ainsi obtenue sont représentés sur les figures 5.8 à 5.13. On observe comme dans la section 5.2 que les faibles valeurs de i représentent les grandes échelles et que les plus grandes valeurs de i représentent les petites échelles.

FIGURE 5.8 – ϕ_1^{div}.

FIGURE 5.9 – ϕ_4^{div}.

FIGURE 5.10 – ϕ_{128}^{div}.

FIGURE 5.11 – ϕ_{256}^{div}.

FIGURE 5.12 – ϕ_{512}^{div}.

FIGURE 5.13 – ϕ_{1000}^{div}.

Base image. Les ψ_i sont obtenus en résolvant séquentiellement les systèmes $S_i^{\mathbf{I}}$ pour $(i = 1 \dots L)$:

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i = \min\limits_{f \in L_2(\Omega)} \langle \nabla f, \nabla f \rangle , & (5.4.5) \\ \langle \psi_i, \psi_k \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.4.6) \\ \nabla \psi_i(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.4.7) \end{cases}$$

Quelques uns des éléments de la base Ψ ainsi obtenue sont représentés sur les figures 5.14 à 5.19. Tout comme pour la base mouvement Φ, les ψ_i sont ordonnés par échelles spatiales décroissantes et le choix de L permet de filtrer les hautes fréquences.

FIGURE 5.14 – ψ_1.

FIGURE 5.15 – ψ_4.

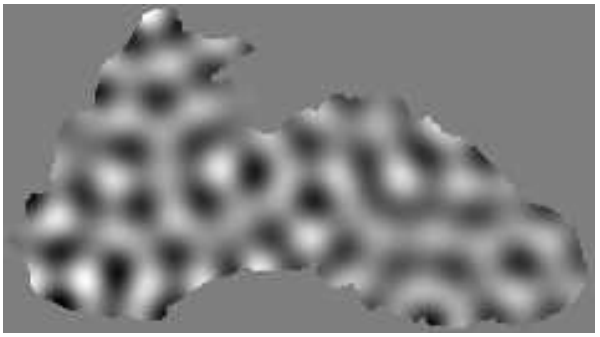


FIGURE 5.16 – ψ_{128}.

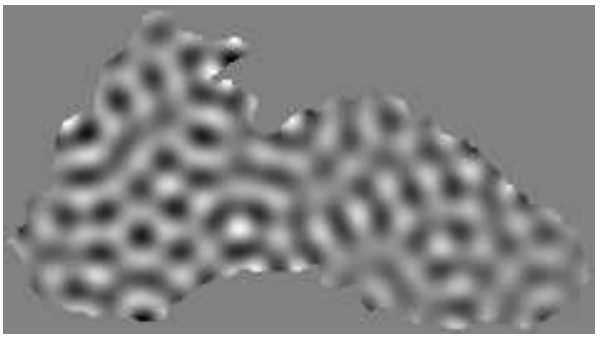


FIGURE 5.17 – ψ_{256}.

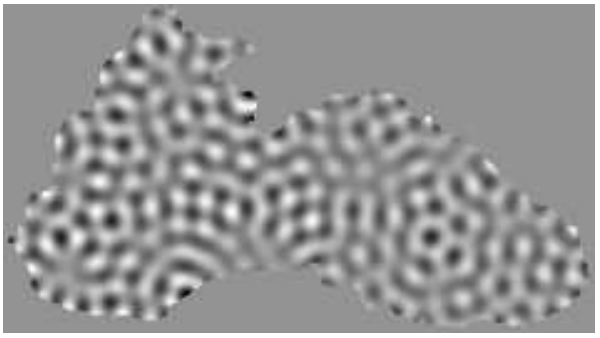


FIGURE 5.18 – ψ_{512}.

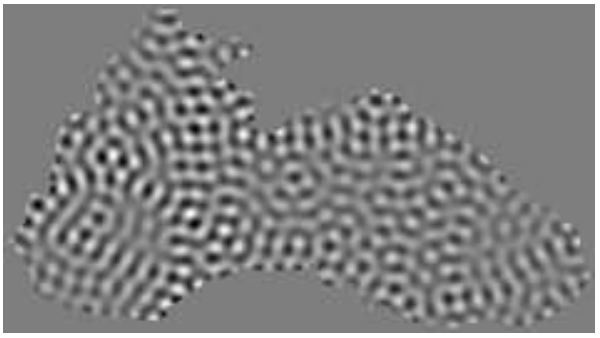


FIGURE 5.19 – ψ_{1000}.

5.4.2 Données océanographiques utilisées pour l'étude

Projet MyOcean. Dans ce chapitre, des données du projet MyOcean [2] sont utilisées pour réaliser l'estimation du mouvement par un modèle réduit sur des bases ondelettes. L'objectif principal du projet MyOcean est de fournir et d'exploiter une surveillance des océans rigoureuse, fiable et durable ainsi qu'un système de prévision pour le service "GMES Marine" (Global Monitoring for Environment and Security – Marine). Le projet concerne tous les types d'applications marines :

- la sécurité maritime,
- les ressources marines,
- l'environnement marin et côtier,
- le climat,
- la prévision saisonnière et la météo.

Différent types de données sont disponibles sur le portail MyOcean, des observations, analyses et prévisions de :

- vents,
- biogéochimie,
- courants marins,
- hauteur d'eau,
- température,
- salinité.

Ces données sont disponibles sur tous les océans ainsi que sur les mers Baltiques, Méditerranéennes et Noire.

Données utilisées. Il s'agit d'analyses de mouvement, de température de surface et de hauteur d'eau. Le modèle océanographique utilisé pour réaliser ces analyses est celui du "MyOcean Black sea monitoring and forecasting centre". C'est un modèle 3D sur une grille régulière. Sa résolution horizontale est d'environ 5 km (11/180 degrés de longitude et 2/45 degrés de latitude). Il est composé de 38 niveaux verticaux irréguliers : 2.5 m pour le niveau de surface et 100 m pour le niveau le plus profond. Les analyses sont produites en assimilant, dans ce modèle, des données de hauteur d'eau et de température de surface d'une résolution d'environ 6 km par la méthode d'interpolation optimale. Les analyses sont des champs instantanés toutes les 3 heures.

Une documentation complète de ces données est disponible en ligne [3] Le produit utilisé est référencé dans la documentation sous le nom de "BLACKSEA-ANALYSIS-FORECAST-PHYS-007-001-a".

5.4.3 Comparaison des simulations complète et réduite

Dans cette sous-section, on compare les simulations des modèles complet et réduit : la simulation du modèle complet AIMIDF est comparé à sa projection de Galerkin sur les bases Φ^{div} et Ψ.

2. http://www.myocean.eu/
3. http://catalogue.myocean.eu.org/static/resources/myocean/pum/MY

La projection du modèle complet sur Φ^{div} et Ψ est notée MR-AIMIDF-MN afin d'indiquer qu'il s'agit d'un modèle réduit (MR) d'advection à divergence nulle (AIMIDF) sur des bases adaptées au bassin de la mer Noire (MN).

On rappelle les équations du modèle complet à divergence nulle AIMIDF, qui concernent l'évolution de la vorticité ξ et de la pseudo-image I_s (système 5.4.8).

$$\begin{cases} \dfrac{\partial \xi}{\partial t}(\mathbf{x},t) + \nabla\left(\mathbf{w}(\xi)\xi\right)(\mathbf{x},t) = 0 \\ \dfrac{\partial I_s}{\partial t}(\mathbf{x},t) + \nabla\left(\mathbf{w}(\xi)I_s\right)(\mathbf{x},t) = 0 \end{cases} \tag{5.4.8}$$

Afin de définir le modèle réduit, on approxime $\mathbf{w}(\mathbf{x},t)$ et $I_s(\mathbf{x},t)$ par leurs projections sur les bases Φ^{div} et Ψ (système 5.4.9).

$$\begin{cases} \mathbf{w}(\mathbf{x},t) & \approx & \displaystyle\sum_{i=1}^{K} a_i(t)\phi_i^{div}(\mathbf{x}) \\ I_s(\mathbf{x},t) & \approx & \displaystyle\sum_{j=1}^{L} b_j(t)\psi_j(\mathbf{x}) \end{cases} \tag{5.4.9}$$

Les équations d'évolution du modèle réduit, calculées au chapitre 2, sont rappelées par le système 5.4.10 :

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T B(k) a = 0, \quad k = 1 \dots K \\ \dfrac{db_l}{dt}(t) + a^T G(l) b = 0, \quad l = 1 \dots L \end{cases} \tag{5.4.10}$$

avec :

$$\begin{cases} B(k)_{i,j} = \dfrac{\left\langle (\phi_i^{div} \cdot \nabla)\phi_j^{div}, \phi_k^{div} \right\rangle}{\left\langle \phi_k^{div}, \phi_k^{div} \right\rangle} \\ G(l)_{i,j} = \dfrac{\left\langle \phi_i^{div} \cdot \nabla\psi_j, \psi_l \right\rangle}{\langle \psi_l, \psi_l \rangle} \end{cases} \tag{5.4.11}$$

Simulation du modèle complet. Le modèle complet AIMIDF (équation 5.4.8) est simulé à partir des conditions initiales $(\mathbf{w}_0, I_0)$ représentées sur la figure 5.20. I_0 est la moyenne temporelle des températures de surface MyOcean du 8 Juillet 2011. $\mathbf{w}_0$ doit être à divergence nulle tout en restant physiquement acceptable. On choisit donc, pour $\mathbf{w}_0$, la valeur de champ correspondant à l'équilibre géostrophique de la hauteur d'eau analysée, issue du modèle MyOcean.

Le code de simulation du modèle complet ne gérant que des conditions aux bords pour des domaines spatiaux rectangulaires, il est nécessaire d'effectuer la simulation sur

un domaine étendu. Le domaine spatial sur lequel est faite la simulation est donc le plus petit rectangle englobant le domaine de la mer Noire Ω. La question se pose donc de savoir quelles valeurs affecter aux zones en dehors du domaine de la mer Noire. Pour le mouvement $\mathbf{w}_0$, on affecte la valeur 0. Pour la pseudo-image I_0, on applique un algorithme de diffusion à partir des valeurs sur les bords de Ω.

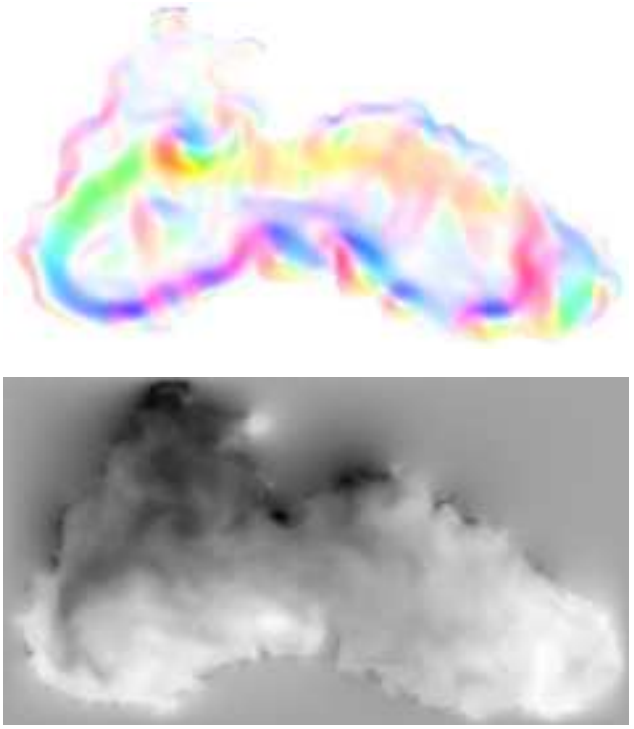


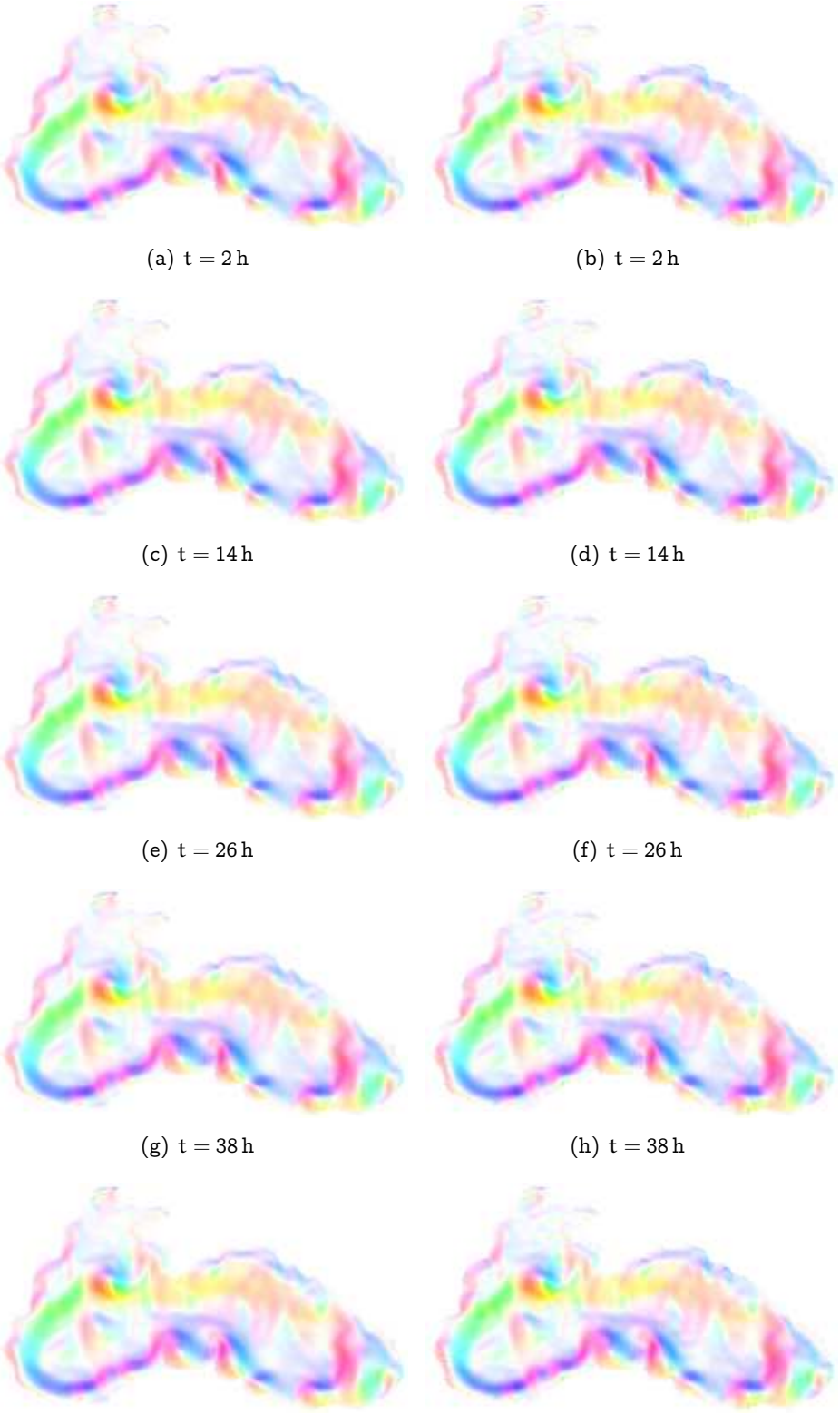
FIGURE 5.20 – Conditions initiales de la simulation – $\mathbf{w}_0$ (haut) et I_0 (bas).

Les paramètres de la simulation sont les suivants :
- $dx = dy = 5\,\text{km}$,
- $dt = 2\,\text{h}$,
- $Nx = 238 \quad (1190\,\text{km})$,
- $Ny = 132 \quad (660\,\text{km})$,
- $Nt = 26 \quad (52\,\text{h})$.

On obtient ainsi une séquence $(\mathbf{w}(\mathbf{x}, t), I_s(\mathbf{x}, t))$.

Modèle réduit. Le modèle réduit est simulé avec des bases Φ et Ψ de taille respective $K = 1000$ et $L = 6000$, sur 26 pas de temps, avec un $dt = 2\,\text{h}$, soit 52 h.

Résultats. Les résultats de simulation sont présentés sur les figures 5.21 et 5.22. Des statistiques d'erreur, fournies sur le tableau 5.4 montrent que le modèle réduit, obtenu par projection de Galerkin sur les bases Φ^{div} et Ψ, de tailles respectives $K = 1000$ et $L = 6000$, constitue une excellente approximation du modèle complet AIMIDF sur la fenêtre temporelle de 52 h. La corrélation entre les séquences de mouvement simulées par les modèles complet et réduit est de 8.2×10^{-3}, la NRMSE est de 1.51%. La corrélation entre les séquences de pseudo-images simulées par les modèles complet et réduit est de 0.99, la NRMSE est de 0.98%.



(a) t = 2 h (b) t = 2 h

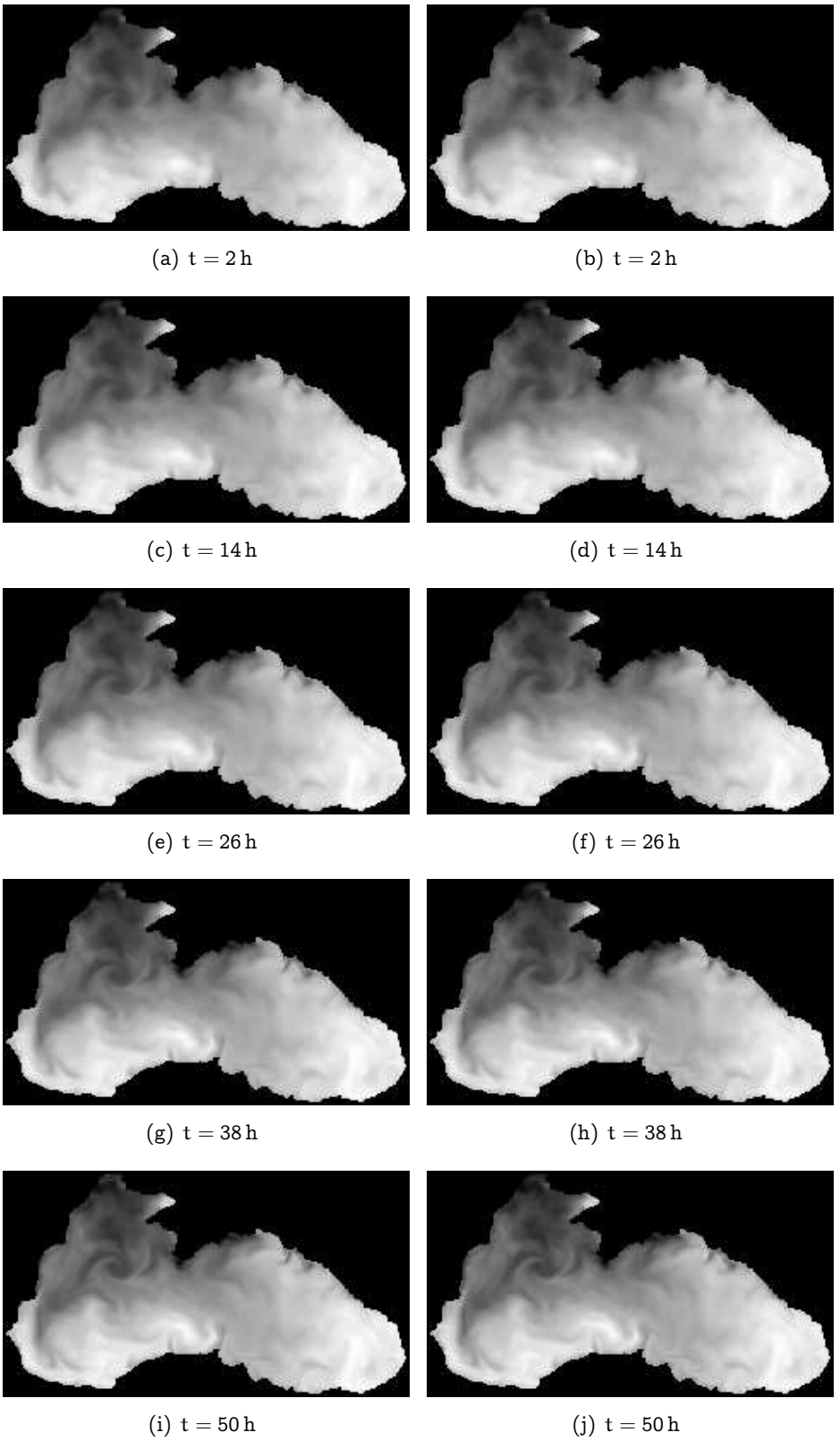
(c) t = 14 h (d) t = 14 h

(e) t = 26 h (f) t = 26 h

(g) t = 38 h (h) t = 38 h

(i) t = 50 h (j) t = 50 h

FIGURE 5.21 – Simulation du mouvement des modèles complet (gauche) et réduit (droite).



(a) t = 2 h (b) t = 2 h

(c) t = 14 h (d) t = 14 h

(e) t = 26 h (f) t = 26 h

(g) t = 38 h (h) t = 38 h

(i) t = 50 h (j) t = 50 h

FIGURE 5.22 – Simulation des pseudo-images des modèles complet (gauche) et réduit (droite).

	Mouvement $\mathbf{w}$			Pseudo-image I_s		
	valeur min	moyenne	valeur max	valeur min	moyenne	valeur max
Modèle complet	-0.331	0.76e-3	0.278	19.5	22.8	24.7
Modèle réduit	-0.330	1.01e-3	0.287	19.5	22.9	24.8

	$\mathbf{w}$	I_s
RMSE	0.0092	0.0516
NRMSE (%)	1.51	0.98
Corrélation	0.99	0.99

TABLE 5.4 – Statistiques d'erreur entre les simulations des modèles complet et réduit.

5.4.4 Assimilation dans le modèle réduit à divergence nulle

On présente, dans cette section, une expérience jumelle d'assimilation. La séquence synthétique utilisée dans cette expérience est décrite dans la sous-section 5.4.3. Les dates d'observation choisies sont les suivantes : $t_z = 2\,h,\ 8\,h,\ 14\,h,\ \ldots,\ 50\,h$ pour former la séquence d'images $I = \{I^z\}_{z=1,\ldots,9}$.

Assimilation dans le modèle réduit. Les coefficients de projection $\{b^z\}_{z=1\ldots Z}$ des images I sont assimilées dans le modèle réduit MR-AIMI-MN, sur la fenêtre temporelle $[0, 27]$. La méthode d'assimilation est décrite dans le chapitre 1. La valeur de background est 0 pour les coefficients a_i du mouvement et on utilise les coefficients de projection de la première image I^1 comme valeur de background sur b. La variable de contrôle est la condition initiale $\mathbf{X}_R(0) = \begin{pmatrix} a(0)^T & b(0)^T \end{pmatrix}^T$ du modèle réduit.

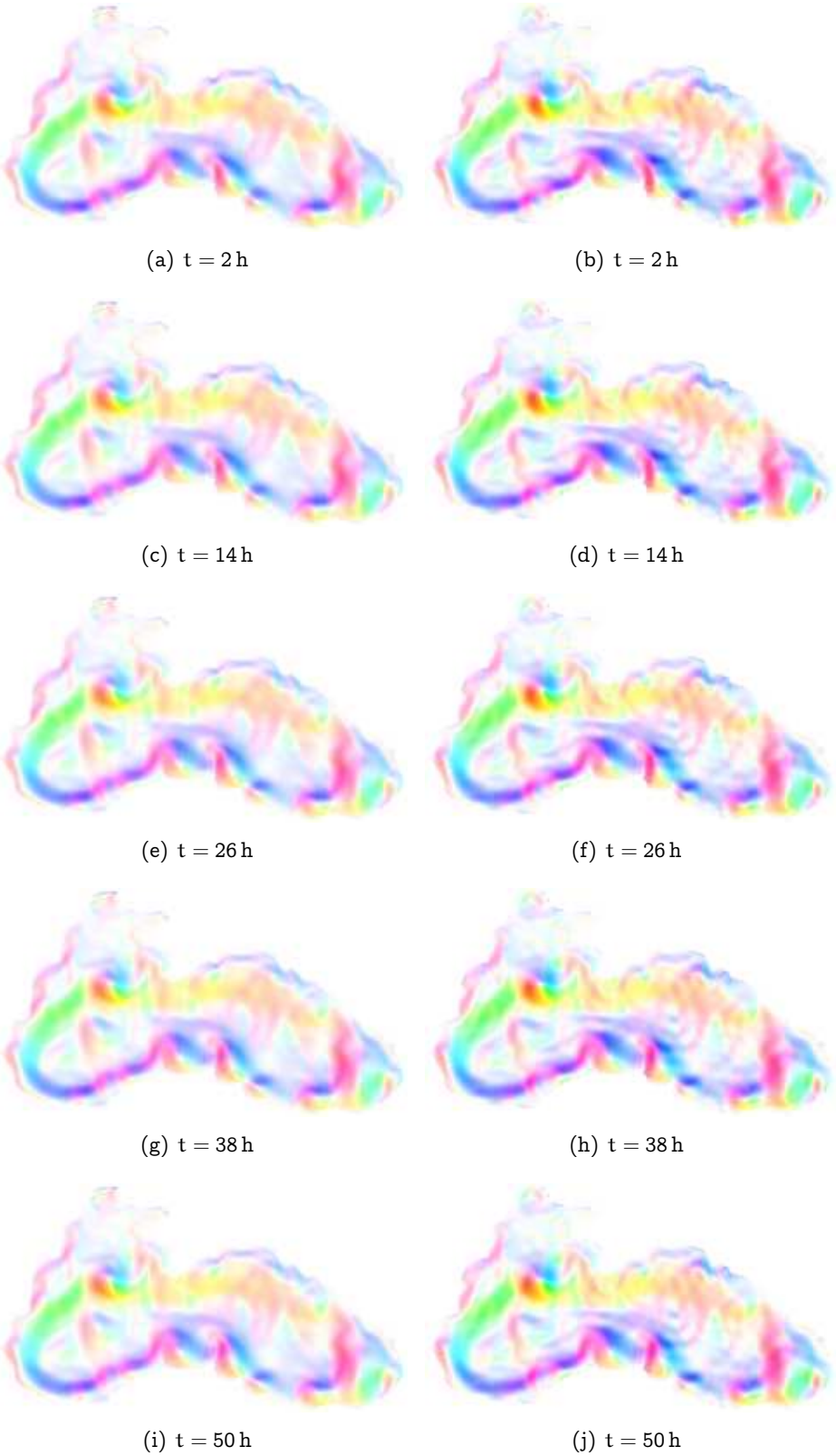
Les paramètres de l'expérience d'assimilation sont les suivants :

- $dt = 2\,h$.
- $Nt = 27 \quad (54\,h)$,
- $B = 1$,
- $R = 1$,
- dates d'observation : $2\,h,\ 8\,h,\ 14\,h,\ \ldots,\ 50\,h$

Choix des covariances. N'ayant aucune raisons de favoriser un cœfficient par rapport à un autre, les écarts-types d'erreur d'ébauche et d'observation sont donc choisis constants. La valeur de 1 pour ces écarts-types est arbitraire, modifier cette valeur revient à minimiser la fonction de coût à un facteur multiplicatif près, ce qui ne peut avoir qu'un impact numérique sur la minimisation.

Résultats. On obtient une séquence $\begin{pmatrix} a^a(t) & b^a(t) \end{pmatrix}^T$ à partir de laquelle on reconstruit l'image $I_s^a(\mathbf{x}, t)$ et le mouvement $\mathbf{w}^a(\mathbf{x}, t)$, pour $t \in [0\,h, 52\,h]$. Les résultats sont présentés sur les figures 5.23 et 5.24 et les statistiques d'erreurs sont fournies sur le tableau 5.5.

La NRMSE pour le mouvement est de 3.69% et la corrélation de 0.97. Les images I_s^a obtenues sont très proches des observations I^z : la corrélation entre les deux séquences est de 0.99 et la NRMSE de 0.71%.



(a) t = 2 h (b) t = 2 h

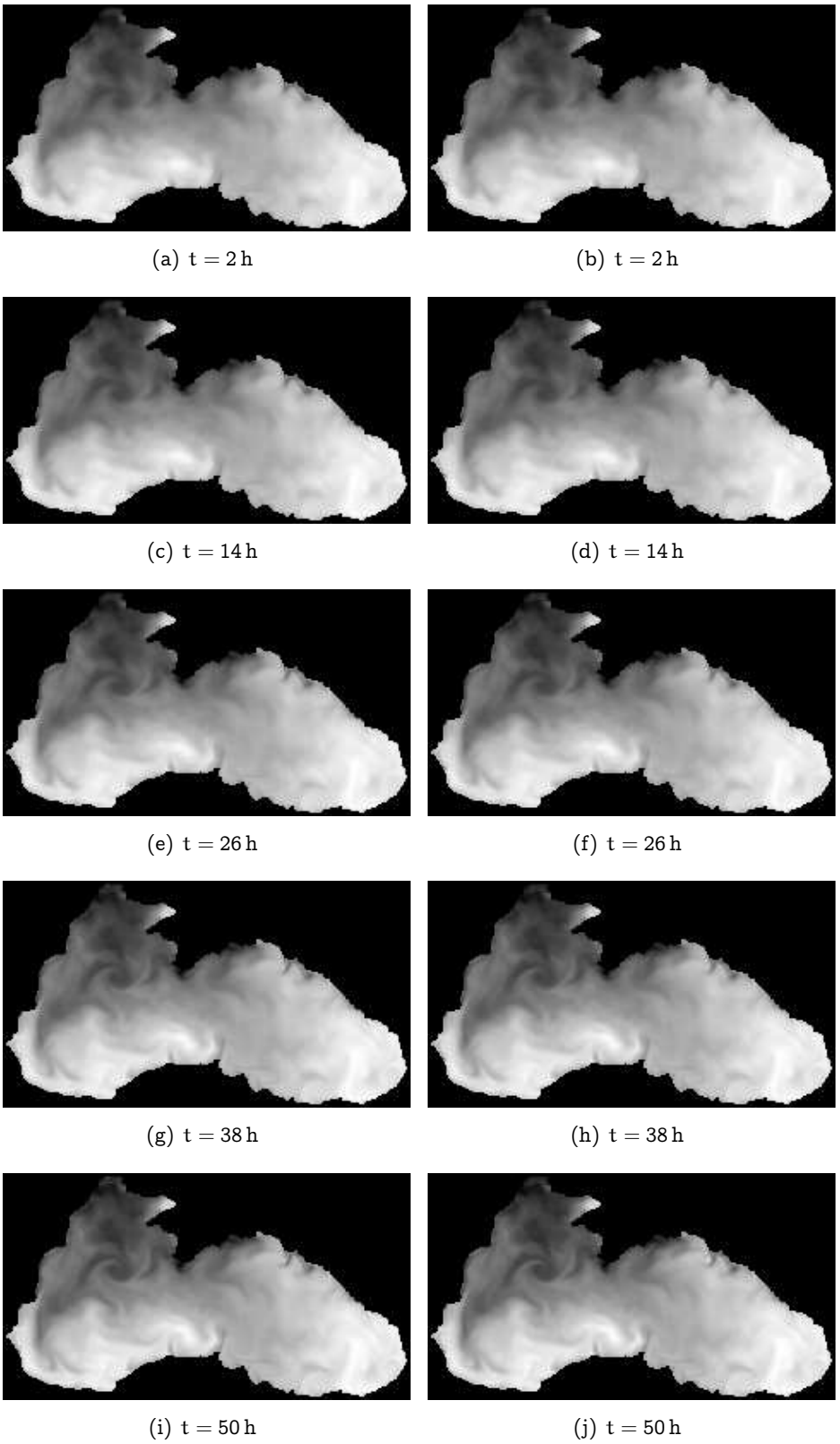
(c) t = 14 h (d) t = 14 h

(e) t = 26 h (f) t = 26 h

(g) t = 38 h (h) t = 38 h

(i) t = 50 h (j) t = 50 h

FIGURE 5.23 – Vérité terrain $\mathbf{w}$ du modèle complet (gauche), comparée aux résultats d'assimilation du modèle réduit $\mathbf{w}^a$ (droite).



(a) t = 2 h (b) t = 2 h

(c) t = 14 h (d) t = 14 h

(e) t = 26 h (f) t = 26 h

(g) t = 38 h (h) t = 38 h

(i) t = 50 h (j) t = 50 h

FIGURE 5.24 – Vérité terrain I_s du modèle complet (gauche), comparée aux résultats d'assimilation du modèle réduit I_s^q (droite).

	Mouvement $\mathbf{w}$			Pseudo-image I_s		
	valeur min	moyenne	valeur max	valeur min	moyenne	valeur max
Vérité terrain	-0.331	0.76e-3	0.278	19.5	22.8	24.7
Analyse	-0.364	2.63e-3	0.327	19.6	22.9	24.7

	$\mathbf{w}$	I_s
RMSE	0.0225	0.0375
NRMSE (%)	3.69	0.71
Corrélation	0.97	0.99

TABLE 5.5 – Statistiques d'erreur entre la vérité terrain et les résultats d'assimilation sur l'ensemble de la séquence.

Conclusions. L'assimilation dans le modèle MR-AIMI-MN permet d'obtenir une estimation précise du mouvement, comme le montrent les statistiques obtenues. Cette expérience jumelle démontre le potentiel de la méthode d'assimilation sur des bases fixes. Une amélioration possible de cette technique consiste à concevoir une base ondelettes image qui s'adapte au type d'images traitées. Cette amélioration potentielle est discutée dans la conclusion du chapitre (section 5.5).

5.5 Conclusions

Nous avons présenté, dans ce chapitre, des bases d'ondelettes pour le mouvement et les images, permettant de définir un modèle réduit adapté à l'estimation du mouvement sur le bassin de la mer Noire. Une expérience synthétique d'assimilation a permis de montrer le potentiel de cette méthode. Les avantages de cette approche sont de pouvoir imposer à la solution les propriétés choisies par l'utilisateur. Dans le cadre d'images océanographiques, nous avons choisi d'imposer une divergence nulle au mouvement estimé afin d'être approximativement cohérent avec la physique de cet écoulement particulier. De plus, cette méthode permet de respecter les conditions aux bords du domaine, choisies par l'utilisateur, pour des géométries complexes.

La principale perspective de ce travail concerne l'amélioration des bases fixes. On rappelle que la base d'ondelettes scalaire est obtenue en minimisant itérativement les systèmes suivants :

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i = \min\limits_{f \in L_2(\Omega)} \langle \nabla f, \nabla f \rangle , & (5.5.1) \\ \langle \psi_i, \psi_k \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.5.2) \\ \nabla \psi_i(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.5.3) \end{cases}$$

Dans le cas d'images de température de surface sur la mer Noire, on dispose de statistiques sur la localisation des zones de fort gradients. Soit $\{I^i(\mathbf{x})\}_{i=1\ldots N}$ un ensemble de N images satellites de température de surface et soit :

$$g(\mathbf{x}) = \frac{N}{\sum\limits_i^N \|\nabla I^i(\mathbf{x})\|} \tag{5.5.4}$$

l'inverse de la moyenne des normes des gradients spatiaux. En résolvant le système composé des équations 5.5.12 à 5.5.14, qui prend en compte les zones de fort gradient, on obtiendra une base adaptative codant l'information aux petites échelles dans ces zones :

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i^g = \min\limits_{f \in L_2(\Omega)} \langle \nabla f, \nabla f \times g \rangle, & (5.5.5) \\ \langle \psi_i^g, \psi_k^g \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.5.6) \\ \nabla \psi_i^g(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.5.7) \end{cases}$$

La pondération spatiale de la minimisation de la norme du gradient par la fonction g, dans l'équation 5.5.12, permet d'obtenir des fonctions pour lesquelles on pénalise moins la norme du gradient dans certaines zones. Ces fonctions représentent donc de plus petites échelles et permettent d'assimiler l'information à ces échelles, alors que le système constitué des équations 5.5.8 à 5.5.10 ne le permet pas.

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i = \min\limits_{f \in L_2(\Omega)} \langle \nabla f, \nabla f \rangle, & (5.5.8) \\ \langle \psi_i, \psi_k \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.5.9) \\ \nabla \psi_i(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.5.10) \end{cases}$$

Dans le cas d'images de température de surface sur la mer Noire, on dispose de statistiques sur la localisation des zones de fort gradients. Soit $\{I^i(\mathbf{x})\}_{i=1\ldots N}$ un ensemble de N images satellites de température de surface et soit :

$$g(\mathbf{x}) = \frac{N}{\sum\limits_i^N \|\nabla I^i(\mathbf{x})\|} \tag{5.5.11}$$

l'inverse de la moyenne des normes des gradients spatiaux. En résolvant le système composé des équations 5.5.12 à 5.5.14, qui prend en compte les zones de fort gradient, on obtiendra une base adaptative codant l'information aux petites échelles dans ces zones :

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i^g = \min\limits_{f \in L_2(\Omega)} \langle \nabla f, \nabla f \times g \rangle, & (5.5.12) \\ \langle \psi_i^g, \psi_k^g \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!], & (5.5.13) \\ \nabla \psi_i^g(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in \partial\Omega. & (5.5.14) \end{cases}$$

La pondération spatiale de la minimisation de la norme du gradient par la fonction g, dans l'équation 5.5.12, permet d'obtenir des fonctions pour lesquelles on pénalise moins la norme du gradient dans certaines zones. Ces fonctions représentent donc de plus petites échelles et permettent d'assimiler l'information à ces échelles, alors que le système constitué des équations 5.5.8 à 5.5.10 ne le permet pas.

D'une manière plus générale, on peut imaginer pour des applications différentes de l'océanographie, minimiser différents critères quadratiques et imposer d'autres contraintes linéaires et homogènes à chacune des bases. L'objectif de ce types de travaux serait, encore une fois, de profiter des qualités de l'assimilation de données pour l'estimation du mouvement, tout en imposant des propriétés issues de connaissances a priori sur la solution.

Chapitre 6

Conclusion générale et perspectives

Conclusions de la thèse

L'analyse d'une séquence d'images pour estimer le mouvement apparent nécessite, comme nous l'avons vu dans ce document, de mettre à profit diverses sources d'information. En effet, si ce problème est uniquement formulé par la contrainte du flot optique, il est mal posé comme beaucoup d'autres en vision par ordinateur. L'introduction de contraintes supplémentaires, issues d'informations a priori sur la séquence, est indispensable pour obtenir une solution unique. Dans le contexte de l'étude d'écoulements géophysiques, le formalisme de l'assimilation variationnelle de données se révèle être très adapté à l'analyse des images acquises par satellite. Qu'il s'agisse de météorologie ou d'océanographie, l'assimilation variationnelle de données fournit un cadre propice à l'introduction d'un modèle dynamique. C'est précisément ce modèle dynamique qui sert de contrainte supplémentaire à l'équation du flot optique, afin d'obtenir unicité de la solution. Modèles et données sont deux représentations complémentaires d'une même réalité et l'assimilation de données permet de réaliser un compromis entre ces deux représentations.

Cependant, la taille des images satellite rend difficile l'utilisation de ces approches. C'est pour cette raison que nous avons présenté, dans cette thèse, des méthodes permettant de contourner cette difficulté. L'objectif était d'appliquer des techniques de réduction de la dynamique, adaptées au traitement des images et en particulier des images satellite.

La première contribution de cette thèse a été d'estimer le mouvement en appliquant une projection de Galerkin sur des sous-espaces obtenus par décomposition orthogonale propre des images et des champs de mouvement. L'assimilation des cœfficients de projection des images $b_j^{obs}(t)$ dans le modèle réduit obtenu permet de calculer les cœfficients $a_i(t)$ du mouvement et de reconstruire les champs de mouvement. La méthode a été illustrée avec le modèle d'advection du mouvement et des images AIMI et celui du mouvement à divergence nulle AIMIDF. L'approche a été validée et quantifiée au travers d'expériences jumelles sur des données synthétiques. Ces expériences ont démontré que la réduction de la dynamique sur des sous-espaces POD permet de considérablement diminuer le temps de calcul et la mémoire nécessaires à l'application des algorithmes d'assimilation de données. Par ailleurs, la qualité des résultats obtenus en utilisant ces modèles réduits a été comparée aux résultats des modèles complets. Il en résulte que les modèles réduits fournissent des estimations presque aussi bonnes que les modèles complets.

Si la base image est calculée de façon immédiate par décomposition orthogonale propre

de la séquence d'images, l'obtention de la base mouvement requiert l'estimation préalable d'un champ de vitesse initial $\mathbf{w}_0$. Ce champ peut, par exemple, être calculé par des méthodes classiques de traitement d'images. Nous avons toutefois proposé au chapitre 3, une alternative à l'utilisation de ces techniques : un couplage des algorithmes d'assimilation dans les modèles complet et réduit est décrit dans la section 3.6. Pour cela, une expérience dite « fenêtre glissante »a été conçue. Elle permet de combiner les qualités respectives des deux modèles. Le principal intérêt de ce couplage réside dans la possibilité d'estimer le mouvement sur de longues séquences d'images en un temps de calcul raisonnable, tout en conservant la cohérence temporelle du résultat.

Le point commun à toutes les méthodes proposées dans la partie I est que les sous-espaces de projection dépendent des données image. Dans la seconde partie de la thèse, nous avons exploré la projection des équations d'évolution sur des bases fixes. Un des avantages majeurs de ce type de projection est de choisir a priori les propriétés de la solution estimée. En effet, comme expliqué dans la partie II, si $\Phi = \{\phi_i(\mathbf{x})\}_{i=1\ldots K}$ est la base fixe choisie, le résultat d'assimilation s'écrira $\mathbf{w}(\mathbf{x},t) = \sum_i^K a_i(t)\phi_i(\mathbf{x})$. Si l'on impose des propriétés aux fonctions ϕ_i, $\mathbf{w}(\mathbf{x},t)$ héritera ce ces mêmes propriétés en tant que combinaison linéaire des ϕ_i. Le second avantage d'utilisation de ces bases fixes est de significativement réduire la taille du vecteur d'état, et donc la mémoire nécessaire.

Dans le chapitre 4, une base de fonctions sinus a été utilisée pour représenter la vorticité. Ces fonctions ont été choisies pour leur régularité spatiale ainsi que leur respect des conditions aux bords du domaine spatial. De nombreuses expériences numériques et une comparaison à l'état de l'art ont été réalisées dans ce chapitre. Elles ont montré le potentiel de la méthode et de bons résultats sur des séquences d'images satellite bruitées.

Un autre type de bases fixes a été proposé au chapitre 5 : des bases d'ondelettes. L'application visée dans ce chapitre était l'estimation des courants de surface de la mer Noire, à partir d'images satellite de température de surface. Une des difficultés pour l'obtention de ces bases était de gérer un domaine spatial complexe (voir la figure 5.7). Les propriétés à imposer à la solution et donc à la base étaient les suivantes :

- régularité spatiale, il s'agit d'un écoulement géophysique et les variations spatiales brutales sont à exclure,
- respect des conditions de Neumann aux limites du domaine, car les courants sur les bords de la mer ne peuvent pas aller dans la direction perpendiculaire à ces bords,
- divergence nulle car, comme expliqué dans le chapitre 5, les courants de surface sont essentiellement horizontaux.

Des bases d'ondelettes vérifiant ces conditions ont étés calculées en minimisant l'intégrale de la norme des gradients spatiaux des fonctions, sous les contraintes de divergence nulle et de respect des conditions de Neumann aux bords du domaine. Une expérience jumelle d'assimilation sur des sorties de modèle MyOcean (sous-section 5.4.2) a montré le potentiel de la méthode pour l'estimation du mouvement.

Pour résumer, nous avons au cours de cette thèse, exploré l'utilisation de modèles dynamiques d'ordre réduit pour estimer le mouvement par assimilation de données. La première partie a permis de proposer des techniques de réduction du temps de calcul et de la mémoire nécessaire. La seconde partie a mis en évidence les divers avantages de la

projection de la dynamique sur bases fixes, en particulier dans le contexte de l'océanographie.

Perspectives de recherche

Réduction du temps de calcul

L'obtention d'un sous-espace POD pour le mouvement nécessite, comme nous l'avons déjà évoqué, l'estimation préalable d'un champ de vitesse initial $\mathbf{w}_0$. Ce champ est intégré en temps et la décomposition orthogonale propre est appliquée à la séquence obtenue. La qualité de la base POD représentant le mouvement repose exclusivement sur la qualité d'estimation de $\mathbf{w}_0$. La prise en compte de l'incertitude liée à cette estimation semble donc fondamentale pour s'assurer de la qualité de la base. Une manière de prendre en compte cette incertitude serait de concevoir une estimation robuste au bruit de la base mouvement. Cette base robuste, évoquée dans la conclusion du chapitre 2, est la principale perspective de développement pour ce qui concerne l'estimation du mouvement sur des bases réduites POD.

Projection de la dynamique sur bases fixes

Une étape importante dans la suite de ce travail concerne la projection de la dynamique sur bases fixes. Nous avons présenté, au chapitre 5, des bases adaptées aux traitement de séquences d'images océanographiques de la mer Noire. On rappelle que la base d'ondelettes scalaires est obtenue en minimisant itérativement les systèmes suivants :

$$S_i^{\mathbf{I}} = \begin{cases} \psi_i = \min\limits_{\mathbf{f}\in L_2(\Omega)} \langle \nabla \mathbf{f}, \nabla \mathbf{f} \rangle , \\ \nabla \psi_i \cdot \mathbf{n} = 0, \\ \langle \psi_i, \psi_k \rangle = \delta_{i,k}, \quad k \in [\![1, i]\!]. \end{cases} \tag{6.0.1}$$

Comme évoqué dans la conclusion du chapitre 5, une amélioration possible du calcul de cette base serait de pondérer spatialement la minimisation de la norme du gradient en fonction d'informations statistiques sur les zones de fort gradient dans les images. La base ainsi obtenue prendrait donc en compte une autre source d'information et pourrait ainsi coder de plus petites échelles dans certaines régions du domaine spatial.

Les perspectives pour l'assimilation sur base d'ondelettes sont nombreuses. Nous avons dans le contexte de l'estimation du mouvement sur le bassin de la mer Noire choisi un critère de minimisation quadratique et des contraintes linéaires homogènes adaptées à notre problématique. La démarche reste valable pour d'autres domaines d'application. Quel que soit le critère de minimisation quadratique et les contraintes linéaires homogènes, on peut, en résolvant un système similaire au système 6.0.1, construire une base adaptée à une autre problématique.

Troisième partie

Annexes

Annexe A

Composantes de l'algorithme d'assimilation 4D-Var

A.1 Calcul du gradient de la fonction de coût associé au système d'assimilation

On présente, dans cette section, le calcul du gradient de la fonction de coût associée à un système d'assimilation sous contrainte dynamique forte : c'est-à-dire sans terme d'erreur dans l'équation d'évolution. On rappelle tout d'abord les équations du système d'assimilation A.1.1 :

$$\begin{cases} \dfrac{\partial \mathbf{X}}{\partial t}(\mathbf{x},t) + \mathbb{M}(\mathbf{X})(\mathbf{x},t) = 0, \\ \mathbf{X}(\mathbf{x},0) = \mathbf{X}_b(\mathbf{x}) + \mathcal{E}_b(\mathbf{x}) \\ \mathbf{Y}(\mathbf{x},t) = \mathbb{H}\mathbf{X}(\mathbf{x},t) + \mathcal{E}_o(\mathbf{x},t) \end{cases} \tag{A.1.1}$$

où $\mathbb{M}$ est l'opérateur d'évolution et $\mathbb{H}$ est, dans tous les cas étudiés dans cette thèse, un opérateur de projection.

La fonction de coût, définie afin de minimiser $\mathcal{E}_o$ et $\mathcal{E}_b$, s'écrit :

$$\begin{aligned} J[\mathbf{X}(0)] &= \frac{1}{2}\int_\Omega (\mathbf{X}(\mathbf{x},0) - \mathbf{X}_b(\mathbf{x}))^T B^{-1}(\mathbf{x}) (\mathbf{X}(\mathbf{x},0) - \mathbf{X}_b(\mathbf{x})) \, d\mathbf{x} \\ &\quad + \frac{1}{2}\int_A [\mathbf{Y} - \mathbb{H}\mathbf{X}]^T(\mathbf{x},t) R^{-1}(\mathbf{x},t) [\mathbf{Y} - \mathbb{H}\mathbf{X}](\mathbf{x},t) d\mathbf{x}dt \end{aligned} \tag{A.1.2}$$

Minimiser J par rapport à $\mathbf{X}(0)$ étant équivalent à minimiser J par rapport à $\mathcal{E}_b$, on présente le calcul du gradient en fonction de $\mathcal{E}_b$. On commence donc par réécrire J en fonction de $\mathcal{E}_b$:

$$J[\mathcal{E}_b] = \frac{1}{2}\int_\Omega \mathcal{E}_b^T(\mathbf{x}) B^{-1}(\mathbf{x}) \mathcal{E}_b(\mathbf{x}) d\mathbf{x} + \frac{1}{2}\int_A [\mathbf{Y} - \mathbb{H}\mathbf{X}]^T(\mathbf{x},t) R^{-1}(\mathbf{x},t) [\mathbf{Y} - \mathbb{H}\mathbf{X}](\mathbf{x},t) d\mathbf{x}dt$$

On écrit la valeur de δJ associée à un incrément $\delta\mathcal{E}_b$ de l'erreur $\mathcal{E}_b$, avec $\delta\mathbf{X}$ l'incrément résultant sur $\mathbf{X}$:

$$\delta J = \int_\Omega \delta\mathcal{E}_b^T(\mathbf{x})B^{-1}(\mathbf{x})\mathcal{E}_b(\mathbf{x})d\mathbf{x} + \int_A \delta\mathbf{X}(\mathbf{x},t)^T\mathbb{H}^T R^{-1}(\mathbf{x},t)[\mathbb{H}\mathbf{X}-\mathbf{Y}](\mathbf{x})d\mathbf{x}dt \quad (A.1.3)$$

En utilisant les deux premières équations du système A.1.1, on obtient :

$$\begin{aligned} \frac{\partial(\mathbf{X}+\delta\mathbf{X})}{\partial t}(\mathbf{x},t) + \mathbb{M}(\mathbf{X}+\delta\mathbf{X})(\mathbf{x},t) &= 0 \\ \mathbf{X}(\mathbf{x},0)+\delta\mathbf{X}(\mathbf{x},0) &= \mathbf{X}_b(\mathbf{x})+\mathcal{E}_b(\mathbf{x})+\delta\mathcal{E}_b(\mathbf{x}) \end{aligned}$$

On fait un développement de Taylor pour obtenir :

$$\begin{aligned} \frac{\partial\delta\mathbf{X}}{\partial t}(\mathbf{x},t) + \frac{\partial\mathbb{M}}{\partial\mathbf{X}}(\mathbf{X})(\mathbf{x},t)\delta\mathbf{X}(\mathbf{x},t) = 0 \quad (A.1.4)\\ \delta\mathbf{X}(\mathbf{x},0) = \delta\mathcal{E}_b(\mathbf{x}) \end{aligned}$$

On définit la variable adjointe $\lambda(\mathbf{x},t)$ vérifiant le système suivant :

$$\begin{aligned} -\frac{\partial\lambda}{\partial t}(\mathbf{x},t) + \left(\frac{\partial\mathbb{M}}{\partial\mathbf{X}}\right)^*\lambda(\mathbf{x},t) &= \mathbb{H}^T R^{-1}(\mathbf{x},t)[\mathbb{H}\mathbf{X}(t)-Y(t)](\mathbf{x},t) \quad (A.1.5)\\ \lambda(\mathbf{x},T) &= 0 \end{aligned}$$

Dans la première équation du système A.1.4, on multiplie chaque membre par $\lambda(\mathbf{x},t)$ et on intègre en temps et en espace afin d'obtenir :

$$\int_t \left\langle \frac{\partial\delta\mathbf{X}(t)}{\partial t}, \lambda(t)\right\rangle + \int_t \left\langle \frac{\partial\mathbb{M}}{\partial\mathbf{X}}\delta\mathbf{X}(t), \lambda(t)\right\rangle = 0$$

On intègre le premier terme par parties, et on utilise la définition de l'adjoint pour le deuxième :

$$-\int_t \left\langle \delta\mathbf{X}(t), \frac{\partial\lambda(t)}{\partial t}\right\rangle + <\delta\mathbf{X}(T),\lambda(T)> - <\delta\varepsilon_B,\lambda(0)> + \int_t \left\langle \delta\mathbf{X}(t), \frac{\partial\mathbb{M}}{\partial\mathbf{X}}^*\lambda(t)\right\rangle = 0$$

le terme $<\delta\mathbf{X}(T),\lambda(T)>$ est nul, et on utilise la première équation du système A.1.5 pour écrire :

$$-\left\langle \delta\mathbf{X}(t), \frac{\partial\lambda(t)}{\partial t}\right\rangle + \left\langle \delta\mathbf{X}(t), \frac{\partial\mathbb{M}}{\partial\mathbf{X}}^*\lambda(t)\right\rangle = <\delta\mathbf{X}(t), \mathbb{H}^T R^{-1}[\mathbb{H}\mathbf{X}(t)-\mathbf{Y}(t)]>$$

On obtient alors l'équation A.1.6 :

$$\int_t <\delta\mathbf{X}(t), \mathbb{H}^T R^{-1}[\mathbb{H}\mathbf{X}(t)-Y(t)]> = <\delta\mathcal{E}_b,\lambda(0)> \quad (A.1.6)$$

On reprend l'équation A.1.3, et on utilise l'équation A.1.6 pour écrire :

$$\delta J = <\delta\mathcal{E}_b, B^{-1}\mathcal{E}_b> + <\delta\mathcal{E}_b,\lambda(0)>$$

Donc :

$$\nabla J_{\mathcal{E}_b}[\mathcal{E}_b] = B^{-1}\mathcal{E}_b + \lambda(0) \quad (A.1.7)$$

Remarque. Dans le cas d'un système d'assimilation pour un modèle réduit, ne dépendant que du temps selon le système A.1.8 et l'équation A.1.9, un raisonnement analogue aboutit au même résultat que A.1.7.

$$\begin{cases} \dfrac{d\mathbf{X}_R}{dt}(t) + \mathbb{M}_R(\mathbf{X})(t) = 0 \\ \mathbf{X}_R(0) = \mathbf{X}_b + \mathcal{E}_b \\ \mathbf{Y}(t) = \mathbb{H}\mathbf{X}_R(t) + \mathcal{E}_o(t), \end{cases} \tag{A.1.8}$$

$$\begin{aligned} J(\mathbf{X}_R(0)) = & \frac{1}{2}\left(\mathbf{X}_R(0) - \mathbf{X}_b\right)^T B^{-1}\left(\mathbf{X}_R(0) - \mathbf{X}_b\right) \\ & + \frac{1}{2}\int_{[0,T]} \left(\mathbf{Y}(t) - \mathbb{H}\mathbf{X}_R(t)\right)^T R^{-1}\left(\mathbf{Y}(t) - \mathbb{H}\mathbf{X}_R(t)\right) dt \end{aligned} \tag{A.1.9}$$

A.2 Algorithme d'assimilation du modèle réduit

A.2.1 Description synthétique de l'algorithme

1. Lecture des paramètres
2. Lecture du vecteur de Background $\mathbf{X}_b$
3. Initialisation
   - $k = 0$, k est le numéro d'itération
   - $\mathbf{X}_R(0)^0 = \mathbf{X}_b$
   - Run Forward, Calcul de J^0
   - Run Backward, Calcul de ∇J^0
4. Boucle itérative sur k, faire :
   - $k = k + 1$
   - Mise à jour de $\mathbf{X}_R^k$: $\mathbf{X}_R(0)^k = \mathbf{X}_R(0)^{k-1} - \gamma \nabla J^{k-1}$. Ce calcul est effectué par le minimiseur Broyden-Fletcher-Goldfarb-Shanno [47].
   - Si la minimisation est terminée, fin de la boucle.
   - Run Forward, Calcul de J^k
   - Run Backward, Calcul de ∇J^k.
5. Run Forward, calcul du résultat $X_R(t)$.

A.2.2 Définition des variables et des dimensions

- N_f : Nombre de pas de temps de la fenêtre d'assimilation.
- N_{obs} : Nombre d'observations utilisées pour l'assimilation.
- $\mathbf{X}_b$: vecteur d'ébauche de taille L,
- $\mathbf{X}_R$: vecteur d'état de taille $N_f \times (K + L)$,
- J : fonction de coût scalaire,
- $J1$, $J2$: termes de la fonction de coût (respectivement terme d'ébauche et d'observation, scalaires),
- ∇J : gradient de la fonction de coût, de taille $K + L$,

- ∇J_b : composante en b du vecteur ∇J, de taille L,
- bObs : vecteur d'observation, de taille $N_{obs} \times L$.
- i : indice d'observation,
- k : indice d'itération,
- Pour $i = 1 \dots Nobs$, R_i : matrice diagonale qui représente la variance d'erreur des observations, de taille $L \times L$.
- B : matrice diagonale qui représente la variance d'erreur de l'ébauche, de taille $L \times L$.
- γ : paramètre de descente de gradient, utilisé par le minimiseur BFGS.

A.2.3 Description des runs Forward et Backward

Run Forward.

1. Initialisation :
   - $J = J_1 = J_2 = 0$
   - $i = 0$
2. Boucle temporelle : $t = 1 \rightarrow N_f - 1$:
   - Integre($\mathbf{X}_R[t-1]$,$\mathbf{X}_R[t]$)
     Calcul de $\mathbf{X}_R[t]$ en fonction de $\mathbf{X}_R[t-1]$.
   - Si t est une date d'observation :
     - ◇ Pour $l = 1 \dots L$: $J2+ = \dfrac{(bObs[i] - b[t])^2}{R[i,l]}$
     - ◇ $i = i + 1$
   - Pour $l = 1 \dots L$: $J1+ = \dfrac{(b[0,i] - bObs[0,i])^2}{B[l]}$
3. $J = \dfrac{1}{2}J_1 + \dfrac{1}{2}J_2$

Run Backward.

1. Initialisation
   - $\nabla J = 0$
   - Création d'une variable auxiliaire ∇J_{tmp} pour le run backward
   - $i = N_{obs} - 1$
2. Boucle temporelle : $t = N_f - 1 \rightarrow 1$:
   - Integre-b($\mathbf{X}_R[t]$,∇J_{tmp},$\mathbf{X}_R[t-1]$,∇J)
     Calcul de $\lambda[t-1]$ (contenu dans ∇J_{tmp}) en fonction de $\lambda[t]$ (contenu dans ∇J).
   - $\nabla J = \nabla J_{tmp}$
   - Si t est une date d'observation :
     - ◇ Pour $l = 1 \dots L$: $\nabla J_l - = \dfrac{(bObs[i,l] - b[t,l])}{R[i,l]}$ où ∇J_l est la $l^{ième}$ composante de ∇J.
     - ◇ $i = i - 1$
3. Pour $l = 1 \dots L$: $\nabla J_l + = \dfrac{(b[0,l] - bObs[0,l])}{B_l}$,
   on ajoute le terme d'ébauche.

Annexe B

Calcul des modèles tangent et adjoint dans le cas des modèles réduits

B.1 Modèle réduit d'advection

Soit $a = \begin{pmatrix} a_1 & \dots & a_K \end{pmatrix}^T$ le vecteur des cœfficients a_i du vecteur d'état.
On rappelle les équations du modèle réduit pour a :

$$\mathbb{M}(a) = \begin{pmatrix} \mathbb{M}_1(a) \\ \cdot \\ \cdot \\ \mathbb{M}_K(a) \end{pmatrix} = \begin{pmatrix} a^T B(1) a \\ \cdot \\ \cdot \\ a^T B(K) a \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{K} \sum_{j=1}^{K} a_i B(1)_{i,j} a_j \\ \cdot \\ \cdot \\ \sum_{i=1}^{K} \sum_{j=1}^{K} a_i B(K)_{i,j} a_j \end{pmatrix} \tag{B.1.1}$$

On souhaite calculer le modèle tangent $\left(\frac{\partial \mathbb{M}}{\partial a}\right)\Big|_a$ et le modèle adjoint $\left(\frac{\partial \mathbb{M}}{\partial a}\right)^*\Big|_a$ de $\mathbb{M}$.

Calcul du tangent.

$$\begin{aligned} \mathbb{M}_j(a) &= a^T B(j) a = \sum_{l=1}^{K} \sum_{m=1}^{K} a_l B(j)_{l,m} a_m \\ &= \sum_{l=1, l\neq i}^{K} \sum_{m=1}^{K} a_l B(j)_{l,m} a_m + \sum_{m=1}^{K} a_i B(j)_{i,m} a_m \\ &= \sum_{l=1, l\neq i}^{K} \sum_{m=1, m\neq i}^{K} a_l B(j)_{l,m} a_m + \sum_{m=1, m\neq i}^{K} a_i B(j)_{i,m} a_m \\ &\quad + \sum_{l=1, l\neq i}^{K} a_l B(j)_{l,i} a_i + a_i^2 B(j)_{i,i} \end{aligned} \tag{B.1.2}$$

donc :

$$\left(\frac{\partial \mathbb{M}_j}{\partial a_i}\right)\Big|_a = \sum_{m=1,m\neq i}^{K} B(j)_{i,m}a_m + \sum_{l=1,l\neq i}^{K} a_l B(j)_{l,i} + 2a_i B(j)_{i,i}$$

$$= \sum_{l=1,l\neq i}^{K} a_l \left[B(j)_{i,l} + B(j)_{l,i}\right] + 2a_i B(j)_{i,i}$$

$$\left(\frac{\partial \mathbb{M}_j}{\partial a_i}\right)\Big|_a = \sum_{l=1}^{K} a_l \left[B(j)_{l,i} + B(j)_{i,l}\right], \tag{B.1.3}$$

soit :

$$\left(\frac{\partial \mathbb{M}_j}{\partial a}\right)\Big|_a = a^T \left[B(j) + B(j)^T\right], \tag{B.1.4}$$

En reprenant une écriture matricielle, la matrice tangente de taille $K \times K$ s'écrit :

$$\left(\frac{\partial \mathbb{M}}{\partial a}\right)\Big|_a = \begin{pmatrix} a^T \left[B(1) + B(1)^T\right] \\ \vdots \\ a^T \left[B(K) + B(K)^T\right] \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{l=1}^{K} a_l \left[B(1)_{l,1} + B(1)_{1,l}\right] & \dots & \sum_{l=1}^{K} a_l \left[B(1)_{l,K} + B(1)_{K,l}\right] \\ \vdots & & \\ \sum_{l=1}^{K} a_l \left[B(K)_{l,1} + B(K)_{1,l}\right] & \dots & \sum_{l=1}^{K} a_l \left[B(K)_{l,K} + B(K)_{K,l}\right] \end{pmatrix} \tag{B.1.5}$$

Calcul de l'adjoint. L'adjoint du tangent est dans ce cas la transposée du modèle tangent, décrit par l'équation B.1.5, soit la matrice suivante de taille $K \times K$:

$$\left(\frac{\partial \mathbb{M}}{\partial a}\right)^*\Big|_a = \begin{pmatrix} \sum_{l=1}^{K} a_l \left[B(1)_{l,1} + B(1)_{1,l}\right] & \dots & \sum_{l=1}^{K} a_l \left[B(K)_{l,1} + B(K)_{1,l}\right] \\ \vdots & & \\ \sum_{l=1}^{K} a_l \left[B(1)_{l,K} + B(1)_{K,l}\right] & \dots & \sum_{l=1}^{K} a_l \left[B(K)_{l,K} + B(K)_{K,l}\right] \end{pmatrix} \tag{B.1.6}$$

B.2 Calcul du tangent et de l'adjoint des équations sur b

On souhaite ajouter, à l'équation sur les cœfficients a_i, de la section précédente (section B.1), les équations sur les cœfficients b_j correspondant aux pseudo-images. On note $\mathbf{X}_R = \begin{pmatrix} a \\ b \end{pmatrix}$ le vecteur d'état. Avec $a = \begin{pmatrix} a_1 & \dots a_K \end{pmatrix}^T$ vecteur de taille K et $b =$

$\begin{pmatrix} b_1 \dots b_L \end{pmatrix}^T$ de taille L.

On considère le modèle d'évolution suivant :

$$\mathbb{M}(\mathbf{X}_R) = \begin{pmatrix} \mathbb{M}^{(1)}(\mathbf{X}_R) \\ \mathbb{M}^{(2)}(\mathbf{X}_R) \end{pmatrix} = \begin{pmatrix} \mathbb{M}^{(1)}(a) \\ \mathbb{M}^{(2)}(a,b) \end{pmatrix} = \begin{pmatrix} \mathbb{M}_1^{(1)}(a) \\ \cdot \\ \cdot \\ \mathbb{M}_K^{(1)}(a) \\ \mathbb{M}_1^{(2)}(a,b) \\ \cdot \\ \cdot \\ \mathbb{M}_L^{(2)}(a,b) \end{pmatrix} = \begin{pmatrix} a^T B(1) a \\ \cdot \\ \cdot \\ a^T B(K) a \\ a^T G(1) b \\ \cdot \\ \cdot \\ a^T G(L) b \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{K}\sum_{j=1}^{K} a_i B(1)_{i,j} a_j \\ \cdot \\ \cdot \\ \sum_{i=1}^{K}\sum_{j=1}^{K} a_i B(K)_{i,j} a_j \\ \sum_{i=1}^{K}\sum_{j=1}^{L} a_i G(1)_{i,j} b_j \\ \cdot \\ \cdot \\ \sum_{i=1}^{K}\sum_{j=1}^{L} a_i G(L)_{i,j} b_j \end{pmatrix} \tag{B.2.1}$$

Calcul du tangent.

$$\left(\frac{\partial \mathbb{M}}{\partial \mathbf{X}_R}\right)\Big|_{\mathbf{X}} = \begin{pmatrix} \left(\frac{\partial \mathbb{M}^{(1)}}{\partial a}\right)\Big|_a & 0 \\ \left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}} & \left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)\Big|_{\mathbf{X}} \end{pmatrix} \tag{B.2.2}$$

La sous-section B.1 explicite le calcul de $\left(\frac{\partial \mathbb{M}^{(1)}}{\partial a}\right)\Big|_a$.

Reste à calculer $\left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}}$ et $\left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)\Big|_{\mathbf{X}}$.

On obtient pour $i = 1 \dots K$ et $j = 1 \dots L$:

$$\left(\frac{\partial \mathbb{M}_j^{(2)}}{\partial a_i}\right)\Big|_{\mathbf{X}} = \sum_{m=1}^{L} G(j)_{i,m} b_m \tag{B.2.3}$$

donc $\left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}}$ est une matrice de taille $L \times K$ qui s'écrit :

$$\left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}} = \begin{pmatrix} \sum_{m=1}^{L} G(1)_{1,m} b_m & \dots & \sum_{m=1}^{L} G(1)_{K,m} b_m \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \sum_{m=1}^{L} G(L)_{1,m} b_m & \dots & \sum_{m=1}^{L} G(L)_{K,m} b_m \end{pmatrix} = \begin{pmatrix} b^T G(1)^T \\ \cdot \\ \cdot \\ b^T G(L)^T \end{pmatrix} \tag{B.2.4}$$

De même, on obtient pour $i = 1 \dots L$ et $j = 1 \dots L$:

$$\left(\frac{\partial \mathbb{M}_j^{(2)}}{\partial b_i}\right)\Big|_{\mathbf{X}} = \sum_{l=1}^{K} a_l G(j)_{l,i} \tag{B.2.5}$$

donc $\left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)_{\mathbf{X}}$ est une matrice de $L \times L$ qui s'écrit :

$$\left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)\Big|_{\mathbf{X}} = \begin{pmatrix} \sum_{l=1}^{K} a_l G(1)_{l,1} & \dots & \sum_{l=1}^{K} a_l G(1)_{l,L} \\ \vdots & & \vdots \\ \sum_{l=1}^{K} a_l G(L)_{l,1} & \dots & \sum_{l=1}^{K} a_l G(L)_{l,L} \end{pmatrix} = \begin{pmatrix} a^T G(1) \\ \vdots \\ a^T G(L) \end{pmatrix} \tag{B.2.6}$$

Calcul de l'adjoint.

$$\left(\frac{\partial \mathbb{M}}{\partial \mathbf{X}_R}\right)\Big|_{\mathbf{X}}^{*} = \begin{pmatrix} \left(\frac{\partial \mathbb{M}^{(1)}}{\partial a}\right)\Big|_{a}^{*} & \left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}}^{*} \\ 0 & \left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)\Big|_{\mathbf{X}}^{*} \end{pmatrix} \tag{B.2.7}$$

La sous-section B.1 explicite le calcul de $\left(\frac{\partial \mathbb{M}^{(1)}}{\partial a}\right)\Big|_{a}^{*}$.

Il reste à calculer $\left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}}^{*}$ et $\left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)\Big|_{\mathbf{X}}^{*}$.

On obtient par application de la transposition :

$$\left(\frac{\partial \mathbb{M}^{(2)}}{\partial a}\right)\Big|_{\mathbf{X}}^{*} = \begin{pmatrix} \sum_{m=1}^{L} G(1)_{1,m} b_m & \dots & \sum_{m=1}^{L} G(L)_{1,m} b_m \\ \vdots & & \vdots \\ \sum_{m=1}^{L} G(1)_{K,m} b_m & \dots & \sum_{m=1}^{L} G(L)_{K,m} b_m \end{pmatrix} = (G(1)b, \dots, G(L)b) \tag{B.2.8}$$

et :

$$\left(\frac{\partial \mathbb{M}^{(2)}}{\partial b}\right)\Big|_{\mathbf{X}}^{*} = \begin{pmatrix} \sum_{l=1}^{K} a_l G(1)_{l,1} & \dots & \sum_{l=1}^{K} a_l G(L)_{l,1} \\ \vdots & & \vdots \\ \sum_{l=1}^{K} a_l G(1)_{l,L} & \dots & \sum_{l=1}^{K} a_l G(L)_{l,L} \end{pmatrix} = (G(1)^T a, \dots, G(L)^T a) \tag{B.2.9}$$

Annexe C

Intégration numérique des modèles complets

C.1 Modèle d'advection du mouvement et des images (nommé AIMI)

C.1.1 Définition du vecteur d'état

Cette partie concerne le modèle 2D d'advection du mouvement et des images, nommé AIMI, dont l'état s'écrit : $\mathbf{X}(\mathbf{x},t) = \begin{pmatrix} u(\mathbf{x},t) \\ v(\mathbf{x},t) \\ I_s(\mathbf{x},t) \end{pmatrix}$. Il est défini sur le domaine spatio-temporel $\Omega * [t_0, t_1]$. Les variables d'état sont les composantes u et v de la vitesse, et une pseudo-image I_s. La fonction $\mathbf{X}(\mathbf{x},t) = (u(\mathbf{x},t), v(\mathbf{x},t), I_s(\mathbf{x},t))^T$ est à valeurs dans $\mathbb{R}^3$.

C.1.2 Dynamique du modèle

On considère le modèle d'évolution temporelle : $\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}(\mathbf{X}(t)) = 0$ avec une équation d'advection pour la vitesse et la pseudo-image.
L'équation en u s'écrit :

$$\begin{aligned} \frac{du}{dt} &= 0 \\ \Leftrightarrow \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} &= 0 \end{aligned}$$

L'équation en v s'écrit :

$$\begin{aligned} \frac{dv}{dt} &= 0 \\ \Leftrightarrow \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} &= 0 \end{aligned}$$

Le modèle s'écrit donc :

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = 0 \quad \text{(C.1.1)}$$

$$\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial y} + u\frac{\partial v}{\partial x} = 0 \quad \text{(C.1.2)}$$

$$\frac{\partial I_s}{\partial t} + u\frac{\partial I_s}{\partial x} + v\frac{\partial I_s}{\partial y} = 0 \quad \text{(C.1.3)}$$

C.1.3 Split des équations d'évolution

Évolution de u. Pour l'évolution en u, on splitte l'équation C.1.1 en deux composantes : une advection non-linéaire en x et une advection linéaire en y :

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0 \quad \text{(C.1.4)}$$

$$\frac{\partial u}{\partial t} + v\frac{\partial u}{\partial y} = 0 \quad \text{(C.1.5)}$$

On réécrit l'équation C.1.4 sous une forme conservative :

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + \frac{\partial B(u)}{\partial x} = 0$$
$$\text{avec } B(u) = \frac{1}{2}u^2$$

Évolution de v. Pour l'évolution en v, on splitte l'équation C.1.2 en deux composantes : une advection linéaire en x et une advection non-linéaire en y :

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} = 0 \quad \text{(C.1.6)}$$

$$\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial y} = 0 \quad \text{(C.1.7)}$$

L'équation C.1.7 est également réécrite sous une forme conservative :

$$\frac{\partial v}{\partial t} + v\frac{\partial v}{\partial y} = \frac{\partial v}{\partial t} + \frac{\partial B(v)}{\partial y} = 0$$
$$\text{avec } B(v) = \frac{1}{2}v^2$$

Évolution de I_s. L'équation d'évolution de I_s, équation C.1.3, est splittée en 2 composantes : advection linéaire en x et advection linéaire en y :

$$\frac{\partial I_s}{\partial t} + u\frac{\partial I_s}{\partial x} = 0 \quad \text{(C.1.8)}$$

$$\frac{\partial I_s}{\partial t} + v\frac{\partial I_s}{\partial y} = 0 \quad \text{(C.1.9)}$$

C.1.4 Schémas de discrétisation

Advection non linéaire du mouvement. Pour la partie non linéaire de l'advection de u et de v (équations C.1.4 et C.1.7), on utilise le schéma de Godounov.

Le schéma de Godounov. Le schéma de Godounov s'écrit :

$$\begin{aligned} u_{ij}^{n+1} &= u_{ij}^{n} - \frac{\Delta t}{\Delta x}(g_{godounov}(u_{ij}^{n}, u_{i+1j}^{n}) - g_{godounov}(u_{i-1j}^{n}, u_{ij}^{n})) \\ v_{ij}^{n+1} &= v_{ij}^{n} - \frac{\Delta t}{\Delta y}(g_{godounov}(v_{ij}^{n}, v_{i+1j}^{n}) - g_{godounov}(v_{i-1j}^{n}, v_{ij}^{n})) \end{aligned}$$

Soit la fonction $f(s) = \frac{s^2}{2}$. $g_{godounov}(s_1, s_2)$ s'écrit :

- si $s_1 < s_2$:

$$\begin{aligned} g_{godounov}(s_1, s_2) &= f(s_2) \text{ si } s_2 \leq 0 \\ g_{godounov}(s_1, s_2) &= f(s_1) \text{ si } s_1 \geq 0 \\ g_{godounov}(s_1, s_2) &= 0 \text{ si } s_1 \times s_2 \leq 0 \end{aligned}$$

- si $s_1 \geq s_2$:

$$\begin{aligned} g_{godounov}(s_1, s_2) &= f(s_2) \text{ si } s_1 \leq 0 \\ g_{godounov}(s_1, s_2) &= f(s_1) \text{ si } s_2 \geq 0 \end{aligned}$$

Advection linéaire. Plusieurs schémas sont possibles pour l'advection linéaire de u, v et I_s (équations C.1.5, C.1.6, C.1.8, et C.1.9) :

- Upwind d'ordre 1 (nommé UP1)
- Schéma centré

On utilise le schéma UP1 pour l'advection linéaire du mouvement et le schéma centré pour l'advection de I_s.

Les équations C.1.5, C.1.6, C.1.8 et C.1.9 se discrétisent comme :

$$\begin{aligned} u_{ij}^{n} &= u_{ij}^{n-1} - \frac{\Delta t}{\Delta y}\left[\max\left(v_{ij}^{n-1}, 0\right)\left(u_{ij}^{n-1} - u_{ij-1}^{n-1}\right) + \min\left(v_{ij}^{n-1}, 0\right)\left(u_{ij+1}^{n-1} - u_{ij}^{n-1}\right)\right] \\ v_{ij}^{n} &= v_{ij}^{n-1} - \frac{\Delta t}{\Delta x}\left[\max\left(u_{ij}^{n-1}, 0\right)\left(v_{ij}^{n-1} - v_{i-1j}^{n-1}\right) + \min\left(u_{ij}^{n-1}, 0\right)\left(v_{i+1j}^{n-1} - v_{ij}^{n-1}\right)\right] \\ I_{s,ij}^{n} &= I_{s,ij}^{n-1} - \frac{\Delta t}{2\Delta x}\left[u_{i+1j}^{n}\left(I_{s,i+1j} - I_{s,ij}\right) + u_{ij}^{n}\left(I_{s,ij} - I_{s,i-1j}\right)\right] \\ I_{s,ij}^{n} &= I_{s,ij}^{n-1} - \frac{\Delta t}{2\Delta y}\left[v_{ij+1}^{n}\left(I_{s,ij+1} - I_{s,ij}\right) + v_{ij}^{n}\left(I_{s,ij} - I_{s,ij-1}\right)\right] \end{aligned}$$

C.1.5 Conditions aux bords

On choisit pour u et u des conditions nulles aux bords et pour I_s on opte pour une copie. Les variables u, v et I_s sont des images de taille $N_x \times N_y$. Afin d'appliquer les conditions aux bords, on ajoute un pixel de chaque côté du domaine image. On travaille donc sur des images étendues de taille $(N_x+2) \times (N_y+2) = N_x 2b \times N_y 2b$. Lors de l'initialisation, on applique les conditions aux bords afin de donner une valeur aux points ajoutés.

Les points de bord sont notés :

- $u(0, j)$ et $u(N_x 2b - 1, j)$ pour les bords ouest et est, $u(i, 0)$ et $u(i, N_y 2b - 1)$ pour les bords sud et nord.

- $v(0,j)$ et $v(N_x 2b-1,j)$ pour les bords ouest et est, $v(i,0)$, $v(i,N_y 2b-1)$ pour les bords sud et nord.
- $I_s(0,j)$ et $I_s(N_x 2b-1,j)$ pour les bords ouest et est, $I_s(i,0)$ et $I_s(i,N_y 2b-1)$ pour les bords sud et nord.

Conditions nulles aux bords.
On annule les points de bords pour u et v.
Pour u :

- boucle de $j = 1 \rightarrow N_y 2b-2$ (bords ouest et est)
  $u(0,j) = 0$
  $u(N_x 2b-1,j) = 0$
- boucle de $i = 0 \rightarrow N_x 2b-1$ (bords sud et nord)
  $u(i,0) = 0$
  $u(i,N_y 2b-1) = 0$

Pour v :

- boucle de $i = 1 \rightarrow N_x 2b-2$ (bords sud et nord)
  $v(i,0) = 0$
  $v(i,N_y 2b-1) = 0$
- boucle de $j = 0 \rightarrow N_y 2b-1$ (bords ouest et est)
  $v(0,j) = 0$
  $v(N_x 2b-1,j) = 0$

Conditions aux bords pour I_s.
Les conditions aux bords pour I_s sont les suivantes :

- boucle de $j = 1 \rightarrow N_y 2b-2$ (bords ouest et est)
  $I_s(0,j) = I_s(1,j)$
  $I_s(N_x 2b-1,j) = I_s(N_x 2b-2,j)$
- boucle de $i = 0 \rightarrow N_x 2b-1$ (bords sud et nord)
  $I_s(i,0) = I_s(i,1)$
  $I_s(i,N_y 2b-1) = I_s(i,Ny2b-2)$

C.2 Modèle d'advection du mouvement et des images sous la contrainte d'une divergence nulle (AIMIDF)

C.2.1 État

Cette partie concerne le modèle 2D d'advection du mouvement et des images sous la contrainte d'une divergence nulle AIMIDF, dont l'état s'écrit : $\mathbf{X}(\mathbf{x},t) = \begin{pmatrix} \xi(\mathbf{x},t) \\ I_s(\mathbf{x},t) \end{pmatrix}$. Il est défini sur le domaine spatio-temporel $\Omega * [t_0,t_1]$. Les variables d'état sont la vorticité ξ, et une pseudo-image I_s. La fonction $\mathbf{X}(\mathbf{x},t) = (\xi(\mathbf{x},t), I_s(\mathbf{x},t))^T$ est à valeur dans $\mathbb{R}^2$.

C.2.2 Dynamique du modèle

On considère le modèle d'évolution temporelle : $\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}(\mathbf{X}(t)) = 0$ avec une équation d'advection pour la vorticité et la pseudo-image par la vitesse. Le modèle s'écrit :

$$\frac{\partial \xi}{\partial t} + u\frac{\partial \xi}{\partial x} + v\frac{\partial \xi}{\partial y} = 0 \quad \text{(C.2.1)}$$

$$\frac{\partial I_s}{\partial t} + u\frac{\partial I_s}{\partial x} + v\frac{\partial I_s}{\partial y} = 0 \quad \text{(C.2.2)}$$

C.2.3 Schémas de discrétisation

Une fois le vecteur vitesse $\mathbf{w}(\xi)(\mathbf{x}, t)$ calculé par la méthode décrite dans la section 3.2, il reste à discrétiser l'advection linéaire de la vorticité $\xi(\mathbf{x}, t)$ et des pseudo-images $I_s(\mathbf{x}, t)$.

Les schémas de discrétisation choisis pour l'advection linéaire de $\xi(\mathbf{x}, t)$ et $I_s(\mathbf{x}, t)$ sont des schémas centrés.

C.2.4 Conditions aux bords

On choisit pour ξ des conditions nulles aux bords et pour I_s on opte pour une copie. Les variables u, v et I_s sont des images de taille $N_x \times N_y$ Afin d'appliquer les conditions nulles aux bords pour ξ et de copie pour I_s, on ajoute un pixel de chaque coté de l'image. On travaille donc sur des images étendues de taille $(N_x + 2) \times (N_y + 2) = N_x 2b \times N_y 2b$. Lors de l'initialisation on applique les conditions aux bords pour donner une valeur aux points ajoutées.

Les points de bords sont notés :

- $\xi(0, j)$ et $\xi(N_x 2b - 1, j)$ pour les bords ouest et est, $\xi(i, 0)$ et $\xi(i, N_y 2b - 1)$ pour les bords sud et nord.
- $I_s(0, j)$ et $I_s(N_x 2b - 1, j)$ pour les bords ouest et est, $I_s(i, 0)$ et $I_s(i, N_y 2b - 1)$ pour les bords sud et nord.

Conditions nulles aux bords.
On annule les points de bords pour ξ :

- boucle de $j = 1 \rightarrow N_y 2b - 2$ (bords ouest et est)
  $\xi(0, j) = 0$
  $\xi(N_x 2b - 1, j) = 0$
- boucle de $i = 0 \rightarrow N_x 2b - 1$ (bords sud et nord)
  $\xi(i, 0) = 0$
  $\xi(i, N_y 2b - 1) = 0$

Conditions aux bords pour I_s.
Les conditions aux bords pour I_s sont les suivantes :

- boucle de $j = 1 \rightarrow N_y 2b - 2$ (bords ouest et est)
  $I_s(0, j) = I_s(1, j)$
  $I_s(N_x 2b - 1, j) = I_s(N_x 2b - 2, j)$

- boucle de $i = 0 \rightarrow N_x 2b - 1$ (bords sud et nord)
  $I_s(i, 0) = I_s(i, 1)$
  $I_s(i, N_y 2b - 1) = I_s(i, Ny2b - 2)$

Annexe D

Articles

D.1 Complément au chapitre 2 : Assimilation d'images dans un modèle réduit pour l'estimation du mouvement.

Cet article [14] a été publié dans le colloque Gretsi 2011 (Groupement de Recherche en Traitement du Signal et des Images [1]). La conférence s'est déroulée sur le campus bordelais à l'ENSEIRB-MATMECA [2] de Talence du 06 au 09 septembre 2011. Les travaux présentés dans cette publication sont décrits dans le chapitre 2.

1. http://www.gretsi.fr/
2. http://www.enseirb-matmeca.fr/

Assimilation d'images dans un modèle réduit pour l'estimation du mouvement

Karim DRIFI[1, 2], Isabelle HERLIN[1, 2]

[1]INRIA, Institut National de Recherche en Informatique et Automatique
Domaine de Voluceau, Rocquencourt - BP 105 78153 Le Chesnay Cedex, France

[2]CEREA, joint laboratory ENPC - EDF R&D, Université Paris-Est
6-8 avenue Blaise Pascal, Cité Descartes Champs-sur-Marne 77455 Marne la Vallée Cedex 2, France
karim.drifi@inria.fr, isabelle.herlin@inria.fr

Résumé – Cet article décrit une méthode d'estimation du champ de vitesse apparent, sous-jacent à l'évolution temporelle d'une séquence d'images. Un modèle d'évolution, dit **complet**, est choisi pour représenter la dynamique du champ de vitesse et des images. La méthode de décomposition orthogonale propre est appliquée et fournit des bases de représentation des champs de vitesse et des images. La projection de Galerkin du modèle complet sur ces bases réduites définit alors le modèle **réduit**. Un algorithme d'assimilation variationelle de données est conçu afin d'estimer les cœfficients des champs de vitesse à partir des cœfficients des images observées. Le mouvement est ensuite restitué à partir de ces cœfficients estimés. La méthode est validée sur des données synthétiques afin de quantifier les résultats.

Abstract – This paper concerns the estimation of apparent motion from an image sequence. A model, named **full model**, is defined to describe the dynamics of motion field and images. The Proper Orthogonal Decomposition (POD) is used to derive basis to characterize motion fields and images. The Galerkin projection of the full model on these basis defines the so-called **reduced model**. Coefficients of motion fields are retrieved from those of observations with the help of a 4D-var assimilation method. Last motion results are obtained from the coefficients. The method has been tested on twin experiments to validate the approach.

1 Introduction

L'estimation du mouvement est un problème fondamental du traitement d'image. Il s'agit d'estimer le champ de vitesse $\mathbf{v}(\mathbf{x}, t)$ à partir d'une séquence d'images. Dans le cas d'acquisitions satellite, les applications concernent principalement l'océanographie et la météorologie. La littérature décrit l'estimation du mouvement par des algorithmes de traitement d'image, par exemple Horn &Schunck [4], ou par des algorithmes d'assimilation de données [1, 2, 7]. Ces dernières méthodes exploitent un modèle dynamique, issu d'une connaissance a priori sur la séquence. Leurs principales limitations sont le temps de calcul et le volume de mémoire nécessaires. Un moyen d'éviter ces difficultés est d'utiliser une technique dite de réduction. Un modèle réduit, obtenu par projection de Galerkin, a été utilisé par D'Adamo et Papadakis dans [3, 6] pour estimer la dynamique à partir d'observations du champ de mouvement acquises par « Particle Image Velocimetry ». Le modèle dynamique réduit décrit l'évolution temporelle des cœfficients de projection $a_i(t)$ sur une base $\Phi = \{\phi_i(\mathbf{x})\}_{i=1\ldots K}$ afin d'approximer l'évolution temporelle du champ de vitesse $\mathbf{v}(\mathbf{x}, t)$. Cette base Φ est obtenue par décomposition orthogonale propre [5] des observations.

Dans cet article, on souhaite estimer un mouvement, $\mathbf{v}(\mathbf{x}, t)$ à partir d'une séquence discrète d'images $I = \{I^z\}_z = \{I(\mathbf{x}, t_z)\}_z$, acquise aux dates t_i.
On considère le vecteur d'état $\mathbf{X}_C(\mathbf{x}, t) = (\mathbf{v}(\mathbf{x}, t), q(\mathbf{x}, t))^T$, où le traceur passif q est une quantité scalaire équivalente aux images de I. L'évolution temporelle de $\mathbf{X}_C(\mathbf{x}, t)$ est décrite par un modèle complet $\mathbb{M}_C$, choisi à partir d'heuristiques. Le vecteur $\mathbf{X}_C(\mathbf{x}, t)$ est projeté sur un espace réduit, engendré par des bases Φ et Ψ, dont l'obtention est décrite ultérieurement :

- $\mathbf{v}(\mathbf{x}, t)$ est projeté sur Φ et associé aux cœfficients $a_i(t)$,
- $q(\mathbf{x}, t)$ est projeté sur Ψ et associé aux cœfficients $b_j(t)$.

La projection de Galerkin du modèle complet $\mathbb{M}_C$, sur Φ et Ψ, fournit un modèle réduit $\mathbb{M}_R$ décrivant l'évolution des cœfficients $a_i(t)$ et $b_j(t)$. Afin de réaliser l'estimation du mouvement, les images observées I^z sont projetées sur Ψ, fournissant des valeurs $b_j^{obs}(t_z)$, qui sont ensuite assimilées dans $\mathbb{M}_R$. Les cœfficients $a_i(t)$ ainsi obtenus permettent de restituer le champs de mouvement $\mathbf{v}(\mathbf{x}, t)$.

Comment sont obtenues les bases Ψ et Φ ? La méthode de décomposition orthogonale propre, appliquée à I, permet de calculer Ψ. Un algorithme de calcul de mouvement est utilisé sur les deux premières images I^1 et I^2 de la séquence afin

d'obtenir une initialisation $\mathbf{v}_0$. Une séquence $V = \{\mathbf{v}(\mathbf{x}, t_z)\}_z$ est ensuite calculée en intégrant $\mathbf{v}_0$ avec le modèle $\mathbb{M}_C$. Une décomposition orthogonale propre est enfin appliquée à V afin d'obtenir la base Φ.

La Section 2 décrit le calcul des bases réduites au moyen de la décomposition orthogonale propre. La Section 3 résume la projection de Galerkin du modèle complet afin d'obtenir le modèle réduit. La Section 4 décrit le cadre de l'assimilation variationnelle de données dans le cas particulier d'un modèle réduit pour l'estimation du mouvement. Enfin, la Section 5 présente l'application de la méthode pour des expériences jumelles.

2 Calcul des bases réduites

On résume, dans cette section, l'obtention des bases réduites $\Psi = \{\psi_j(\mathbf{x})\}_{j=1\dots L}$, pour la représentation du traceur q, et $\Phi = \{\phi_i(\mathbf{x})\}_{i=1\dots K}$, pour le mouvement $\mathbf{v}$.

Décomposition orthogonale propre Dans un cadre général, on considère une séquence discrète $E = \{E^m\}_{m\in\{1,\dots,M\}}$. E peut être scalaire, $E = I$, ou vectorielle $E = V$. Un élément E^m est représenté sur la base canonique $\{e_n\}_{n=1\dots N}$:

$$\forall m \in \{1,\dots,M\}, \qquad E^m = \sum_{n=1}^{N} E_n^m e_n \tag{1}$$

avec N le nombre de pixels. Considérons la matrice $\underline{E}$ de taille $N \times M$, telle que l'élément de la $n^{\text{ième}}$ ligne et $m^{\text{ième}}$ colonne est E_n^m. Soit $G = \underline{E}\,\underline{E}^T$ la matrice de Gram de taille $N \times N$. Les P premiers vecteurs propres, expliquant 90% de la variance de E, sont choisis pour constituer la base réduite $\Gamma = \{\gamma_p\}_{p=1\dots P}$.

3 Projection de Galerkin

Soit $\mathbb{M}_C$ le modèle dynamique complet associé au vecteur d'état $\mathbf{X}_C(\mathbf{x}, t)$. On a :

$$\frac{\partial \mathbf{X}_C}{\partial t}(\mathbf{x}, t) + \mathbb{M}_C\left(\mathbf{X}_C(\mathbf{x}, t)\right) = 0$$

Dans l'article, $\mathbf{X}_C(\mathbf{x}, t) = (\mathbf{v}(\mathbf{x}, t), q(\mathbf{x}, t))$. $\mathbb{M}_C$ caractérise la dynamique de $\mathbf{v}(\mathbf{x}, t)$ et de $q(\mathbf{x}, t)$. Les équations d'évolution choisies sont l'advection de la vitesse par elle-même et l'advection du traceur par la vitesse. La condition initiale, à $t = 0$, de $\mathbf{v}(\mathbf{x}, t)$ et $q(\mathbf{x}, t)$ est $\mathbf{X}_C(\mathbf{x}, 0) = (\mathbf{v}_0(\mathbf{x}), q_0(\mathbf{x}))$. On obtient ainsi le système :

$$\begin{cases} \dfrac{\partial \mathbf{v}}{\partial t}(\mathbf{x}, t) + \left(\mathbf{v}(\mathbf{x}, t) \cdot \nabla\right) \mathbf{v}(\mathbf{x}, t) = 0 \\ \dfrac{\partial q}{\partial t}(\mathbf{x}, t) + \mathbf{v}(\mathbf{x}, t) \cdot \left(\nabla q(\mathbf{x}, t)\right) = 0 \\ \mathbf{v}(\mathbf{x}, 0) = \mathbf{v}_0(\mathbf{x}), \quad q(\mathbf{x}, 0) = q_0(\mathbf{x}) \end{cases} \tag{2}$$

Soient $a_i(t)$ les cœfficients de projection de $\mathbf{v}(\mathbf{x}, t)$ sur ϕ_i, $\mathbf{v}(\mathbf{x}, t)$ peut être approximé par $\sum_{i=1}^{K} a_i(t)\phi_i(\mathbf{x})$. De même $q(\mathbf{x}, t)$ est approximé par $\sum_{j=1}^{L} b_j(t)\psi_j(\mathbf{x})$ avec b_j les cœfficients de projection de $q(\mathbf{x}, t)$ sur ψ_j. $\mathbf{v}(\mathbf{x}, t)$ et $q(\mathbf{x}, t)$ sont remplacés par leurs approximations dans le Système (2) afin d'obtenir :

$$\begin{cases} \displaystyle\sum_{i=1}^{K} \frac{da_i}{dt}(t)\phi_i(\mathbf{x}) \\ \quad + \left[\sum_{i=1}^{K} \left(a_i(t)\phi_i(\mathbf{x})\right) \cdot \nabla\right] \sum_{j=1}^{K} a_j(t)\phi_j(\mathbf{x}) \quad = 0 \\ \displaystyle\sum_{i=1}^{L} \frac{db_i}{dt}(t)\psi_i(\mathbf{x}) \\ \quad + \left[\sum_{i=1}^{K} \left(a_i(t)\phi_i(\mathbf{x})\right)\right] \cdot \nabla \left[\sum_{j=1}^{L} b_j(t)\psi_j(\mathbf{x})\right] \quad = 0 \\ \sum_{i=1}^{K} a_i(0)\phi_i(\mathbf{x}) = \sum_{i=1}^{K} a_{i0}\phi_i(\mathbf{x}) \\ \sum_{j=1}^{L} b_j(0)\psi_j(\mathbf{x}) = \sum_{j=1}^{L} b_{j0}\psi_j(\mathbf{x}) \end{cases} \tag{3}$$

On fait ensuite le produit scalaire du Système d'équations (3) avec les ϕ_k et ψ_l afin d'obtenir le système d'ODE suivant :

$$\begin{cases} \dfrac{da_k}{dt}(t) + \left\langle \left(\displaystyle\sum_{i=1}^{K} a_i(t)\phi_i \cdot \nabla\right) \sum_{i=1}^{K} a_i(t)\phi_i, \phi_k \right\rangle = 0, \\ \dfrac{db_l}{dt}(t) + \left\langle \left(\displaystyle\sum_{i=1}^{K} a_i(t)\phi_i\right) \cdot \nabla \left[\sum_{j=1}^{L} b_j(t)\psi_j, \psi_l\right] \right\rangle = 0, \\ a_k(0) = a_{k0}, \quad b_l(0) = b_{l0}, \end{cases} \tag{4}$$

où $\langle \cdot, \cdot \rangle$ représente le produit scalaire dans l'espace des fonctions de carré sommable L_2 :

$$\langle f, g \rangle = \int f(\mathbf{x}) g(\mathbf{x}) d\mathbf{x}. \tag{5}$$

Le Système (4) se simplifie en :

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T(t) B(k) a(t) = 0, \quad k = 1 \dots K. \\ \dfrac{db_l}{dt}(t) + a^T(t) G(l) b(t) = 0, \quad l = 1 \dots L. \\ a(0) = a_0, \quad b(0) = b_0 \end{cases} \tag{6}$$

avec :

- $a(t) = (a_1(t), \dots, a_K(t))^T$, $b(t) = (b_1(t), \dots, b_L(t))^T$,
- $B(k)$ la matrice de taille $K \times K$ dont les éléments sont $B(k)_{i,j} = \langle (\phi_i \cdot \nabla)\, \phi_j, \phi_k \rangle$,
- $G(l)$ la matrice de taille $K \times L$ dont les éléments sont $G(l)_{i,j} = \langle \phi_i \cdot \nabla \psi_j, \psi_l \rangle$,
- a_0 et b_0 vecteurs de tailles respective K et L.

Soit $\mathbf{X}_R(\mathbf{x}, t) = (a(t), b(t))^T$, appelé vecteur d'état réduit, le Système (6) se réécrit :

$$\begin{cases} \dfrac{d\mathbf{X}_R}{dt}(\mathbf{x}, t) + \mathbb{M}_R\left(\mathbf{X}_R(\mathbf{x}, t)\right) = 0 \\ \mathbf{X}_R(\mathbf{x}, 0) = \mathbf{X}_{R0}(\mathbf{x}) = (a_0, b_0)^T \end{cases} \tag{7}$$

$\mathbb{M}_R$ est appelé modèle réduit et correspond à la projection de Galerkin de $\mathbb{M}_C$ sur Φ et Ψ.

4 Assimilation variationnelle de données

On décrit succinctement, dans cette section, les principes de l'assimilation variationnelle de données, pour le cas d'un modèle réduit ne dépendant que du temps.

4.1 Formulation

Soit $\mathbf{X}_R$ un vecteur d'état dépendant du temps et défini sur $[0,T]$. $\mathbf{X}_R$ obéit à l'équation :

$$\frac{d\mathbf{X}_R}{dt}(t) + \mathbb{M}_R(\mathbf{X})(t) = \mathcal{E}_m(t) \tag{8}$$

$\mathbb{M}_R$ est supposé differentiable et décrit la dynamique du vecteur d'état $\mathbf{X}_R$. $\mathcal{E}_m$ est l'erreur modèle, qui quantifie l'écart entre la dynamique effective de $\mathbf{X}_R$ et $\mathbb{M}_R$. On considère des observations $\mathbf{Y}(t)$, qui sont liées à $\mathbf{X}_R$ par l'équation d'observation :

$$\mathbf{Y}(t) = \mathbb{H}\mathbf{X}_R(t) + \mathcal{E}_O(t) \tag{9}$$

$\mathbb{H}$ est la matrice associée à la projection $(a(t), b(t))^T \rightarrow b(t)^T$. $\mathcal{E}_O$ représente l'erreur liée à l'observation. On considère une estimation $\mathbf{X}_{R,b}$ du vecteur d'état à la date $t = 0$:

$$\mathbf{X}_R(0) = \mathbf{X}_{R,b} + \mathcal{E}_b \tag{10}$$

$\mathcal{E}_b$ est l'erreur associée à cette ébauche. L'intégration de $\mathbf{X}_{R,b}$ sur la fenêtre temporelle $[0,T]$ permet d'obtenir l'ébauche $\mathbf{X}_R(t)$. On fait l'hypothèse que les erreurs $\mathcal{E}_m$, $\mathcal{E}_O$ et $\mathcal{E}_b$ sont décorrélées en temps et gaussiennes de moyenne nulle. Elles sont caractérisées par leurs matrices de covariance respectives Q, R et B.

4.2 Méthode variationnelle

On minimise la fonction de coût E pour estimer $\mathbf{X}_R$:

$$\begin{aligned} E(\mathbf{X}_R) = \; & \tfrac{1}{2}\int_{[0,T]} \left(\tfrac{d\mathbf{X}_R}{dt} + \mathbb{M}_R(\mathbf{X}_R)\right)^T Q^{-1} \left(\tfrac{d\mathbf{X}_R}{dt} + \mathbb{M}_R(\mathbf{X}_R)\right) dt \\ & + \tfrac{1}{2}\int_{[0,T]} \left(\mathbf{Y} - \mathbb{H}(\mathbf{X}_R)\right)^T R^{-1} \left(\mathbf{Y} - \mathbb{H}(\mathbf{X}_R)\right) dt \\ & + \tfrac{1}{2} \left(\mathbf{X}_R(0) - \mathbf{X}_{R,b}\right)^T B^{-1} \left(\mathbf{X}_R(0) - \mathbf{X}_{R,b}\right) \end{aligned}$$

Cette fonction exprime une contrainte dynamique faible, avec une erreur modèle représentée par le premier terme. Le système d'optimalité à résoudre pour minimiser E s'obtient en calculant les équations d'Euler-Lagrange :

$$\lambda(T) = 0 \tag{11}$$

$$-\frac{d\lambda}{dt} + \left(\frac{\partial \mathbb{M}_R}{\partial \mathbf{X}}\right)^* \lambda = \mathbb{H}^T R^{-1} \left(\mathbf{Y} - \mathbb{H}(\mathbf{X}_R)\right) \tag{12}$$

$$\mathbf{X}_R(0) = B\lambda(0) + \mathbf{X}_{R,b} \tag{13}$$

$$\frac{d\mathbf{X}_R}{dt} + \mathbb{M}_R(\mathbf{X}_R) = Q\lambda(t) \tag{14}$$

avec :

- λ : variable adjointe du système (même dimension que $\mathbf{X}_R$),
- $\left(\frac{\partial \mathbb{M}_R}{\partial \mathbf{X}}\right)$ le modèle tangent linéaire et $\left(\frac{\partial \mathbb{M}_R}{\partial \mathbf{X}}\right)^*$ son adjoint.

Le calcul du système d'optimalité (Eqs.11 à 14) ainsi que sa résolution par un algorithme incrémental est détaillé dans [1]. Le principe est d'estimer à chaque itération l'ébauche $\mathbf{X}_{R,b}$, en ajoutant un incrément $\delta\mathbf{X}$, calculé par la méthode adjointe.

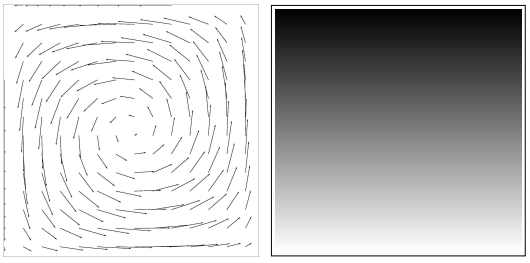
Vecteur d'état Dans le cas de l'estimation du mouvement, le vecteur d'état est constitué de l'ensemble des cœfficients a_i et b_j du modèle réduit : $\mathbf{X}_R(t) = (a_1(t), \ldots, b_L(t))^T$.

Matrices de covariances d'erreur Les matrices de covariance d'erreur Q, R et B proviennent d'une modélisation des erreurs $\mathcal{E}_m$, $\mathcal{E}_o$ et $\mathcal{E}_b$ suivant une loi normale. On estime ces matrices à partir des observations pour les b_j et de la projection sur Φ de la séquence V pour les a_i.

5 Expérience et Résultats

5.1 Description de l'expérience jumelle

Une simulation du modèle complet, Équation (2), est réalisée sur une fenêtre temporelle $[0,T]$, à partir de conditions initiales $(\mathbf{v}(\mathbf{x},0), q(\mathbf{x},0))$ (voir Figure 1). Une séquence $(\mathbf{v}(\mathbf{x},t), q(\mathbf{x},t))$ est ainsi obtenue. Le choix des dates d'observation t_z fournit les images $I^z = q(\mathbf{x},t_z)$, utilisées pour calculer la base Ψ, et les champs $\mathbf{v}(\mathbf{x},t_z)$, nécessaires à l'obtention de la base Φ. 4 vecteurs sont conservés pour chaque base ($K = 4$, $L = 4$). Le modèle réduit (6) est alors défini. Les observations utilisées



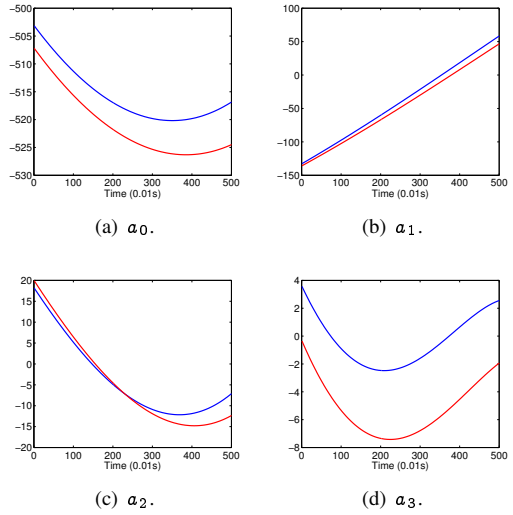
(a) $\mathbf{v}_0$. (b) q_0.

FIGURE 1 – Conditions initiales.

pour l'assimilation sont les cœfficients de projection $b_j^{obs}(t_z)$ des acquisitions images I^z. La première observation est choisie comme ébauche pour b à la date $t = 0$. L'ébauche $\mathbf{X}_{R,b} = (a_b, b_b)^T$ est donc obtenue par projection de $(0, I^1)^T$ et vaut $\mathbf{X}_{R,b} = (0, b^{obs}(t_1))^T$.

5.2 Résultats

Les courbes a_i^a, obtenues après le processus d'assimilation et les courbes a_i^p, obtenues par projection de $\mathbf{v}(\mathbf{x},t)$ sur Φ, sont représentées sur la Figure 2. Les courbes a_i^a sont très proches



(a) a_0. (b) a_1. (c) a_2. (d) a_3.

FIGURE 2 – Résultats d'assimilation : a_i^a estimés par assimilation (rouge), projections a_i^p (bleu).

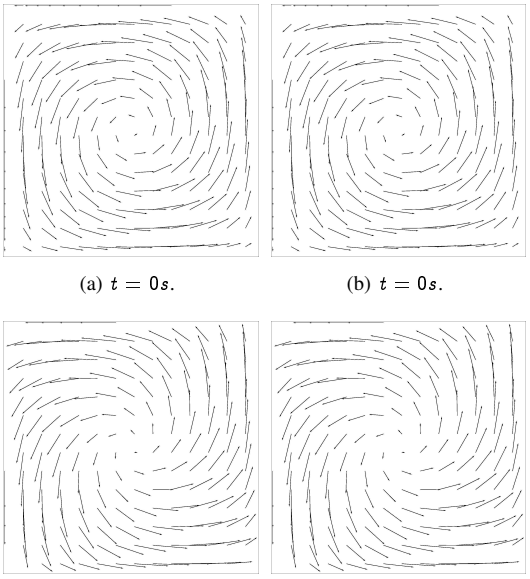
des courbes a_i^p. En particulier a_1, a_2 et a_3 sont très bien estimés. L'évolution de a_4^a est très proche de celle de a_4^p avec un biais constant de 4. Les courbes a_i^a obtenues par assimilation permettent ensuite de restituer des champs de vitesses $\mathbf{v}^a(\mathbf{x},t)$. Ces champs sont comparés Figure 3 à la séquence originale $\mathbf{v}(\mathbf{x},t)$. Quelques statistiques d'erreurs sont fournies Tableau 1 pour différentes dates. Les champs $\mathbf{v}^a(\mathbf{x},t)$ approximent avec précision la séquence $\mathbf{v}(\mathbf{x},t)$. L'erreur relative en norme et l'erreur angulaire restent inférieures, sur toute la séquence, à $\mathbf{1.9\%}$ et $\mathbf{3.70}$ degrés respectivement.

TABLE 1 – Statistiques d'erreur entre la séquence synthétique $\mathbf{v}(\mathbf{x},t)$ et la séquence reconstruite $\mathbf{v}^a(\mathbf{x},t)$.

	Erreurs	
	relative en norme	**maximum angulaire**
t		
$0s$	0.011	2.901
$2.5s$	0.014	2.780
$5s$	0.019	1.671

5.3 Conclusions et perspectives

Dans cet article, nous avons conçu et présenté une méthode d'assimilation de données dans un modèle dynamique réduit. Cette approche permet d'estimer des champs de vitesse à partir d'une séquence d'images discrète. Une expérience jumelle, réalisée sur des données synthétiques, démontre l'intérêt de la méthode. Des tests ont étés effectués en ajoutant un bruit additif sur les observations (écart type du bruit à 5% de celui du signal). Ils ont démontré la stabilité des résultats obtenus. Les champs de vitesse sont correctement estimés à partir de vecteurs d'état à 8 composantes. Le calcul de la base Φ du mouvement utilise une estimation bruitée de la condition initiale $\mathbf{v}_0$ par un algorithme de traitement d'image. Une perspective de l'étude est donc de réaliser une base robuste Φ à partir de $\mathbf{v}_0$.



(a) $t = 0s$. (b) $t = 0s$. (c) $t = 5s$. (d) $t = 5s$.

FIGURE 3 – Champs de vitesse $\mathbf{v}^a(\mathbf{x},t)$ (gauche) estimés comparés aux données synthétiques $\mathbf{v}(\mathbf{x},t)$ (droite).

Acknowledgements Ces travaux sont partiellement soutenus par le projet Geo-FLUIDS (ANR 09 SYSC 005 02).

Références

[1] Dominique Béréziat and Isabelle Herlin. Solving ill-posed image processing problems using data assimilation. *Numerical Algorithms*, 54, 2011.

[2] Th. Corpetti, P. Héas, E. Mémin, and N. Papadakis. Variational pressure image assimilation for atmospheric motion estimation. In *Proc. Int. Geoscience and Remote Sensing Symp. (IGARSS'08)*, volume 2, pages 505–508, Boston, MA, July 2008.

[3] J. D'Adamo, N. Papadakis, E. Mémin, and Artana G. Variational assimilation of POD low-order dynamical systems. *Journal of Turbulence*, 8(9) :1–22, 2007.

[4] B.K.P. Horn and B.G. Schunk. Determining optical flow. *Artificial Intelligence*, 17 :185–203, 1981.

[5] M. Loève. *Probability Theory*. Springer-Verlag, 1978.

[6] N. Papadakis. *Assimilation de données images : application au suivi de courbes et de champs de vecteurs*. PhD thesis, université de Rennes 1, Mention Mathématiques et Applications, 2007.

[7] O. Titaud, A. Vidard, I. Souopgui, and F.-X. Le Dimet. Assimilation of image sequences in numerical models. *Tellus A*, 62 :30–47, 2010.

D.2 Complément au chapitre 3 : Learning reduced models for motion estimation on ocean satellite images.

Cet article [18] à été publié dans la conférence « Hydrodynamic modeling of the Black Sea Dynamics » qui s'est déroulée à Sevastopol en Ukraine au mois de septembre 2011. Les travaux présentés dans cette publication sont décrits dans le chapitre 3.

Learning reduced models for motion estimation on ocean satellite images

Isabelle Herlin[1,2] Dominique Béréziat[3] Karim Drifi[1,2]

[1]: INRIA – B.P. 105, 78153 Le Chesnay Cedex, France.
[2]: CEREA, joint laboratory ENPC - EDF R&D, Université Paris-Est – Cité Descartes, Champs-sur-Marne, 77455 Marne la Vallée, France.
[3]: Université Pierre et Marie Curie – 4 place Jussieu, 75005 Paris, France.

Abstract

The paper describes a learning method on sliding windows for estimating apparent motion on long temporal satellite sequences acquired over oceans. A "full model", which is defined on the pixel grid, is chosen to describe the dynamics of motion fields and images, based on heuristics of divergence-free motion and advection of image brightness by the velocity. The image sequence is split into small temporal windows that half overlap in time. Image assimilation in the full model is applied on the first window to retrieve its motion field. This makes it possible to define subspaces of motion fields and images and a "reduced model" is defined by applying the Galerkin projection of the full model on these subspaces. Data assimilation in the reduced model is applied on this second window. The process is iterated for the next window until the end of the whole image sequence. Each reduced model is then learned from the previous one. The main advantage of the approach is the small computational requirements of the assimilation in the reduced models that make it feasible to process in quasi-real time image acquisitions. Twin experiments have been designed to quantify the full model and the learning method on sliding windows and demonstrate the quality of the motion fields estimated by the approach.

Keywords: Motion Estimation, Data Assimilation, Model Reduction, Galerkin projection

1 Introduction

Motion estimation from an image sequence has been intensively studied since the beginning of image processing (Horn and Schunk, 1981; Isambert et al., 2008). The aim is to retrieve the velocity field $\mathbf{w}(\mathbf{x},t)$ visualised by a discrete image sequence $\mathbf{I} = \{\mathbf{I}^z\}_{z=1...Z} = \{\mathbf{I}(\mathbf{x},t_z)\}_{z=1...Z}$. The application of data assimilation techniques to motion estimation also emerged a few years ago (Papadakis et al., 2007; Titaud et al., 2010; Béréziat and Herlin, 2011). In the case of motion estimation, these techniques aim to find the optimal solution to the equations describing the temporal evolution of motion fields and images and to the observation equation, which links the motion field to the observed image data. Their major drawbacks are the memory and computer resources required that do not allow to process long temporal sequences of large size images. To get round this problem, reduction methods are required to apply the data assimilation on subspaces. In (Drifi and Herlin, 2011), such reduced model has been proposed. Coefficients characterizing image observations in the image subspace are assimilated in the reduced model to estimate those characterizing the motion field.

In this paper, we focus on the estimation of motion on long temporal windows of satellite images acquired over oceans. The image sequence is split into small windows that half overlap in time. A "full model" is chosen in order to approximately describe the dynamics of motion fields and images. Image assimilation in the full model is applied on the first window to retrieve its motion field. A learning process is designed that defines a "reduced model" from the full model in the second window. This learning defines the subspaces used to characterize motion and images and applies the Galerkin projection of the full model on these subspaces. Data assimilation in the reduced model is then applied for this second window. The learning method is iterated on the next window until the whole image sequence has been processed.

The paper describes the two main components of the learning method on sliding windows: the full model and its image assimilation process, the learning of reduced models and their data assimilation systems.

Oceans are incompressible fluids and the 2D incompressible hypothesis still remains a good approximation for image sequences if no or small vertical motion occurs (no upwelling or downwelling). If the motion field is divergence-free ($\mathrm{div}(\mathbf{w}) = 0$), it is then only characterized by its vorticity ξ, according to the Helmholtz orthogonal decomposition (Deriaz and Perrier, 2006). An equation on the dynamics of vorticity ξ is then included in the full model. As temporal integration of the vorticity requires the knowledge of the velocity value at each time step, the discrete computation of $\mathbf{w}$ from ξ is performed, based on an algebraic decomposition of vorticity. The transport of image brightness by velocity, which is the usual optical flow equation, is chosen to describe the image dynamics.

Section 2 describes the divergence-free image model used for motion estimation on an image sequence. The algebraic method that computes $\mathbf{w}$ from its vorticity ξ is also given. Section 3 explains how the solution is obtained by minimizing a cost function with a strong 4D-Var (no error on the dynamics) data assimilation method. The derivation of a reduced model by the Galerkin projection is provided in Section 4. The learning method used to process long temporal image sequences is fully described in Section 5. Section 6 provides results on synthetic data for the full model and Section 7 for the learning method on a long temporal window.

2 Definition of the full model

This section describes the divergence-free model that is used to determine velocity from images, on the pixel grid, on the first window of the long temporal sequence.

2.1 Divergence-free model

Vorticity characterizes a rotational motion while divergence characterizes sinks and sources in a flow. A fluid motion $\mathbf{w} = (u \quad v)^T$ is described by its vorticity $\xi = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$, under the hypothesis of null divergence (Deriaz and Perrier, 2006). ξ is chosen as the first component of the state vector $\mathbf{X}$ of the full model. Deriving the evolution law for ξ requires heuristics on the velocity $\mathbf{w}$. The Lagrangian constancy hypothesis, $\frac{d\mathbf{w}}{dt} = 0$, is considered in the paper that can be expanded as $\frac{\partial \mathbf{w}}{\partial t} + (\mathbf{w}.\nabla)\mathbf{w} = 0$, or:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = 0 \tag{1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = 0 \tag{2}$$

Let us compute the y-derivative of Eq. (1) and subtract it from the x-derivative of Eq. (2), replace the quantity $\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$ by the vorticity ξ, and we obtain:

$$\frac{\partial \xi}{\partial t} + u\frac{\partial \xi}{\partial x} + v\frac{\partial \xi}{\partial y} + \xi\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) = 0 \tag{3}$$

This is rewritten in a conservative form as:

$$\frac{\partial \xi}{\partial t} + \nabla.(\xi\mathbf{w}) = 0 \tag{4}$$

The observations that are used for the data assimilation process are images acquired by satellites. The second component of the state vector is chosen as a pseudo-image $\mathbf{I}_s$, which has the same dynamics than the image observation. It is included in the state vector in order to allow an easy comparison with the image observations at each acquisition date: they have to be almost identical. The evolution law chosen for $\mathbf{I}_s$ verifies the heuristics for the transport of images by velocities: this is the well known Optical Flow Constraint Equation (Horn and Schunk, 1981) expressed as:

$$\frac{\partial \mathbf{I}_s}{\partial t} + \nabla \mathbf{I}_s . \mathbf{w} = 0 \tag{5}$$

or with the divergence-free hypothesis:

$$\frac{\partial \mathbf{I}_s}{\partial t} + \nabla . (\mathbf{I}_s \mathbf{w}) = 0 \tag{6}$$

The divergence-free model is then defined by the state vector $\mathbf{X} = (\xi \quad \mathbf{I}_s)^T$ and its evolution system:

$$\frac{\partial \xi}{\partial t} + \nabla . (\xi \mathbf{w}) = 0 \tag{7}$$

$$\frac{\partial \mathbf{I}_s}{\partial t} + \nabla . (\mathbf{I}_s \mathbf{w}) = 0 \tag{8}$$

2.2 Algebraic computation of w

When the state vector is integrated in time from an initial condition, using Eqs. (7,8), the knowledge of ξ, $\mathbf{I}_s$ and $\mathbf{w}$ is required. The velocity field $\mathbf{w}$ should then be computed from the scalar field ξ as follow. A stream function φ is first defined as the solution of the Poisson equation:

$$-\Delta \varphi = \xi \tag{9}$$

Then, $\mathbf{w}$ is derived from φ:

$$\mathbf{w} = \begin{pmatrix} \frac{\partial \varphi}{\partial y} & -\frac{\partial \varphi}{\partial x} \end{pmatrix}^T \tag{10}$$

In the literature, Eq. (9) is usually solved in Fourier domain, with periodic boundary conditions. An algebraic solution is proposed in order to allow Dirichlet boundary conditions. An eigenfunction, ϕ, of the linear operator $-\Delta$ has to verify $-\Delta\phi = \lambda\phi$ with λ the associated eigenvalue. Explicit solutions of this eigenvalue problem are the family of bi-periodic functions $\phi_{n,m}(x,y) = \sin(\pi n x)\sin(\pi m y)$ with the associated eigenvalues $\lambda_{n,m} = \pi^2 n^2 + \pi^2 m^2$. These functions form an orthogonal basis of a subspace of $L^2(\Omega)$, space of square-integrable functions defined on the spatial domain Ω. Let $(a_{n,m})$ be the coefficients of ξ in the basis $(\phi_{n,m})$. We have $\xi(x,y) = \sum_{n,m} a_{n,m}\phi_{n,m}(x,y)$. It comes:

$$\varphi(x,y) = \sum_{n,m} \frac{a_{n,m}}{\lambda_{n,m}} \phi_{n,m}(x,y) \tag{11}$$

We verify:

$$-\Delta\varphi(x,y) = -\sum_{n,m} \frac{a_{n,m}}{\lambda_{n,m}} \Delta\phi_{n,m}(x,y) = \sum_{n,m} \frac{a_{n,m}}{\lambda_{n,m}} \lambda_{n,m}\phi_{n,m}(x,y) = \xi$$

At each time step, having knowledge of ξ and $(\phi_{n,m})$, the values of $(a_{n,m})$ are first computed. Then φ is derived by Eq. (11), using the $(\lambda_{n,m})$ values, and $\mathbf{w}$ by Eq. (10).

3 Strong 4D-Var Data Assimilation

Image assimilation is applied on the first window of the long sequence with the full model described in Section 2.

We consider the state vector $\mathbf{X}(x,y,t) = (\xi(x,y,t) \quad \mathbf{I}_s(x,y,t))^T$ defined on the space-time domain $\Omega \times [0,t_N]$. In order to determine $\mathbf{X}$ on this domain, the 4D-Var framework considers a system of three equations to be solved.

The first equation describes the evolution in time of the state vector $\mathbf{X}$. This is given by Eqs. (7,8). For sake of simplicity, we summarize the system and introduce the evolution model M for the state vector $\mathbf{X}$:

$$\frac{\partial \mathbf{X}}{\partial t} + M(\mathbf{X}) = 0 \tag{12}$$

We consider having some knowledge of the state vector value at initial date 0 which is described by the background value $\mathbf{X}_b(x,y)$. As this initial condition is uncertain, the second equation of the system involves an error term:

$$\mathbf{X}(x,y,0) = \mathbf{X}_b(x,y) + \varepsilon_B(x,y) \tag{13}$$

The error $\varepsilon_B(x,y)$ is supposed Gaussian and characterized by its covariance matrix $B(x,y)$.

The last equation, named observation equation, links the state vector to the image observations $\mathbf{I}(x,y,t)$. It is expressed as:

$$\mathbf{I}(x,y,t) = H(\mathbf{X}(x,y,t)) + \varepsilon_R(x,y,t) \tag{14}$$

with H the observation operator. As the component $\mathbf{I}_s$ is directly comparable to the observations, the operator H reduces to a projection: $H(\mathbf{X}) = H\mathbf{X} = \mathbf{I}_s$. Image acquisitions are noisy and their underlying dynamics could be different from the one described by Eq. (8). An observation error, ε_R, is used to model these uncertainties. It is supposed Gaussian and characterized by its covariance matrix $R(x,y,t)$.

For discussing how Eqs. (12,13,14) are solved by the data assimilation method, the state vector and its evolution equation are first discretized in time with an Euler scheme. The space variables x and y are omitted for sake of simplicity. Let dt be the time step, the state vector at discrete index k, $0 \leq k \leq N_t$, is denoted $\mathbf{X}(k) = \mathbf{X}(k \times dt)$. The discrete evolution equation is:

$$\mathbf{X}(k+1) = \mathbf{X}(k) - dtM(\mathbf{X}(k)) = Z_k(\mathbf{X}(k)) \tag{15}$$

with $Z_k(\mathbf{X}(k)) = (\xi(k) - dt\nabla.(\xi(k)\mathbf{w}(\xi(k))) \quad \mathbf{I}_s(k) - dt\nabla.(\mathbf{I}_s(k)\mathbf{w}(\xi(k))))^T$. We assume that N_{obs} image observations $\mathbf{I}(t_i)$ are acquired at indexes $t_1 < \cdots < t_i < \cdots < t_{N_{\text{obs}}}$. Looking for $\mathbf{X} = (\mathbf{X}(0), \cdots, \mathbf{X}(N_t))$ solving Eqs.(15,13,14) is expressed as a constrained optimization problem: the cost function

$$\begin{aligned} J(\mathbf{X}(0)) = & \frac{1}{2}\int_\Omega (\mathbf{X}(0) - \mathbf{X}_b)^T B^{-1} (\mathbf{X}(0) - \mathbf{X}_b) dxdy \\ & + \frac{1}{2}\sum_{i=1}^{N_{\text{obs}}} \int_\Omega (H\mathbf{X}(t_i) - \mathbf{I}(t_i))^T R^{-1}(t_i)(H\mathbf{X}(t_i) - \mathbf{I}(t_i)) dxdy \end{aligned} \tag{16}$$

has to be minimized under the constraint of Eq. (15). The first term of J comes from Eq. (13). The second term of J comes from Eq. (14), which is valid at observation indexes t_i.

The gradient of J is obtained from the directional derivative of J and from the definition of an auxiliary variable λ that verifies the backward equation:

$$\lambda(k) = \left(\frac{\partial Z_k}{\partial \mathbf{X}}\right)^* \lambda(k+1) + H^T R^{-1}(k)\left(H\mathbf{X}(k) - \mathbf{I}(k)\right)$$

with $\lambda(N_t) = 0$, the term $H^T R^{-1}(k)(H\mathbf{X}(k) - \mathbf{I}(k))$ being only taken into account at observation indexes t_i. It can be proven ((Le Dimet and Talagrand, 1986)) that the gradient reduces to:

$$\nabla J_{\mathbf{X}(0)} = B^{-1}(\mathbf{X}(0) - \mathbf{X}_b) + \lambda(0)$$

The cost function J is minimized using an iterative steepest descent method. At each iteration, the forward time integration of $\mathbf{X}$ is performed which provides J, then a backward integration of λ computes $\lambda(0)$ and provides ∇J. An efficient solver (Zhu et al., 1994) is used to perform the steepest descent given J and ∇J.

4 Derivation of a reduced model

This section explains the derivation by Galerkin projection of a reduced model from the full model described in Section 2.

We assume that we have knowledge of the background value ξ_b of vorticity at the beginning of the studied temporal window. The first issue is to define subspaces for vorticity fields and images, onto which the evolution equations (7) and (8) are projected. These subspaces are defined by their respective orthogonal basis Ψ_ξ and $\Psi_\mathbf{I}$. First, a Proper Orthogonal Decomposition transform (POD) is applied to the image observations $\mathbf{I} = \{\mathbf{I}^z\}_{z=1\ldots Z}$ that defines $\Psi'_\mathbf{I}$. Second, ξ_b is numerically integrated in time with Eq. (7). It provides snapshots, on which POD is applied to obtain Ψ'_ξ. We keep the first K modes of Ψ'_ξ and the first L modes of $\Psi'_\mathbf{I}$ to obtain Ψ_ξ and $\Psi_\mathbf{I}$.

Let $a_i(t)$ and $b_j(t)$ be the projection coefficients of $\xi(\mathbf{x},t)$ and $\mathbf{I}_s(\mathbf{x},t)$ on Ψ_ξ and $\Psi_\mathbf{I}$. $\xi(\mathbf{x},t)$ and $\mathbf{I}_s(\mathbf{x},t)$ are then approximated by:

$$\xi(\mathbf{x},t) \approx \sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x}), \tag{17}$$

$$\mathbf{I}_s(\mathbf{x},t) \approx \sum_{j=1}^{L} b_j(t)\psi_{\mathbf{I},j}(\mathbf{x}), \tag{18}$$

and replaced in Eqs. (7) and (8):

$$\sum_{i=1}^{K} \frac{da_i}{dt}(t)\psi_{\xi,i}(\mathbf{x}) + \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right)\cdot\nabla\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right) = 0 \tag{19}$$

$$\sum_{i=1}^{L} \frac{db_i}{dt}(t)\psi_{\mathbf{I},j}(\mathbf{x}) + \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right)\cdot\nabla\left(\sum_{j=1}^{L} b_j(t)\psi_{\mathbf{I},j}(\mathbf{x})\right) = 0 \tag{20}$$

This system is projected on Ψ_ξ and $\Psi_\mathbf{I}$:

$$\frac{da_k}{dt}(t)\left\langle\psi_{\xi,k},\psi_{\xi,k}\right\rangle + \left\langle \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right)\cdot\nabla\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right),\psi_{\xi,k}\right\rangle = 0, \tag{21}$$

$$\frac{db_l}{dt}(t)\left\langle\psi_{\mathbf{I},l},\psi_{\mathbf{I},l}\right\rangle + \left\langle \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right)\cdot\nabla\left(\sum_{j=1}^{L} b_j(t)\psi_{\mathbf{I},j}\right),\psi_{\mathbf{I},l}\right\rangle = 0, \tag{22}$$

with $\langle .,. \rangle$ being the scalar product in the $L^2(\Omega)$ space:

$$\langle f, g \rangle = \int_\Omega f(\mathbf{x}) g(\mathbf{x}) d\mathbf{x}. \tag{23}$$

System (21,22) is simplified to get:

$$\frac{da_k}{dt}(t) + a^T(t) B(k) a(t) = 0, \quad k = 1 \dots K. \tag{24}$$

$$\frac{db_l}{dt}(t) + a^T(t) G(l) b(t) = 0, \quad l = 1 \dots L. \tag{25}$$

with:

- $a(t) = (a_1(t) \quad \dots \quad a_K(t))^T$,
- $b(t) = (b_1(t) \quad \dots \quad b_L(t))^T$,
- $B(k)$ a $K \times K$ matrix :
  $B(k)_{i,j} = \dfrac{\left\langle \mathbf{w}(\psi_{\xi,i}) \cdot \nabla \psi_{\xi,j}, \psi_{\xi,k} \right\rangle}{\left\langle \psi_{\xi,k}, \psi_{\xi,k} \right\rangle}$,
- $G(l)$ a $K \times L$ matrix :
  $G(l)_{i,j} = \dfrac{\left\langle \mathbf{w}(\psi_{\xi,i}) \cdot \nabla \psi_{\mathbf{I},j}, \psi_{\mathbf{I},l} \right\rangle}{\left\langle \psi_{\mathbf{I},l}, \psi_{\mathbf{I},l} \right\rangle}$

Let $\mathbf{X}_R(\mathbf{x},t) = (a(t) \quad b(t))^T$ be the state vector of the reduced model. System (24,25) is rewritten as:

$$\frac{d\mathbf{X}_R}{dt} + M_R(\mathbf{X}_R) = 0 \tag{26}$$

M_R being the Galerkin projection of the full model M on Ψ_ξ and $\Psi_\mathbf{I}$.

5 Learning reduced models on sliding windows

This section describes the learning method on sliding windows, with the full model of Section 2 applied on the first window and the reduced models of Section 4 applied on the following. This learning method allows to process long temporal image sequences.

The discrete sequence $\mathbf{I} = \{\mathbf{I}^z\}_{z=1\dots Z}$ is first split into short temporal windows, with 4 to 6 images, that half overlap in time. These windows are denoted Wi_m, with m the index.

Images belonging to Wi_1 are assimilated in the divergence-free model described in Section 2. This allows the retrieval of the vorticity on Wi_1.

The retrieved value at the beginning of Wi_2 is taken as background vorticity ξ_b required to learn the reduced model on Wi_2, as it has been explained in Section 4. The coefficients of projection of images belonging to Wi_2 are assimilated in the reduced model to retrieve the vorticity coefficients and compute the vorticity values and motion fields over Wi_2.

This again provides ξ_b for Wi_3 and allows to learn the reduced model on Wi_3. The process is then iterated until the whole sequence $\mathbf{I}$ has been analyzed.

The method is summarized in Figure 1.

The major advantage is that full assimilation is only applied on the first temporal window Wi_1 that has a short duration. It requires, at each iteration of the optimisation

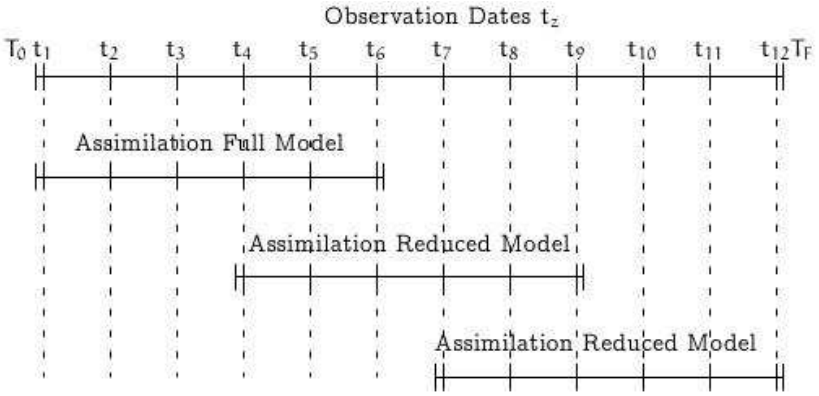


Figure 1: Learning reduced models on sliding windows.

process, a forward integration of M and a backward integration of its adjoint (Béréziat and Herlin, 2011). The complexity is proportional to the image size multiplied by the number of time steps in the assimilation window. On the next window Wi_m, the complexity greatly decreases as the state vector involved in the reduced models M_R is of size $K+L$, which is less than 10 in the experiments.

6 Results of the full model

In order to quantify the method, it is applied on synthetic data produced by twin experiments.

A sequence of five synthetic observations (see Figure 3) is obtained by time integration of the divergence-free model from the initial conditions displayed in Figure 2.

For the assimilation experiment, the background of vorticity is set to zero and the

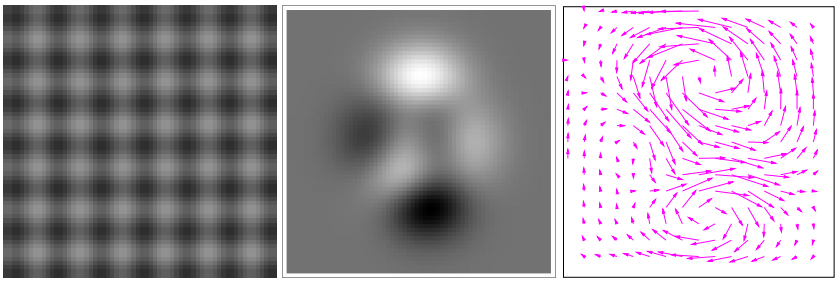


Figure 2: Pseudo-image, vorticity (positive values are drawn in white, negative ones in black) and motion field at $t = 0$.

one of pseudo-image is the first observation. The result of the assimilation process is the state vector $\mathbf{X}(k) = (\xi(k) \quad \mathbf{I}_s(k))^T$ and its associated motion vector $\mathbf{w}(k)$ over the discrete assimilation window. In Table 1, the error between the motion result and the ground truth is given for our approach and four state-of-the-art image processing methods: (Horn and Schunk, 1981; Isambert et al., 2007; Corpetti et al., 2002; Suter, 1994) that use either a L_2 regularization of motion (Horn and Schunk, 1981) or a second or-

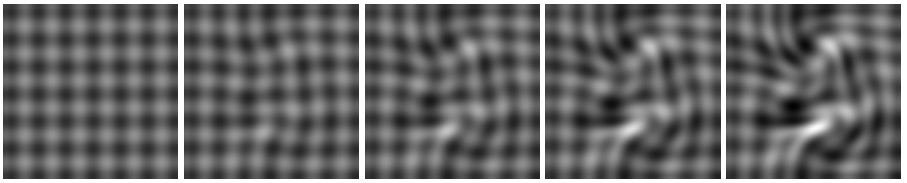


Figure 3: Observations.

der regularization on the divergence (Isambert et al., 2007; Corpetti et al., 2002; Suter, 1994).

Method	Angular error (in deg.)				Norm error (in %)		
	Mean	Std. Dev.	Min	Max	Mean	Min	Max
(Horn and Schunk, 1981)	15.26	9.65	0.33	67.12	24.98	0.85	93.10
(Corpetti et al., 2002)	12.54	9.49	0.17	68.49	20.03	0.51	87.74
(Suter, 1994)	10.41	5.34	0.06	35.58	18.07	0.09	92.31
(Isambert et al., 2007)	10.61	6.92	0.00	56.62	18.01	0.01	97.74
Our approach	0.18	0.10	0.00	0.572	0.41	0.00	19.47

Table 1: Error analysis: misfit between motion results and ground truth.

This demonstrates that our approach is almost exact for this twin experiment.

7 Results of the learning method on sliding windows

Twin experiments were also designed to quantify the learning method on sliding windows and its benefit for motion estimation on long temporal image sequences.

The full model was used, with initial conditions displayed in Figure 4. Snapshots of $\mathbf{I}_s$ were taken to create the observation images $\mathbf{I} = \{\mathbf{I}^z\}_{z=1...Z}$. Assimilation of these data in the full and reduced models is then applied as described in Section 5 on six windows. Results on motion estimation are given in Figure 5 and compared with the ground truth provided by the simulation creating the observations. Each column corresponds to the first frame of one of the six windows Wi_m.

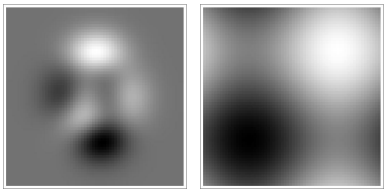


Figure 4: Initialisation for the twin experiment. $\xi(0)$ on the left and $\mathbf{I}_s(0)$ on the right.

In order to demonstrate the potential of the learning method on sliding windows, statistics on the retrieved vorticity are provided. The normalized root mean square error (in percentage) ranges from 1.1 to 4.0% from the first to the sixth window, while the correlation value between the retrieved vorticity and the ground truth decreases from 0.99 to 0.96.

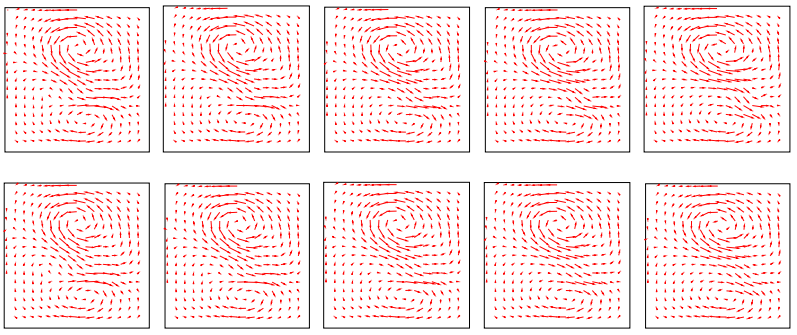


Figure 5: Estimated Motion (first line) compared to the ground truth (second line).

The computing time reduces from around 4 hours for the first window processed by the full model to less than 1 minute for the next five one, processed by reduced models.

8 Conclusions

In the paper, we proposed a learning method on sliding windows for estimating motion on long temporal image sequences with data assimilation techniques. This method couples full and reduced models obtained by Galerkin projection and allows to process images in quasi-real time. The method has been quantified with twin experiments to demonstrate its potential. First, the quality of motion fields retrieved by the full model has been assessed. Second, statistics on performances of the reduced models learned on the sliding windows have been provided.

One perspective is to replace the POD bases Ψ_ξ which were used to define the reduced models by a fixed basis in order to even reduce the computational requirements on the first part of the image sequence.

Acknowledgements

This research is partially supported by the Geo-FLUIDS project (ANR 09 SYSC 005 02).

References

Béréziat, D. and Herlin, I. (2011). Solving ill-posed image processing problems using data assimilation. *Numerical Algorithms*, 56(2):219–252.

Corpetti, T., Mémin, E., and Pérez, P. (2002). Dense estimation of fluid flows. *Pat. Anal. and Mach. Int.*, 24(3):365–380.

Deriaz, E. and Perrier, V. (2006). Divergence-free and curl-free wavelets in two dimensions and three dimensions: application to turbulent flows. *Journal of Turbulence*, 7(3):1–37.

Drifi, K. and Herlin, I. (2011). Assimilation d'images dans un modèle réduit pour l'estimation du mouvement. In *GRETSI 2011 - Groupe d'Etudes du Traitement du Signal et des Images*, Bordeaux, France.

Horn, B. and Schunk, B. (1981). Determining optical flow. *Art. Int.*, 17:185–203.

Isambert, T., Berroir, J., and Herlin, I. (2008). A multiscale vector spline method for estimating the fluids motion on satellite images. In *ECCV*, Marseille, France. Springer.

Isambert, T., Herlin, I., and Berroir, J.-P. (2007). Fast and stable vector spline method for fluid flow estimation. In *ICIP*, pages 505–508.

Le Dimet, F.-X. and Talagrand, O. (1986). Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus*, 38A:97–110.

Papadakis, N., Corpetti, T., and Mémin, E. (2007). Dynamically consistent optical flow estimation. In *ICCV*, pages 1–7.

Suter, D. (1994). Motion estimation and vector splines. In *CVPR*, pages 939–942.

Titaud, O., Vidard, A., Souopgui, I., and Dimet, F.-X. L. (2010). Assimilation of image sequences in numerical models. *Tellus A*, 62:30–47.

Zhu, C., Byrd, R., Lu, P., and Nocedal, J. (1994). L-BFGS-B: a limited memory FORTRAN code for solving bound constrained optimization problems. Technical Report NAM-11, EECS Department, Northwestern University.

D.3 Complément au chapitre 3 : Learning reduced models for motion estimation on long temporal image sequences.

Cet article [21] a été publié dans la conférence Igarss 2012 [3] (IEEE International Geoscience and Remote Sensing Symposium). La conférence s'est déroulée à Munich en Allemagne du 22 au 27 juillet 2012. Les travaux présentés dans cette publication sont décrits dans le chapitre 3.

3. www.igarss2012.org

LEARNING REDUCED MODELS FOR MOTION ESTIMATION ON LONG TEMPORAL IMAGE SEQUENCES

Isabelle Herlin[1,2]*, Karim Drifi*[1,2]

[1]INRIA, Institut National de Recherche en Informatique et Automatique,
Domaine de Voluceau, Rocquencourt - BP 105, 78153 Le Chesnay Cédex, France

[2]CEREA, joint laboratory ENPC - EDF R&D,
Université Paris-Est, 6-8 avenue Blaise Pascal,
Cité Descartes, Champs-sur-Marne, 77455 Marne la Vallée Cédex 2, France

ABSTRACT

Index Terms— Motion, Model Reduction, Principal Order Decomposition, Model Coupling, Galerkin Projection

1. INTRODUCTION

Motion estimation from an image sequence has been intensively studied since the beginning of image processing [1, 2]. The aim is to retrieve the velocity field $\mathbf{w}(\mathbf{x}, t)$ visualized by a discrete image sequence $I = \{I^z\}_{z=1...Z} = \{I(\mathbf{x}, t_z)\}_{z=1...Z}$. The application of data assimilation techniques to motion estimation also emerged in the last five years [3, 4, 5]. Data assimilation aims to find the optimal solution to the equations describing the temporal evolution of motion and image fields and to that, named observation equation, which links the motion field to the observed image acquisitions. Its advantage, compared to image processing methods, is the retrieval of a continuous motion field, allowing to assess the processes occurring between two acquisitions. Its major drawbacks are the memory and computer resources required when using the full model, defined on the pixel grid. This does not allow to process long temporal sequences of large size images such as long-term satellite data. To get round this problem, reduction methods are required to apply data assimilation on subspaces. In [6], such reduced model is proposed to describe the dynamics of motion and image fields. Coefficients characterizing observations in the image subspace are then assimilated in the reduced model to estimate those characterizing the motion field.

In this paper, we focus on the estimation of motion on a long temporal window. A method, named sliding windows assimilation, is designed: image assimilation is first applied on an initial sub-window with the pixel grid, and then on sliding temporal windows with reduced models, obtained by Galerkin projection on motion and image subspaces. The full model, used on the pixel grid in this paper, is described in Section 2. The main concepts of data assimilation are briefly summarized in the same section. The derivation of the reduced model by Galerkin projection is then explained in Section 3. Last, the sliding windows assimilation method is fully described in Section 4 and its results quantified in Section 5.

2. FULL MODEL DESCRIPTION AND DATA ASSIMILATION

In order to describe the sliding windows assimilation method, we consider, as an example, divergence-free motion fields $\mathbf{w}(\mathbf{x}, t)$, with $\mathbf{w} = \begin{pmatrix} u & v \end{pmatrix}^T$, $\mathbf{x} = \begin{pmatrix} x & y \end{pmatrix}^T \in \Omega$, a bounded domain, and $t \in [t_0, t_N]$, a closed interval. However, the discussion would be still valid for other motion dynamics. As a divergence-free motion field $\mathbf{w}(\mathbf{x}, t)$ is fully characterized by its vorticity value $\xi(\mathbf{x}, t)$, $\mathbf{w}$ is a function of ξ, and written $\mathbf{w}(\xi)$. We also assume that the motion field satisfies the heuristics of Lagrangian constancy described by: $\frac{d\mathbf{w}}{dt} = \frac{\partial \mathbf{w}}{\partial t} + (\mathbf{w}^T \nabla)\mathbf{w} = 0$. Under the divergence-free assumption, this is rewritten as an equation describing the evolution of ξ:

$$\frac{\partial \xi}{\partial t} + \mathbf{w}(\xi) \cdot \nabla \xi = 0. \tag{1}$$

The state vector of the full model is defined as the function: $\mathbf{X}(\mathbf{x}, t) = \begin{pmatrix} \xi(\mathbf{x}, t) & I_s(\mathbf{x}, t) \end{pmatrix}^T$. I_s is a pseudo-image, which has the same dynamics as the image observation: the motion field transports it according to the so-called optical flow equation [1]:

$$\frac{\partial I_s}{\partial t} + \mathbf{w}(\xi) \cdot \nabla I_s = 0. \tag{2}$$

This pseudo-image I_s is included in the state vector in order to allow an easy comparison with the image observations at each acquisition date: they have to be almost identical.

Eqs. 1 and 2 are summarized as:

$$\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}(\mathbf{X}) = 0 \tag{3}$$

Data assimilation aims to find an optimal solution to the evolution equation (Eq. 3) and to the observation equation that links the state vector to the image observations $I(\mathbf{x}, t)$:

$$\mathbb{H}\mathbf{X} = I \tag{4}$$

The observation operator $\mathbb{H}$ projects the state vector into the space of observations. In our case, $\mathbb{H}$ reduces to a projection on the pseudo-image component, $\mathbb{H}\mathbf{X} = I_s$, and Eq. 4 rewrites as: $I_s = I$.

Image observations are assimilated in the full model $\mathbb{M}$, defined by Eqs. 1 and 2, to estimate vorticity and motion. Full description of the data assimilation method is given in [5].

3. GALERKIN PROJECTION

The aim is to define a reduced model of the full model described in Section 2. We assume that we have knowledge of a rough estimation ξ_b, named background value, of vorticity at the beginning of the temporal window. The first issue is to define subspaces for vorticity and image fields, onto which the evolution equations 1 and 2 are respectively projected to obtain the reduced model. These subspaces are defined by their orthogonal basis Ψ_ξ and Ψ_I, obtained as explained below. First, a Proper Orthogonal Decomposition transform (POD) [7] is applied to the image observations $I = \{I^z\}_{z=1\ldots Z}$, that defines Ψ_I^f. Second, ξ_b is integrated in time with Eq. 1. This simulation provides snapshots, on which POD is applied to obtain Ψ_ξ^f. We keep the first K modes of Ψ_ξ^f and the first L modes of Ψ_I^f to define the reduced basis Ψ_ξ and Ψ_I. K and L are chosen in order to keep 95% of variance.

Let $a_i(t)$ and $b_j(t)$ be the projection coefficients of $\xi(\mathbf{x}, t)$ and $I_s(\mathbf{x}, t)$ on Ψ_ξ and Ψ_I. $\xi(\mathbf{x}, t)$ and $I_s(\mathbf{x}, t)$ are approximated by: $\xi(\mathbf{x}, t) \approx \sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})$ and $I_s(\mathbf{x}, t) \approx \sum_{j=1}^{L} b_j(t)\psi_{I,j}(\mathbf{x})$. After replacing in Eqs. 1 and 2, it comes:

$$\left\{\begin{array}{l} \displaystyle\sum_{i=1}^{K} \frac{da_i}{dt}(t)\psi_{\xi,i}(\mathbf{x}) + \\ \mathbf{w}\left(\displaystyle\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right) \cdot \nabla \left(\displaystyle\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right) = 0 \\ \displaystyle\sum_{j=1}^{L} \frac{db_j}{dt}(t)\psi_{I,j}(\mathbf{x}) + \\ \mathbf{w}\left(\displaystyle\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}(\mathbf{x})\right) \cdot \nabla \left(\displaystyle\sum_{j=1}^{L} b_j(t)\psi_{I,j}(\mathbf{x})\right) = 0 \end{array}\right. \tag{5}$$

This system is projected on each vector of Ψ_ξ and Ψ_I to derive:

$$\left\{\begin{array}{l} \dfrac{da_k}{dt}(t)\,\langle \psi_{\xi,k}, \psi_{\xi,k}\rangle + \\ \left\langle \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right) \cdot \nabla \left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,j}\right), \psi_{\xi,k} \right\rangle = 0, \\ \dfrac{db_l}{dt}(t)\,\langle \psi_{I,l}, \psi_{I,l}\rangle + \\ \left\langle \mathbf{w}\left(\sum_{i=1}^{K} a_i(t)\psi_{\xi,i}\right) \cdot \nabla \left(\sum_{j=1}^{L} b_j(t)\psi_{I,j}\right), \psi_{I,l} \right\rangle = 0, \end{array}\right. \tag{6}$$

$\langle ., . \rangle$ being the scalar product: $\langle f, g\rangle = \int f(\mathbf{x})g(\mathbf{x})d\mathbf{x}$.

As $\mathbf{w}$ is a linear function of the vorticity ξ, System (6) is simplified as:

$$\left\{\begin{array}{l} \dfrac{da_k}{dt}(t) + a^T(t)B(k)a(t) = 0, \quad k = 1\ldots K. \\ \dfrac{db_l}{dt}(t) + a^T(t)G(l)b(t) = 0, \quad l = 1\ldots L. \end{array}\right. \tag{7}$$

with:

- $a(t) = \begin{pmatrix} a_1(t) & \ldots & a_K(t) \end{pmatrix}^T$ and $b(t) = \begin{pmatrix} b_1(t) & \ldots & b_L(t) \end{pmatrix}^T$
- $B(k)$ a $K \times K$ matrix: $B(k)_{i,j} = \dfrac{\langle \mathbf{w}(\psi_{\xi,i}) \cdot \nabla \psi_{\xi,j}, \psi_{\xi,k}\rangle}{\langle \psi_{\xi,k}, \psi_{\xi,k}\rangle}$,
- $G(l)$ a $K \times L$ matrix: $G(l)_{i,j} = \dfrac{\langle \mathbf{w}(\psi_{\xi,i}) \cdot \nabla \psi_{I,j}, \psi_{I,l}\rangle}{\langle \psi_{I,l}, \psi_{I,l}\rangle}$

Let $\mathbf{X}_R(\mathbf{x}, t) = \begin{pmatrix} a(t)^T & b(t)^T \end{pmatrix}^T$ be the state vector of the reduced model. System (7) is rewritten as:

$$\frac{d\mathbf{X}_R}{dt} + \mathbb{M}_R(\mathbf{X}_R) = 0, \tag{8}$$

$\mathbb{M}_R$ being the projection [8] of the full model on Ψ_ξ and Ψ_I.

The same data assimilation method than that described in Section 2 is then applied. The state vector is $\mathbf{X}_R$, the evolution equation Eq. 3 is replaced by Eq. 8, and the components $b_j(t)$ of the state vector are compared with the projections of image acquisitions on $\psi_{I,j}$ in order to implement Eq. 4.

4. SLIDING WINDOWS ASSIMILATION METHOD

This section describes the sliding windows assimilation method, which is defined to process long temporal image sequences.

The discrete sequence $I = \{I^z\}_{z=1\ldots Z}$ is first split into short sub-sequences, that half overlap in time. The corresponding temporal intervals or windows are denoted Wi_m, with m being their index. Images belonging to Wi_1 are assimilated in the full model $\mathbb{M}$ described in Section 2. This allows the retrieval of the vorticity on the whole interval Wi_1. Its value at the beginning of Wi_2 is taken as the background value ξ_b, required to define the reduced model $\mathbb{M}_R^2$ on Wi_2,

as explained in Section 3. The coefficients of projection $b_j(t)$, of images belonging to Wi_2, are assimilated in this reduced model in order to retrieve the vorticity coefficients $a_i(t)$, and compute the vorticity values $\xi(t)$ over Wi_2. This again provides the background value of vorticity for Wi_3 and allows to define the reduced model $\mathbb{M}_R^3$ on the third window. The process is then iterated, until the whole image sequence I has been analyzed.

The major advantage is that full assimilation is only applied on the first temporal window Wi_1, that has a short duration. This assimilation is based on an optimization method [9] and requires, at each iteration, a forward integration of $\mathbb{M}$ and a backward integration of its adjoint [5]. The complexity is proportional to the image size multiplied by the number of time steps in the assimilation window. On the next windows Wi_m, the complexity greatly decreases as the state vector involved in the reduced models is of size $K + L$, which is less than 10 in the experiments.

5. RESULTS

Twin experiments were designed to quantify the sliding windows assimilation method and its ability to process long-term satellite data.

On a first experiment, the full model is run to obtain a four-day simulation, with a time step of 15 minutes and initial conditions displayed on Fig. 1. Snapshots of I_s are taken as observation images $I = \{I^z\}_{z=1...Z}$. Four of them are displayed on Fig. 2. The whole temporal window corresponds

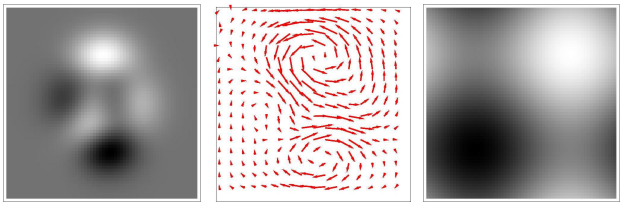
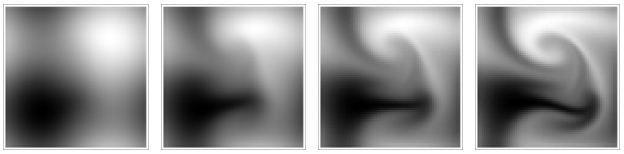


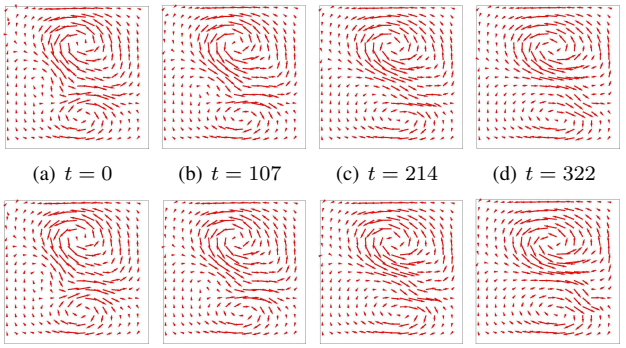
Fig. 1. Left to right: Initial condition $\xi(0)$, $\mathbf{w}(0)$and $I_s(0)$.



(a) $t = 21$ (b) $t = 121$ (c) $t = 221$ (d) $t = 321$

Fig. 2. Observations at temporal indexes 21, 121, 221, 321.

to indices from 0 to 322. It is split in subwindows and assimilation is performed, as described in Section 4, with the sliding windows assimilation method. Results are displayed on Fig. 3. In order to demonstrate the potential of this sliding windows method, we made statistics on the quality of the



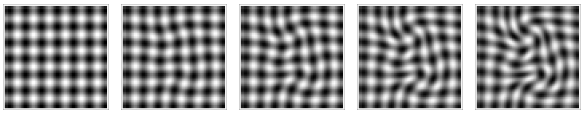
(a) $t = 0$ (b) $t = 107$ (c) $t = 214$ (d) $t = 322$

(e) $t = 0$ (f) $t = 107$ (g) $t = 214$ (h) $t = 322$

Fig. 3. Up: ground truth, Down: estimation with reduced model at temporal indexes 0, 107, 214, 322.

retrieved vorticity on the whole sequence. The NRMSE (in percentage) ranges from 1.1 to 4.0% from the first to the sixth sub-window. The correlation value between the retrieved vorticity and the ground truth decreases from 0.99 to 0.96 from the first to the last sub-window.

A second experiment modifies the initial condition on the pseudo-image, in the simulation providing the snapshots. The full and POD models are quantified on the sub-window with indexes from 0 to 82. Observations are considered at indexes 1, 21, 41, 61, 81, and displayed on Fig. 4. Ground truth

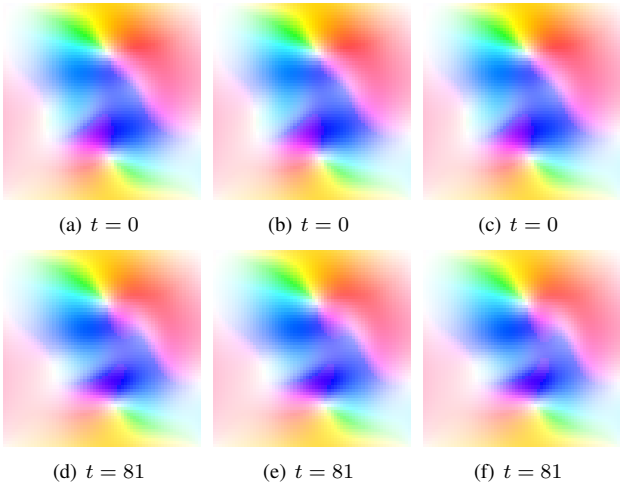


(a) $t = 1$ (b) $t = 21$ (c) $t = 41$ (d) $t = 61$ (e) $t = 81$

Fig. 4. Observations.

and motion results are displayed on Fig. 5. Visualization is made with the colored representation tool of the Middlebury database[1]. On these data, the memory size required by the full model is equal to the size of the image domain multiplied by the size of the temporal window and by the size of the state vector (which is 2), while those of the reduced model is equal to the size of the temporal window multiplied by the size of the reduced state vector (K + L, which is less than 10 in experiments). Results obtained with the reduced model have also been compared with four well-known state-of-the-art methods: [1, 10, 11, 12]. For these methods, optimal parameter values have been used, that provides the smallest errors. In Table 1, the error between the motion result and the ground truth is given for all methods.

[1] http://vision.middlebury.edu/flow/



(a) $t = 0$ (b) $t = 0$ (c) $t = 0$

(d) $t = 81$ (e) $t = 81$ (f) $t = 81$

Fig. 5. Left to right: groundtruth, full model, reduced model.

Table 1. Error analysis: misfit between motion results and ground truth.

Method	Ang. err. (in deg.) Mean	Ang. err. (in deg.) Std. Dev.	Relative norm err. Mean (in %)
Horn[1]	15.26	9.65	45.75
Isambert [10]	10.61	6.92	34.84
Suter[11]	10.41	5.34	37.65
Sun [12]	8.76	4.26	29.07
Full Model	0.18	0.10	0.65
Reduced Model	0.19	0.11	6.50

6. CONCLUSIONS

This paper describes a sliding windows assimilation method, that allows estimating motion on long temporal image sequences, thanks to data assimilation techniques. The method splits the initial temporal window in sub-windows, on which reduced models are computed that allow to process images in quasi-real time. The method is quantified with twin experiments to demonstrate its potential for processing long-term satellite data. The main perspective is to replace the bases Ψ_ξ of the reduced models, which are obtained with a Principal Order Decomposition, by a fixed basis. In that case, even the first sub-window could be processed by a reduced model, in order to further reduce the computational requirements. Moreover, this fixed basis should be defined as satisfying optimality criteria, which translate properties on motion fields and image data. In that way, the method will be able to process long satellite sequences acquired over a full basin, as the Black Sea.

7. REFERENCES

[1] B.K.P. Horn and B.G. Schunk, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[2] I. Cohen and I. Herlin, "Optical flow and phase portrait methods for environmental satellite image sequences," in *Proceedings of European Conference on Computer Vision*, Oxford, UK, April 1996, pp. 141–150.

[3] N. Papadakis and É. Mémin, "Variational optimal control technique for the tracking of deformable objects," in *Proceedings of International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007.

[4] O. Titaud, A. Vidard, I. Souopgui, and F.-X. Le Dimet, "Assimilation of image sequences in numerical models," *Tellus A*, vol. 62, pp. 30–47, 2010.

[5] D. Béréziat and I. Herlin, "Solving ill-posed image processing problems using data assimilation," *Numerical Algorithms*, vol. 56, no. 2, pp. 219–252, Feb. 2011.

[6] K. Drifi and I. Herlin, "Assimilation d'images dans un modèle réduit pour l'estimation du mouvement," in *GRETSI 2011 - Groupe d'Etudes du Traitement du Signal et des Images*, Bordeaux, France, Sept. 2011.

[7] P. Holmes, Lumley J. L., and G. Berkooz, *Turbulence, Coherent Structures, Symmetry and Dynamical Systems*, Cambridge Monographs on Mechanics and Applied Mathematics. Cambridge University Press, 1996.

[8] J. L. Lumley, "The structure of inhomogeneous turbulence," *Atmospheric Turbulence and Radio Wave Propagation*, pp. 166–178, 1967.

[9] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal, "Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization," *ACM Trans. Math. Softw.*, vol. 23, no. 4, pp. 550–560, Dec. 1997.

[10] T. Isambert, I. Herlin, and J.-P. Berroir, "Fast and stable vector spline method for fluid flow estimation," in *Proceedings of International Conference on Image Processing*, 2007, pp. 505–508.

[11] D. Suter, "Motion estimation and vector splines," in *Proceedings of Conference on Computer Vision and Pattern Recognition*, 1994, pp. 939–942.

[12] D. Sun, S. Roth, and M. Black, "Secrets of optical flow estimation and their principles," in *Proceedings of European Conference on Computer Vision*, 2010, pp. 2432–2439.

D.4 Complément au chapitre 4 : Coupling Reduced Models for Optimal Motion Estimation.

Cet article [15] a été publié dans la conférence ICPR 2012 [4] (« 21st International Conference on Pattern Recognition »). La conférence s'est déroulée à Tsukuba au Japon du 11 au 15 novembre 2012. Les travaux présentés dans cet article sont décrits dans le chapitre 4.

4. www.icpr2012.org/

Coupling Reduced Models for Optimal Motion Estimation

Karim Drifi[1,2], Isabelle Herlin[1,2]
[1]*INRIA, Institut National de Recherche en Informatique et Automatique,*
[2]*CEREA, joint laboratory ENPC - EDF R&D*
Karim.Drifi@inria.fr, Isabelle.Herlin@inria.fr

Abstract

The paper discusses the issue of motion estimation by image assimilation in numerical models, based on Navier-Stokes equations. In such context, models' reduction is an attractive approach that is used to decrease cost in memory and computation time. A reduced model is obtained from a Galerkin projection on a subspace, defined by its orthogonal basis. Long temporal image sequences may then be processed by a sliding-window method. On the first sub-window, a fixed basis is considered to define the reduced model. On the next ones, a Principal Order Decomposition is applied, in order to define a basis that is simultaneously small-size and adapted to the studied image data. Results are given on synthetic data and quantified according to state-of-the-art methods. Application to satellite images demonstrates the potential of the approach.

1. Introduction

Many authors investigate the issue of fluid flow motion estimation and a complete survey can be found for instance in [2]. In this paper, we are interested by the approach of data assimilation using a dynamic equation on the velocity field: motion is estimated as a compromise between that dynamics and image observations [1]. As the memory requirement and computation time of these data assimilation methods are proportional to image size, the issue of reduction arises. We describe, in Section 3, the reduction on a sine basis, whose results are analyzed in Section 5. This reduced model is applicable to estimate motion on a short temporal sequence. In Section 4, processing of long sequences is described with the use of the sine basis on a first short temporal sub-window and of Principal Order Decomposition on the following sub-windows. This is a coupling of reduced models with a sliding-window approach. Results are given in Section 5. The next section, Section 2, describes first the mathematical formalism of the paper and the model used for illustrating the coupling of reduced models.

2. Data Assimilation and Model Reduction

To illustrate our approach of model reduction, we consider divergence-free motion fields $\mathbf{w}(\mathbf{x}, t)$, with $\mathbf{x} = \begin{pmatrix} x & y \end{pmatrix}^T \in \Omega$, a bounded domain, $t \in [t_0, t_N]$, a closed interval, and $\mathbf{w} = \begin{pmatrix} u & v \end{pmatrix}^T$. We assume that the motion field satisfies the heuristics of Lagrangian constancy described by: $\frac{d\mathbf{w}}{dt} = \frac{\partial \mathbf{w}}{\partial t} + (\mathbf{w} \cdot \nabla)\mathbf{w} = 0$. This is rewritten as an equation on the evolution of the vorticity ξ under the divergence-free assumption:

$$\frac{\partial \xi}{\partial t} + \mathbf{w}(\xi) \cdot \nabla \xi = 0 \tag{1}$$

We consider a variable I_s, named pseudo-image, which has the same dynamics than the image observation: the motion field transports it according to:

$$\frac{\partial I_s}{\partial t} + \mathbf{w}(\xi) \cdot \nabla I_s = 0 \tag{2}$$

This pseudo-image is included in the state vector in order to allow an easy comparison with image observations at acquisition dates: they have to be almost identical. The state vector of the model is then defined as $\mathbf{X}(\mathbf{x}, t) = \begin{pmatrix} \xi(\mathbf{x}, t) & I_s(\mathbf{x}, t) \end{pmatrix}^T$, and Eqs. 1 and 2 are summarized as:

$$\frac{\partial \mathbf{X}}{\partial t} + \mathbb{M}(\mathbf{X}) = 0 \tag{3}$$

Data assimilation aims to find an optimal solution to the evolution equation (Eq. 3) and to the observation equation that links the state vector, in fact its pseudo-image component I_s, to image observations $I(\mathbf{x}, t)$:

$$I_s = I \tag{4}$$

Images are assimilated in the Full Model (FM), $\mathbb{M}$, in order to estimate vorticity and motion. Detailed description of the data assimilation method is given in [1].

In order to obtain a reduced model of $\mathbb{M}$, subspaces for vorticity fields and pseudo-images have to be chosen, defined by their respective orthogonal basis Φ and Ψ. Let $a_i(t)$ and $b_j(t)$ be the projection coefficients of $\xi(\mathbf{x},t)$ and $I_s(\mathbf{x},t)$ on Φ and Ψ, it comes: $\xi(\mathbf{x},t) \approx \sum_{i=1}^{K} a_i(t)\phi_i(\mathbf{x})$ and $I_s(\mathbf{x},t) \approx \sum_{j=1}^{L} b_j(t)\psi_j(\mathbf{x})$. After replacing in Eqs. 1 and 2, simplifying the equations, and using the property that $\mathbf{w}$ is a linear function of ξ, it comes:

$$\begin{cases} \dfrac{da_k}{dt}(t) + a^T(t)B(k)a(t) = 0, \quad k = 1 \dots K. \\ \dfrac{db_l}{dt}(t) + a^T(t)G(l)b(t) = 0, \quad l = 1 \dots L. \end{cases} \tag{5}$$

with:

- $a(t) = \begin{pmatrix} a_1(t) & \dots & a_K(t) \end{pmatrix}^T$,
- $b(t) = \begin{pmatrix} b_1(t) & \dots & b_L(t) \end{pmatrix}^T$,
- $B(k)$ a $K \times K$ matrix: $B(k)_{i,j} = \dfrac{\langle \mathbf{w}(\phi_i) \cdot \nabla \phi_j, \phi_k \rangle}{\langle \phi_k, \phi_k \rangle}$,
- $G(l)$ a $K \times L$ matrix: $G(l)_{i,j} = \dfrac{\langle \mathbf{w}(\phi_i) \cdot \nabla \psi_j, \psi_l \rangle}{\langle \psi_l, \psi_l \rangle}$
- $\langle ., . \rangle$ the scalar product: $\langle f, g \rangle = \int f(\mathbf{x})g(\mathbf{x})d\mathbf{x}$,
- $\mathbf{w}(\phi_i)$ the motion field associated with the vorticity field ϕ_i.

Let $\mathbf{X}_R(\mathbf{x},t) = \begin{pmatrix} a(t)^T & b(t)^T \end{pmatrix}^T$ be the state vector of the reduced model. System 5 is rewritten as:

$$\frac{d\mathbf{X}_R}{dt} + \mathbb{M}_R(\mathbf{X}_R) = 0 \tag{6}$$

$\mathbb{M}_R$ is the Galerkin projection of the full model $\mathbb{M}$ on Φ and Ψ.

3. Sine Basis

A sine basis Φ is chosen to define the vorticity subspace, whose element i is:

$$\phi_i = \phi_{(i_1,i_2)} = \begin{pmatrix} \sin(\pi i_1 h_x) sin(\pi i_2 h_y) \\ \dots \\ \sin(N_x \pi i_1 h_x) sin(\pi i_2 h_y) \\ \sin(\pi i_1 h_x) sin(2\pi i_2 h_y) \\ \dots \\ \sin(N_x \pi i_1 h_x) sin(2\pi i_2 h_y) \\ \dots \\ \sin(\pi i_1 h_x) sin(N_y \pi i_2 h_y) \\ \dots \\ \sin(N_x \pi i_1 h_x) sin(N_y \pi i_2 h_y) \end{pmatrix}$$

with:

- $i = (i_1, i_2)$ a double index,
- $h_x = \dfrac{1}{N_x + 1}$, N_x image size in direction x,
- $h_y = \dfrac{1}{N_y + 1}$, N_y image size in direction y.

ϕ_i is an eigenvector of the Laplace operator Δ associated to the eigenvalue

$$\lambda_i = -2[2 - \cos(\pi i_1 h_x) - \cos(\pi i_2 h_y)] \times \frac{1}{d^2} \tag{7}$$

d being the pixel resolution in both directions x and y. To compute $\mathbf{w}(\phi_i)$, we use this eigenvector property and derive the stream function φ_i that is solution of the Poisson equation:

$$\Delta \varphi_i = \phi_i \tag{8}$$

We have $\varphi_i = \phi_i / \lambda_i$ and $\mathbf{w}(\phi_i)$ is then derived from φ_i by:

$$\mathbf{w} = \begin{pmatrix} -\frac{\partial \varphi_i}{\partial y} & \frac{\partial \varphi_i}{\partial x} \end{pmatrix}^T \tag{9}$$

A Sine Reduced Model (SRM) is obtained by applying the Galerkin projection to the full model, defined by Eq. 3: vorticity is projected on Φ and pseudo-image is projected on itself (no reduction). As vorticity and motion are weighted sums of the sine functions, the result has the same spatial properties. In particular, the result is smooth, which is of major interest if image observations are noisy. This is a major advantage of SRM. However, this model has the same size as the full model if the full basis Φ is used. In this case, it offers no gain in memory size and computation time.

4. Sliding Windows

Having obtained the reduced model SRM for processing short temporal image sequences, the issue of processing long time intervals arises, which is solved by the sliding-window method.

The discrete sequence $I = \{I^z\}_{z=1\dots Z}$ is first split into short sub-sequences, for instance 4 images, that half overlap in time. The corresponding temporal intervals or windows are denoted W_m, with m being their index. Images belonging to W_1 are assimilated in SRM. This allows the retrieval of the vorticity on W_1. Its value at the beginning of W_2 is taken as initial condition for a simulation by the full model of Eq. 3. Principal Order Decomposition (POD) is then applied to the simulated sequences of vorticity ξ and pseudo-image I_s in order to generate bases Φ and Ψ and obtain a reduced model, named POD-POD Reduced Model of the 2nd window (PPRM2). The coefficients of projection of images belonging to W_2 are assimilated in PPRM2 to retrieve the vorticity coefficients and compute the vorticity values over W_2. This again provides the initial condition

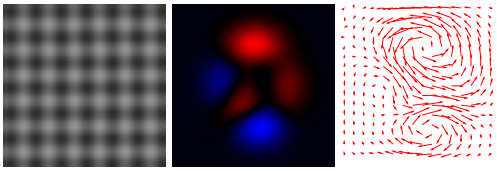


Figure 1. Pseudo-image, vorticity and motion field at $t = 0$. Positive vorticity values are red and negative one blue.

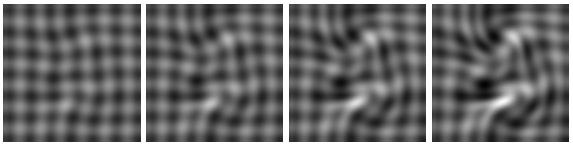


Figure 2. Four observations.

for W_3 and allows to define a new POD-POD Reduced Model on the third window (PPRM3). The process is then iterated until the whole image sequence I has been analyzed.

The major advantage of this approach is that assimilation in the Sine Reduced Model is only applied on the first temporal window W_1, that has short duration. On the next windows W_m, the complexity greatly decreases, as the state vectors involved in the POD-POD Reduced Models are of size less than 10 in the experiments.

5. Results

Results of the Sine Reduced Model are first provided on synthetic and satellite images. Then the sliding-window method is tested on synthetic data in order to demonstrate the potential of the method to process long temporal windows.

The divergence-free model is run from the initial conditions displayed in Figure 1. This provides a sequence of five observations (the first one is the initial condition and the next four are displayed on Figure 2) and the ground-truth of vorticity and motion over the whole temporal domain. Assimilation experiments are performed with these five observations in order to retrieve the vorticity and motion fields with the Full Model and the Sine Reduced Model. For these experiments, the background of vorticity (FM) or vector a (SRM) is set to zero and the one of pseudo-image is the first observation. The result of the assimilation process is the state vector $\mathbf{X}(k) = \begin{pmatrix} \xi(k) & I_s(k) \end{pmatrix}^T$ and its associated motion vector $\mathbf{w}(k)$ over the same temporal interval than the image sequence. In Table 1, the error

Table 1. Error analysis: misfit between motion results and ground truth.

Method	Ang. err. (in deg.) Mean	Ang. err. (in deg.) Std. Dev.	Relative norm err. Mean (in %)
Horn[3]	15.26	9.65	45.75
Papadakis [5]	13.89	5.03	45.59
Isambert [4]	10.61	6.92	34.84
Suter[7]	10.41	5.34	37.65
Sun [6]	8.76	4.26	29.07
FM	0.18	0.10	0.06
SRM	1.53	1.10	0.65

between the motion result and the ground truth is given for the Full Model, the Sine Reduced Model and five known state-of-the-art methods. For these five methods, optimal parameter values have been used. Four of them are image processing methods, that rely on L_2 regularization of motion [3, 6] or on a second-order regularization of the divergence [4, 7]. These methods are said static, as they do not use any model of motion evolution. Moreover, we compare with [5], that also applies data assimilation for a divergence-free model: the state vector reduces to vorticity and the observation equation is the optical flow equation. Results demonstrate the quality of the Full Model on this so-called twin experiment, and its efficient approximation by the Sine Reduced Model.

The approach is furthermore applied on satellite data. Observations are images acquired by NOAA/AVHRR sensors over Black Sea [1], and measure the Sea Surface Temperature (SST) with a spatial resolution of about 1 km at nadir. In the upper layer of Black Sea, horizontal motion is around 30 $^{cm}/_{s}$ for mesoscale eddies, while vertical motion is around 10^{-4} $^{cm}/_{s}$ and can be neglected. The 2D divergence-free assumption is then roughly verified and the Full Model and Sine Reduced model are applicable. The sequence has four observations (see two of them on Figure 3). The results of motion estimation with FM and SRM are displayed on the same figure. Visualisation is made with the coloured representation tool of the Middlebury database [2]. On these coloured images, the orientation and norm of velocity are respectively represented by hue (colour) and saturation.The data assimilation methods also compute the pseudo-image values, that achieve the best compromise between dynamics and observations. At acquisition dates, these pseudo-images are not exactly equal

[1] Data have been provided by E. Plotnikov and G. Korotaev from the Marine Hydrophysical Institute of Sevastopol, Ukraine.

[2] http://vision.middlebury.edu/flow/

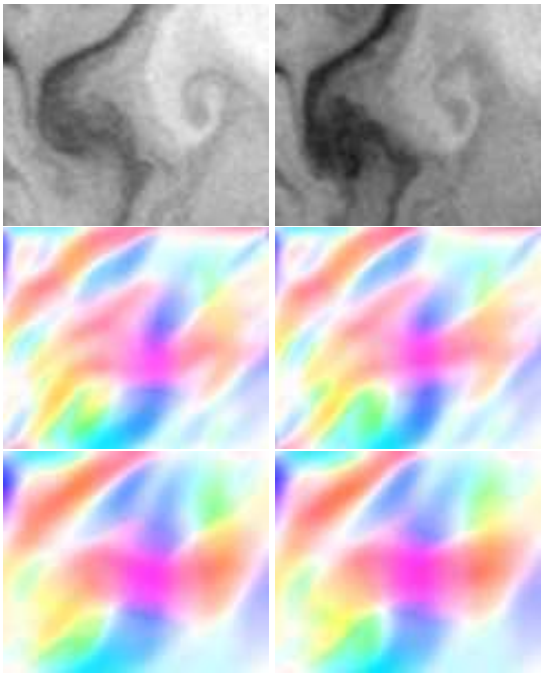


Figure 3. Observations (top), FM (middle), SRM (bottom) at $t = 1$ (left) and 3 (right).

Table 2. Correlation between pseudo-images and observations.

Date	1	2	3	4
FM	0.99	0.93	0.94	0.97
SRM	0.99	0.94	0.94	0.96

to the observed images. Their correlation measures if the structures (edges) are correctly assessed and if motion is accurately estimated. Correlation results of FM and SRM are given in Table 2: values are close to 1, proving that the motion retrieved by both models are coherent with the dynamics underlying the evolution displayed by the observations. This also points out the performance of SRM. Another mathematical criteria is the RMSE between estimations of vorticity by FM and SRM. Its value is 0.01, which proves that SRM is a good approximation of FM.

The sliding-window method described in Section 4 is then applied on a sequence of 19 image observations, obtained from a run of the Full Model with initial conditions of Figure 1. The discrete sequence is split in 7 windows of five images. The first five observations of W_1 are assimilated in SRM. The result is used to define the POD-POD Reduced Model (PPRM2) of W_2. The five observations of W_2 are then assimilated in PPRM2 and so on until the end of the studied sequence. Comparison of estimated vorticity with ground-truth gives that the RMSE ranges from 0.0016 on W_2 to 0.005 on W_7, which demonstrates the robustness obtained by coupling the Sine Reduced Model with the coupled POD-POD Reduced Models.

6. Conclusions

The paper describes an approach for coupling reduced models and optimally estimate motion on long temporal image sequences. One of these reduced models is the Sine Reduced Basis, which uses a fixed basis and presents smoothness properties. Its results have been quantified and compared to state-of-the-art methods. The second reduced model is the Pod-Pod Reduced Model, which relies on Principal Order Decomposition. Its state vector has a small size, less than 10 in experiments, that allow processing long sequences in almost real time.

7. Acknowledgements

Research has been partly funded by the ANR project Geo-FLUIDS (ANR 09 SYSC 005 02).

References

[1] D. Béréziat and I. Herlin. Solving ill-posed image processing problems using data assimilation. *Numerical Algorithms*, 56(2):219–252, Feb. 2011.

[2] D. Heitz, E. Mémin, and C. Schnörr. Variational fluid flow measurements from image sequences: synopsis and perspectives. *Experiments in Fluids*, 48(3):369–393, 2010.

[3] B. Horn and B. Schunk. Determining optical flow. 17:185–203, 1981.

[4] T. Isambert, I. Herlin, and J.-P. Berroir. Fast and stable vector spline method for fluid flow estimation. In *Proceedings of International Conference on Image Processing*, pages 505–508, 2007.

[5] N. Papadakis, T. Corpetti, and E. Mémin. Dynamically consistent optical flow estimation. In *Proceedings of International Conference on Computer Vision*, pages 1–7, 2007.

[6] D. Sun, S. Roth, and M. Black. Secrets of optical flow estimation and their principles. In *Proceedings of European Conference on Computer Vision*, pages 2432–2439, 2010.

[7] D. Suter. Motion estimation and vector splines. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, pages 939–942, 1994.

D.5 Complément au chapitre 5 : Motion estimation on ocean satellite images by data assimilation in a wavelets reduced model .

Cet abstract a été accepté dans la conférence EGU 2013[5] (« European Geosciences Union General Assembly 2013 »). La conférence se déroulera à Vienne en Autriche du 7 au 12 avril 2013. Les travaux présentés dans cet abstract sont décrits dans le chapitre 5.

5. www.egu2013.eu/

Geophysical Research Abstracts
Vol. 15, EGU2013-**PREVIEW**, 2013
EGU General Assembly 2013
© Author(s) 2013. CC Attribution 3.0 License.

Motion estimation on ocean satellite images by data assimilation in a wavelets reduced model

Etienne Huot (1,2,3), Giueseppe Papari (4), Isabelle Herlin (1,2), Karim Drifi (1,2)

(1) INRIA, Institut National de Recherche en Informatique et Automatique, Paris-Rocquencourt, France, (2) CEREA, Joint Lab. Ecole nationale des Ponts ParisTech - EDF R&D, Université Paris-Est, (3) Université Versailles St-Quentin, France, (4) Trollhetta AS, Trondheim, Norway

The presentation concerns the issue of estimating surface motion from satellite images, either on rectangular windows or for a whole basin, with complex geographical boundary, such as Black Sea. The spatial domain is in all cases denoted by Ω.

The approach relies on a reduced model that is obtained by Galerkin projection of dynamic equations on subspaces of velocity and image fields. The dynamic model expresses usual assumptions such as the transport of image brightness by motion and advection-diffusion of velocity.

Subspaces of image and motion fields are obtained as finite dimension vectorial spaces with scalar and vector wavelets bases, respectiveley denoted ψ_l, for $l = 1 \dots L$ and ϕ_k, for $k = 1 \dots K$. A method has been defined in order to obtain the ψ_l and ϕ_k elements as solutions of constrained minimization problems described by:

$$\begin{cases} \min\limits_{(\psi_1,\dots,\psi_n)\in\mathfrak{F}^n} \sum\limits_{k=1}^{n} \mathcal{Q}(\psi_k) \\ \mathcal{B}(\psi_k) = 0, \quad k = 1, ..., n \\ \langle \psi_j, \psi_k \rangle = \delta_{j,k} \end{cases} \tag{1}$$

where:

- $\mathfrak{F}$ is a Hilbert functional space, with given inner product $\langle \cdot, \cdot \rangle$,
- $\mathcal{Q}(\psi) = \langle \mathcal{L}(\psi), \psi \rangle$ is a positively defined quadratic functional,
- $\mathcal{B}$ is a linear operator on $\mathfrak{F}$, used to express the chosen boundary conditions on image and motion fields,
- and $\delta_{j,k}$ is equal to 1 for $j = k$ and to 0 otherwise.

In order to define the scalar basis of the image subspace, the minimization problem is solved with $\mathcal{Q}(\psi) \triangleq \int_\Omega \nabla\psi(\mathbf{r})^2 \mathrm{d}^2\mathbf{r}$, and $\mathcal{B}(\psi) \triangleq \mathbf{n}(\mathbf{r})^\intercal \psi(\mathbf{r})$, where the vector $\mathbf{n}(\mathbf{r})$ is zero everywhere apart for the boundary $\partial\Omega$, and it is $\mathbf{n}(\mathbf{r}) = 1, \mathbf{n}(\mathbf{r}) \perp \partial\Omega, \forall \mathbf{r} \in \partial\Omega$. Thus, we are minimizing the smoothness $\int_\Omega \nabla\psi(\mathbf{r})^2 \mathrm{d}^2\mathbf{r}$ with the Neumann boundary conditions.

In order to define the vector basis of the motion subspace, the minimization problem (1), is considered with $\mathcal{Q}(\psi) \triangleq \int_\Omega \nabla\psi(\mathbf{r})^2 \mathrm{d}^2\mathbf{r}$, where $\psi : \Omega \mapsto \mathbb{R}^2$ is a planar vector field. As to the operator $\mathcal{B}(\psi)$, two possibilities are available:

$$\{\mathcal{B}(\psi)\}(\mathbf{r}) = \begin{cases} \mathbf{n}(\mathbf{r})^\intercal \psi(\mathbf{r}), & \mathbf{r} \in \partial\Omega \\ 0, & otherwise \end{cases}$$

$$\{\mathcal{B}(\psi)\}(\mathbf{r}) = \begin{cases} \mathbf{n}(\mathbf{r})^\intercal \psi(\mathbf{r}), & \mathbf{r} \in \partial\Omega \\ \{\mathrm{div}\psi\}(\mathbf{r}), & otherwise \end{cases}$$

In both cases, Neumann boudary conditions are applied, and in the second case, motion is considered as divergence-free.

The sequence of satellite images is projected on the image subspace in order to get a sequence of observation vectors: $b(t) = (\, b \,)_1 (t) \dots b_L(t)^T$. These $b(t)$ are then assimilated in the reduced model in order to estimate the

values $a(t) = (\,a\,)_1\,(t) \ldots a_K(t)^T$, $a_i(t)$ being the coefficient of the projection of motion $\mathbf{w}(t)$ on the basis vector ψ. $\mathbf{w}(t)$ is then obtained as $\sum_{i=1}^{K} a_i(t)\psi_i$.

The approach has been used on satellite data acquired by NOAA-AVHRR sensors and on the MyOCean[1] analysis database for BlackSea.

[1] http://www.myocean.eu.org/

Bibliographie

[1] Dominique Béréziat and Isabelle Herlin. Using models of dynamics for large displacement estimation on noisy acquisitions. Rapport de recherche RR-7408, INRIA, October 2010.

[2] Dominique Béréziat and Isabelle Herlin. Solving ill-posed Image Processing problems using Data Assimilation. *Numerical Algorithms*, 56(2) : 219–252, 2011.

[3] F. Bouttier and P. Courtier. Data assimilation concepts and methods. Meteorological training course lecture series, ECMWF, 1999.

[4] John Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6) : 679 –698, nov. 1986.

[5] Alexandre J. Chorin. Numerical study of slightly viscous flow. *Journal of Fluid Mechanics Digital Archive*, 57(04) : 785–796, 1973.

[6] Th. Corpetti, P. Héas, E. Mémin, and N. Papadakis. Variational pressure image assimilation for atmospheric motion estimation. In *Proc. Int. Geoscience and Remote Sensing Symp. (IGARSS'08)*, volume 2, pages 505–508, Boston, MA, July 2008.

[7] Thomas Corpetti, étienne Mémin, and Patrick Pérez. Dense estimation of fluid flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3) : 365–380, 2002.

[8] Georges-Henri Cottet and Petros Koumoutsakos. *Vortex methods : Theory and practice*. Cambridge University Press, June 2000. 514 pages.

[9] Anne Cuzol, Pierre Hellier, and Etienne Memin. A low dimensional fluid motion estimator. *International Journal of Computer Vision*, 75(3) : 329–349, December 2007.

[10] Juan D'Adamo, Nicolas Papadakis, Etienne Memin, and Guillermo Artana. Variational assimilation of POD low-order dynamical systems. *Journal of Turbulence*, 8(9) : 1–22, 2007.

[11] A. E. Deane, I. G. Kevrekidis, G. E. Karniadakis, and S. A. Orszag. Low-dimensional models for complex geometry flows - Application to grooved channels and circular cylinders. *Physics of Fluids*, 3 : 2337–2354, October 1991.

[12] Rachid Deriche. Using Canny's criteria to derive a recursively implemented optimal edge detector. *International Journal of Computer Vision*, 1 : 167–187, 1987. 10.1007/BF00123164.

[13] François-Xavier Le Dimet. Une application des méthodes de contrôle optimal à l'analyse variationnelle. Technical report, Université Blaise-Pascal, Aubière, 1982.

[14] Karim Drifi and Isabelle Herlin. Assimilation d'images dans un modèle réduit pour l'estimation du mouvement. In *GRETSI 2011 - Groupe d'Etudes du Traitement du Signal et des Images*, Bordeaux, France, September 2011.

[15] Karim Drifi and Isabelle Herlin. Coupling reduced models for optimal motion estimation. In *International Conference on Pattern Recognition - ICPR 2012*, pages 2651–2654, 2012.

[16] P. Dérian, P. Héas, C. Herzet, and E. Mémin. Wavelets to reconstruct turbulence multifractals from experimental image sequences. In *7th Int. Symp. on Turbulence and Shear Flow Phenomena, TSFP-7*, Ottawa, Canada, July 2011.

[17] Jacques Hadamard. *Lectures on Cauchy's problem in linear partial differential equations*. New Haven, 1923.

[18] Isabelle Herlin, Dominique Béréziat, Karim Drifi, and Sergiy Zhuk. Learning reduced models for motion estimation on ocean satellite images. In *Hydrodynamic modeling of the Black Sea Dynamics*, Sevastopol, Ukraine, September 2011.

[19] Isabelle Herlin, Dominique Béréziat, and Nicolas Mercier. Strategies for processing images with 4D-Var data assimilation methods. Rapport de recherche RR-7495, INRIA, December 2010.

[20] Isabelle Herlin, Dominique Béréziat, Nicolas Mercier, and Sergiy Zhuk. Divergence-Free Motion Estimation. In Svetlana Lazebnik Andrew Fitzgibbon, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *ECCV 2012 - European Conference on Computer Vision*, volume 7575 of *Lecture Notes in Computer Science*, pages 15–27, Florence, Italie, October 2012. Springer.

[21] Isabelle Herlin and Karim Drifi. Learning reduced models for motion estimation on long temporal image sequences. In *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Munich, Allemagne, 2012.

[22] Isabelle Herlin and Etienne Huot. Monitoring surface currents from uncertain image observations. In *XX International Conference on Problems of Decision Making under Uncertainties (PDMU)*, Brno, Tchèque, République, September 2012.

[23] Philip Holmes, John Leask Lumley, and Gal Berkooz. *Turbulence, Coherent Structures, Symmetry and Dynamical Systems*. Cambridge Monographs on Mechanics and Applied Mathematics. Cambridge University Press, 1996.

[24] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artifical intelligence*, 17 : 185–203, 1981.

[25] Étienne Huot, Isabelle Herlin, Nicolas Mercier, and Evgeny Plotnikov. Estimating apparent motion on satellite acquisitions with a physical dynamic model. In *International Conference on Pattern Recognition - ICPR 2010*, pages 41–44, Istambul, Turquie, August 2010. IEEE.

[26] Till Isambert, Isabelle Herlin, and Jean-Paul Berroir. Fast and stable vector spline method for fluid apparent motion estimation. In *ICIP 2007 - Proceedings of the International Conference on Image Processing (ICIP)*, volume 6, pages 505–508, San Antonio, Texas, États-Unis, 2007.

[27] Kalman, Rudolph, and Emil. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D) :35–45, 1960.

[28] R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Transactions of the ASME. Series D, Journal of Basic Engineering*, 83 : 95–107, 1961.

[29] G.K. Korotaev, E. Huot, F.-X. Le Dimet, I. Herlin, S.V. Stanichny, D.M. Solovyev, and L. Wu. Retrieving ocean surface current by 4-d variational assimilation of sea surface temperature images. *Remote Sensing of Environment*, 112(4) : 1464 – 1475, 2008. Remote Sensing Data Assimilation Special Issue.

[30] François-Xavier Le Dimet and Olivier Talagrand. Variational algorithms for analysis and assimilation of meteorological observations : theoretical aspects. *Tellus*, 38A : 97–110, 1986.

[31] M. Loève. *Probability Theory*. Springer-Verlag, 1978.

[32] John Leask Lumley. The structure of inhomogeneous turbulence. *Atmospheric Turbulence and Radio Wave Propagation*, pages 166–178, 1967.

[33] Olivier Musse, Fabrice Heitz, and Jean-Paul Armspach. 3D deformable image matching using multiscale minimization of global energy functions. In *IEEE Int. Conf. Computer Vision Pattern Recognition, CVPR '99*, Fort Collins, USA, Jun 1999.

[34] N. Papadakis, T. Corpetti, and E. Mémin. Dynamically consistent optical flow estimation. In *Proc. Int. Conf. Comp. Vis.(ICCV'07)*, pages 1–7, Rio de Janeiro, Brazil, October 2007.

[35] Nicolas Papadakis. *Assimilation de données images : application au suivi de courbes et de champs de vecteurs*. Thèse, Université Rennes 1, November 2007.

[36] M. Raffel, C.E. Willert, S.T. Wereley, and J. Kompenhans. *Particle Image Velocimetry : A Practical Guide*. Experimental Fluid Mechanics. Springer London, Limited, 2007.

[37] Mojtaba Rajaee, Sture K. F. Karlsson, and Lawrence Sirovich. Low-dimensional description of free-shear-flow coherent structures and their dynamical behaviour. *Journal of Fluid Mechanics*, 258 : 1–29, 1994.

[38] D. Rempfer. Investigations of boundary layer transition via Galerkin projections on empirical eigenfunctions. *Physics of Fluids*, 8 : 175–188, January 1996.

[39] Innocent Souopgui. *Assimilation d'images pour les fluides géophysiques*. Thèse, Université Joseph-Fourier - Grenoble I, October 2010.

[40] Sridhar Srinivasan and Rama Chellappa. Optical flow using overlapped basis functions for solving global motion problems. In Hans Burkhardt and Bernd Neumann, editors, *Computer Vision — ECCV'98*, volume 1407 of *Lecture Notes in Computer Science*, pages 288–304. Springer Berlin / Heidelberg, 1998. 10.1007/BFb0054748.

[41] Deqing Sun, Stefan Roth, and Michael J. Black. Secrets of optical flow estimation and their principles. In *In Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2432–2439. IEEE, June 2010.

[42] D. Suter. Motion Estimation and Vector Splines. In *Proc. Conf. Comp. Vision Pattern Rec*, pages 939–942. IEEE, 1994.

[43] Richard Szeliski and Heung-Yeung Shum. Motion estimation with quadtree splines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18 : 1199–1210, 1996.

[44] A. N. Tikhonov. Regularization of incorrectly posed problems. *Sov. Math. Dokl.*, 4 :1624–1627, 1963.

[45] Olivier Titaud, Arthur Vidard, Innocent Souopgui, and François-Xavier Le Dimet. Assimilation of Image Sequences in Numerical Models. *Tellus Series A : Dynamic meteorology and oceanography*, 62(1) : 30–47, 2010.

[46] Yu-Te Wu, Takeo Kanade, Ching-Chung Li, and Jeffrey Cohn. Image registration using wavelet-based motion model. *International Journal of Computer Vision*, 38 : 129–152, 2000. 10.1023/A : 1008101718719.

[47] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778 : L-BFGS-B : Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans. Math. Softw.*, 23(4) : 550–560, December 1997.